Chapter 6

Solving the Model on a Computer

The practical use of linear programming as a farm planning tool has been greatly advanced in recent years by the development of computer software packages based on the simplex method. These packages are widely available on modern computer systems and offer a cheap and efficient way to solve even large models. Most packages also provide the user with options for post-optimality analysis for exploring the stability of the solution to changes in the model's coefficients.

One of the most popular linear programming packages is IBM's MPSX (Mathematical Programming System Extended) package. A similar package in terms of the input/output formats is CDC's APEX package. In this chapter we describe how to set up a simple linear programming problem for MPSX and illustrate the procedures involved and the nature of the optimal solution with the simple Mayaland farm example. The discussion draws heavily on Stanton (1977), and is not intended to be an exhaustive treatment of the facilities of the MPSX package.[1]

6.1 PREPARATION OF A LINEAR PROGRAMMING PROBLEM FOR SOLUTION ON A COMPUTER

Submission of a linear programming problem for solution by MPSX requires the following sets of cards:

1 Job and Catalog cards
2 Program cards
3 Data input cards

6.1.1 The Job and Catalog Cards

These cards are unique for each computer system, hence they are not discussed here. They are required to indicate the job to be performed, space requirements, and the procedure to be followed.

6.1.2 The Program Cards

These cards tell the computer which options available in the MPSX package are to be used and specify the sequence that should be followed. Order is important in most cases if the linear programming problem is to be solved. A set of program cards that will solve a standard linear programming problem follows together with some comments on each card. All cards are punched starting in column 10.

Card	*Comment*
`PROGRAM`	Signifies the beginning of the program cards.
`INITIALZ`	Calls a basic set of instructions that are standard for handling all the program cards that follow.
`TITLE('....')`	This optional card allows a user to establish a title for the problem which will be listed at the top of each page of results in the computer output. The title may consist of up to 40 characters.
`MOVE(XDATA, '....')`	This card allows the user to designate the name of the data set that will be used and stored in the system. Up to 8 characters may be used.
`MOVE(XPBNAME, '....')`	This card allows the user to designate which problem is being run using a given data set. Up to 8 characters may be used in making up such names.
`MOVE(XOBJ, '....')`	This card allows the user to designate the name of the objective function row for the problem. It cannot have more than 8 char-

	acters and must be listed exactly the same way in the data input section.
`MOVE(XRHS, '....')`	This card designates the name of the right-hand side of the problem set. Again this name must be exactly the same as the one used in the data input section, and can consist of up to 8 characters.
`CONVERT('SUMMARY')`	Tells the computer to read the data input, convert it to a binary problem format and store the data in a problem file.
`SETUP('MAX')`	Indicates whether the linear programming problem is a maximization ('MAX') or minimization problem ('MIN'). Maximization is assumed if no statement is specified following SETUP.

At this point four optional cards may be included in the program deck to provide additional information to the user in the printout for the problem.

`BCDOUT`	Prints a listing of the data input that allows the user to review the data set analyzed by the computer.
`PICTURE`	Prints the basic matrix to be solved in diagrammatic form. All numbers other than ±1 are converted to an alphabetic code according to their magnitude.
`TRANCOL`	Prints the basic matrix to be solved in numerical form. For most linear programming problems this is the most useful way to compare the data read into the computer with the original matrix constructed in setting up the problem.
`CRASH`	Provides a routine that establishes the most efficient starting point for solving the problem. For small problems it provides little assistance in reaching efficient solutions.

This is the end of this set of optional program cards for standard linear programming problems. The next two cards must be included for all programs.

`PRIMAL`	Invokes the optimizing procedure using the simplex method to search for the optimal solution.
`SOLUTION`	Directs that the optimal solution obtained be printed.

Another set of optional program cards may be introduced at this point to provide additional analytical material to interpret the original solution or to extend the analysis. These extensions include postoptimality procedures as discussed in Sections 6.3 and 6.4.

The remaining program cards that must be included are as follows:

`EXIT`	Provides directions to the computer that the problem set is completed.
`PEND`	Designates the end of the program card deck.

6.1.3 The Data Input Cards

These cards provide the computer with the necessary information about the problem to be solved. They are organized into three sections. A ROWS section defines the name of each of the constraints or rows in the problem, including the objective function. It also specifies the type of constraint. A COLUMNS section defines the activity names in the problem and indicates the data coefficients that are to be entered in each row. Up to two data entries are permitted per card. Finally, a RHS section provides information about the supplies of the fixed farm resources. Detailed instructions for the preparation of the data input cards now follow.

`NAME`	This card is always the first one in the data input deck. NAME is written starting in column 1. The name for the data deck, which must be the same as the name in the XDATA program card, is written starting in column 15.
`ROWS`	An initial card starting in column 1 designating that the cards that follow describe the rows of the problem. Each of the cards that follow have the following format:

Card Columns:	2 or 3	5–12
	Type of constraint	Row name

The type of constraint is coded as L = less than or equal to; G = greater than or equal to; E = equal to; or N = not constrained (such as the objective function). Row names can consist of up to 8 characters, and the name of the objective function row must correspond to the name in the XOBJ program card. The order of the rows does not effect the solution but does determine the order in which they appear in the results.

Table 6.1 Coded Program and Data Cards for the Mayaland Farm MPSX Linear Programming Package

```
(JCL cards)

          PROGRAM
          INITIALZ
          TITLE('MAYALAND FARM')
          MOVE(XDATA,'MAYALAND')
          MOVE(XPBNAME,'PROB1')
          MOVE(XOBJ,'OBJ')
          MOVE(XRHS,'B1')
          CONVERT('SUMMARY')
          SETUP('MAX')
          BCDOUT
          PICTURE
          TRANCOL
          PRIMAL
          SOLUTION
          EXIT
          PEND
(JCL cards)
NAME           MAYALAND
ROWS
 N   OBJ
 L   LAND
 L   LABOR
 L   MULES
 L   MARKET
COLUMNS
    CORN       OBJ         1372.        LAND          1.
    CORN       LABOR          1.42      MULES         1.45
    BEANS      OBJ         1219.        LAND          1.
    BEANS      LABOR          1.87      MULES         1.27
    SORGHUM    OBJ         1523.        LAND          1.
    SORGHUM    LABOR          1.92      MULES         1.16
    PEANUTS    OBJ         4874.        LAND          1.
    PEANUTS    LABOR          2.64      MULES         1.45
    PEANUTS    MARKET         0.983
RHS
    B1         LAND           5.        LABOR        16.5
    B1         MULES         10.        MARKET         .5
ENDATA
(JCL cards)
```

COLUMNS This card, which is punched beginning in column 1, designates the beginning of the data cards for the activities or columns of the linear programming problem. Each of the cards that follows has the following format:

Card Columns:	5–12	15–22	25–36	40–47	50–61
	Column name	Row name	Numerical data	Row name	Numerical data

Each column name can consist of up to 8 characters, and the row names must correspond to those in the ROWS section. Decimals must always be entered, and all the cards corresponding to one column should be grouped together.

RHS The card RHS, punched beginning in column 1, designates that the cards that follow give the right-hand side values of the problem. The corresponding data cards are prepared as follows:

Card Columns:	5–12	15–22	25–36	40–47	50–61
	Right-hand side name	Row name	Numerical data	Row name	Numerical data

The right-hand side name must correspond to that in the XRHS program card. The row names must match those in the ROWS section, and decimals must always be included.

ENDATA This card indicates that the data input has been completed.

To illustrate the preparation of a deck of cards for a linear programming problem, Table 6.1 shows the relevant cards for the Mayaland farm problem in Table 2.2.

6.2 INTERPRETATION OF THE SOLUTION OUTPUT

A problem submitted for solution using the MPSX Linear Programming Package yields a substantial number of pages of computer output if there are no errors and a solution is obtained. The amount of output is determined by the control cards and the size of the problem. In general, the relevant part of the printout consists of two parts.

1 A listing and detailed description of the problem submitted, including a list of the job control cards, a check for major and minor errors

in the data, and a printout of data if BCDOUT, PICTURE, or TRANCOL is requested.

2 The optimal solution.

The first page of the output of the optimal solution lists the number of iterations computed before an optimal solution was reached. In the case of the Mayaland farm example, only two iterations were required before the optimal solution was found. This is followed by a separate page reporting the final value of the objective function; 9319 pesos for the Mayaland farm.

The main printout of the solution consists of a ROWS and COLUMNS section. The ROWS section indicates the status of each of the constraint rows in the problem. It tells whether or not the limiting resources were fully used and lists the DUAL values, or shadow prices, for all the constraints that are used up to capacity.

For the Mayaland farm information is reported in five rows (Table 6.2). The first is the objective function, OBJ, and the other four are the farm constraints.

The columns headed ACTIVITY and SLACK ACTIVITY report the amounts of resources used in the solution, and the amounts of slack or unused resources, respectively. Thus in row 2 all the land was used, and in row 5 the market constraint has been used to the maximum permitted level. However, rows 3 and 4 show that 6.5338 months of labor and 4.0525 months of mules are left slack.

The columns headed LOWER and UPPER LIMIT simply restate the greater than (LOWER) or less than (UPPER) constraints of the problem. All the constraints in Table 6.2 were $\leq$ constraints.

The last column, DUAL ACTIVITY, reports the shadow prices (or dual values) for the fixed resources and constraints that are fully used. These shadow prices are calculated for each resource as the cost to the objective function value if one unit of the resource were withdrawn from use by increasing the corresponding slack activity by one unit. All the shadow prices are therefore negative. The shadow price for land in Table 6.2 is -1523 pesos. This means that if the farmer were to lose 1 hectare of land then his maximum attainable total gross margin would decline by 1523 pesos. The negative of a shadow price indicates the maximum amount by which the model's objective function could be increased if an additional unit of the resource were to become available. Thus, in the case of land, 1523 pesos is the maximum rent that the farmer should be willing to pay for an extra hectare of land beyond his initial 5 hectares.

The shadow price on the market constraint is -3409 pesos. The negative of this amount is the extra total gross margin that could be earned if the farmer could increase his peanut sales by 1 ton. This might be possible by investing in a truck, or by devoting more time to expanding the number of customers.

Table 6.2 MPSX / Computer Output — Status of Rows in Optimal Solution, Mayaland Farm Example

MPSX / 370 MAYALAND FARM
SECTION 1 - ROWS

NUMBER	ROW	AT	ACTIVITY	SLACK ACTIVITY	LOWER LIMIT	UPPER LIMIT	DUAL ACTIVITY
1	OBJ	BS	9319.47609	9319.47609 -	NONE	NONE	1.00000
2	LAND	UL	5.00000	.	NONE	5.00000	1523.00000 -
3	LABOR	BS	9.96623	6.53377	NONE	16.50000	.
4	MULES	BS	5.94751	4.05249	NONE	10.00000	.
5	MARKET	UL	0.50000	.	NONE	0.50000	3408.95219 -

Table 6.3 MPSX Computer Output — Status of Columns in Optimal Solution, Mayaland Farm Example

MPSX / 370 MAYALAND FARM
SECTION 2 - COLUMNS

NUMBER	COLUMNS	AT	ACTIVITY	INPUT COST	LOWER LIMIT	UPPER LIMIT	REDUCED COST
6	CORN	LL	.	1372.00000	.	NONE	151.00000 -
7	BEANS	LL	.	1219.00000	.	NONE	304.00000 -
8	SORGHUM	BS	4.49135	1523.00000	.	NONE	.
9	PEANUTS	BS	0.50865	4874.00000	.	NONE	.

The COLUMNS section of the computer output (Table 6.3) provides information about the optimal farm plan. Each of the farm activities is listed as a row in this output.

The column headed ACTIVITY reports the status of each activity in the final solution. Thus the optimal plan for the Mayaland farm consists of 4.491 hectares of sorghum, 0.509 hectares of peanuts, and zero hectares of corn and beans.

The entries in the column headed INPUT COST are simply the per hectare gross margins from the data input. The LOWER and UPPER LIMIT columns indicate any constraints that were imposed on the individual activity levels. The LOWER LIMIT values are all zero, implying that negative activity levels are not permitted. Positive LOWER LIMITS and finite UPPER LIMITS only become effective if the MPSX option BOUNDS is used. This option enables the user to specify upper or lower limits on any activity. It is a more efficient way of introducing such constraints than adding additional constraints in the ROWS and RHS sections. The last column headed REDUCED COST is similar to a shadow price for the resource constraints. It is the amount by which the objective function value would decline if a unit of the corresponding activity were forced into the solution. Equivalently, the negative of the REDUCED COST is the amount by which the activity per hectare gross margin would have to be increased before that activity is profitable enough to be included in the optimal farm plan. The gross margins would have to increase by 151 pesos and 304 pesos per hectare, respectively, for corn and beans before these activities could be included in an optimal solution. The reduced cost coefficients are of course zero for those activities already in the optimal plan.

6.3 POSTOPTIMALITY ANALYSIS USING THE MPSX OPTION RANGE

The MPSX option RANGE provides information about the stability of the optimal linear programming solution. Stability is tested under a *ceteris paribus* condition, whereby the effect of a change in a single coefficient is considered with all other coefficients held constant. The stability of the solution refers to the degree of variation in the coefficient that can be absorbed by the model before a change in basis occurs. A change in basis is said to occur when a new activity enters the solution, or one previously in solution drops out. The value of the coefficient at which the change in basis occurs is a critical turning point, and the change in a coefficient required to span two critical turning points is referred to as the range for that coefficient. Stability thus depends on the magnitude of the range for each coefficient under the *ceteris paribus* condition.

To obtain a RANGE printout along with the optimal solution, an additional card is inserted immediately after the SOLUTION card in the program cards section of the input deck (see Section 6.1). This card contains the word RANGE punched beginning in column 10.

Table 6.4 Computer Output — Rows at Limit Level; RANGE Analysis, Mayaland Farm Example

MPSX / 370 MAYALAND FARM
SECTION 1 - ROWS AT LIMIT LEVEL

NUMBER	ROW	AT	ACTIVITY	SLACK ACTIVITY	LOWER LIMIT UPPER LIMIT	LOWER ACTIVITY UPPER ACTIVITY	UNIT COST	LOWER COST UPPER COST	LIMITING PROCESS	AT
2	LAND	UL	5.00000	.	NONE	0.50865	1523.00000-		SORGHUM	LL
					5.00000	8.40301	1523.00000		LABOR	UL
5	MARKET	UL	0.50000	.	NONE	.	3408.952158-		PEANUTS	LL
					0.50000	4.91500	3408.95215		SORGHUM	LL

Table 6.5 MPSX Computer Output — Columns at Limit Level; RANGE Analysis, Mayaland Farm Example

MPSX / 370 MAYALAND FARM
SECTION 2 - COLUMNS AT LIMIT LEVEL

NUMBER	COLUMN	AT	ACTIVITY	INPUT COST	LOWER LIMIT UPPER LIMIT	LOWER ACTIVITY UPPER ACTIVITY	UNIT COST	LOWER COST UPPER COST	LIMITING PROCESS	AT
6	CORN	LL	.	1372.00000	.	13.06755-	151.00000	INFINITY-	LABOR	UL
					NONE	4.49135	151.00000-	1523.00000	SORGHUM	LL
7	BEANS	LL	.	1219.00000	.	130.67548-	304.00000	INFINITY-	LABOR	UL
					NONE	4.49135	304.00000-	1523.00000	SORGHUM	LL

Table 6.6 MPSX Computer Output — Rows at Intermediate Level; RANGE Analysis, Mayaland Farm Example

MPSX / 370 MAYALAND FARM
SECTION 3 - ROWS AT INTERMEDIATE LEVEL

NUMBER	ROW	AT	ACTIVITY	SLACK ACTIVITY	LOWER LIMIT UPPER LIMIT	LOWER ACTIVITY UPPER ACTIVITY	UNIT COST	LOWER COST UPPER COST	LIMITING PROCESS	AT
3	LABOR	BS	9.96623	6.53377	NONE	7.72055	302.00000 -		CORN	LL
					16.50000	9.96623	INFINITY -		NONE	
4	MULES	BS	5.94751	4.05249	NONE	0.73754	1312.93115 -		LAND	UL
					10.00000	7.25000	520.68970 -		CORN	LL

Table 6.7 MPSX Computer Output — Columns at Intermediate Level; RANGE Analysis, Mayaland Farm Example

MPSX / 370 MAYALAND FARM
SECTION 4 - COLUMNS AT INTERMEDIATE LEVEL

NUMBER	COLUMN	AT	ACTIVITY	INPUT COST	LOWER LIMIT UPPER LIMIT	LOWER ACTIVITY UPPER ACTIVITY	UNIT COST	LOWER COST UPPER COST	LIMITING PROCESS	AT
8	SORGHUM	BS	4.49135	1523.00000	.	9.48276 -	151.00000 -	1372.00000	CORN	LL
					NONE	5.00000	3351.00000 -	4874.00000	MARKET	UL
9	PEANUTS	BS	0.50864	4874.00000	.	INFINITY -	3351.00000 -	1523.00000	MARKET	UL
					NONE	0.50865	INFINITY -	INFINITY	NONE	

The additional printout generated by RANGE is organized in four sections. These are illustrated in Tables 6.4 to 6.7 using the Mayaland farm example.

6.3.1 Rows at Limit Level

This section (Table 6.4) reports on the stability of the constraints that are used up to their limit in the optimal solution. These are the resources or constraints that have nonzero shadow prices in the optimal solution. For the Mayaland farm these are the land and market constraints. Part of the table simply duplicates the information contained in the ROWS section of the optimal solution printout (Table 6.2). The new information of importance is reported in the column headed LOWER/UPPER ACTIVITY. This column shows the range in the b_i coefficient for each constraint over which the shadow price still holds—the two numbers reported for a row are the critical turning points for the solution basis. The activity levels will of course change as the supply of a limiting resource is changed, but the activities appearing in the solution basis will not change over this range.

In the case of the land, the shadow price of 1523 pesos per hectare would remain the same in the optimal solution if the land constraint were reduced from 5.0 to 0.508 hectares, or if it were increased to 8.403 hectares. This relatively wide range suggests that the optimal basis is robust with respect to land area. It also provides information of potential value to the farmer. For example, we could now advise our Mayaland farmer to pay up to 1523 pesos per hectare to rent an additional 3.403 hectares of land. However, if additional land renting were contemplated beyond 3.403 hectares, we could not advise on the rent to pay without solving a revised linear programming problem.

Similarly, the market constraint for peanuts has a shadow price of 3409 pesos per ton, and this shadow price holds over the range 0 to 4.915 tons. If the market constraint were increased to 4.915 tons the farmer could devote his entire 5 hectares to peanuts (see Table 2.2). This suggests that the farmer should attach high priority to ways of expanding his market for peanuts; each additional ton that he could sell would add 3409 pesos to his total gross margin.

The column headed LIMITING PROCESS provides information about the activity that would leave the solution basis at the critical turning point for each limiting resource. For example, if the available land were reduced to 0.508 hectares, then sorghum would be forced out of the optimal solution. On the other hand, if the available land were increased to 8.403 hectares, then labor would become a binding constraint and the slack labor activity would leave the basis.

If the market constraint for peanuts were reduced to zero, then peanuts would be forced out of the optimal basis. But if the market constraint were increased to 4.915 tons, sorghum would leave the optimal solution and the entire land area would be devoted to peanuts.

6.3.2 Columns at Limit Level

This section (Table 6.5) reports on the stability of those activities that appear at their limit levels in the optimal solution. The relevant activities are corn and beans which are at their lower limit level of zero.[2] We have already seen that these activities have nonzero REDUCED COSTS (Table 6.3), and these reduced costs are reported under the column headed UNIT COST in Table 6.5. They measure the amount by which an activity c_j coefficient would have to increase before the activity could enter the optimal solution. The new information reported in Table 6.5 pertains to the level at which an activity would enter the optimal solution if its c_j coefficient were increased by the required reduced cost. This information is to be found in the column headed UPPER ACTIVITY. For example, corn would enter the optimal solution if its gross margin were increased by 151 pesos to 1523 pesos per hectare (the UPPER COST), and it would enter the optimal solution at 4.491 hectares. Similarly beans would enter the optimal solution at 4.491 hectares if their gross margin were increased by 304 pesos to 1523 pesos per hectare.

The column headed LIMITING PROCESS shows that if corn or beans did enter the solution at 4.491 hectares then sorghum would be forced out of the solution to provide the necessary land. Since sorghum has a gross margin of 1523 pesos per hectare, this explains why the gross margin for corn or beans must increase to 1523 pesos per hectare if they are to enter the solution.

Table 6.5 also shows changes in activity levels corresponding to critical reductions in their gross margins. However, these are of little interest since the activity levels decline from zero to negative levels.

6.3.3 Rows at Intermediate Level

This section of the sensitivity analysis considers the constraints that were not used at capacity in the optimal solution (Table 6.6). These are the rows for labor and mules in the Mayaland farm example. In each case there is a slack capacity available so the shadow price is zero. The relevant information in Table 6.6 shows what happens if the b_i coefficients of these constraints are reduced below the level at which the resource is used in the optimal solution.

The optimal solution uses only 9.966 months of the available 16.5 months of labor. The LOWER ACTIVITY column in Table 6.6 shows that if the supply of labor were reduced to 7.72 months then a change in basis would result. With that amount of labor the LIMITING PROCESS column shows that corn, which requires least labor per hectare (Table 2.2), would enter the optimal basis. Corn would in fact exactly replace sorghum in the solution. Similarly the supply of mules could be reduced from 10.0 to 0.7375 months before a change in basis would occur. At 0.7375 months some land would become idle and hence the slack activity for land would enter the basis.

The UNIT COST entries in Table 6.6 indicate the amount of total gross margin foregone for each unit reduction in a resource below the level at which it is employed in the optimal solution. Each month reduction in labor below

9.966 months would cost 302 pesos, and the value holds for each month until labor has been reduced to 7.72 months. Similarly with mules, each month reduction below 5.948 months would cost 1312.93 pesos, and this value holds for each month reduction until there are only 0.7375 months of mules left.

6.3.4 Columns at Intermediate Level

This section (Table 6.7) reports on the stability of those activities included in the optimal solution at nonzero levels. The column headed LOWER/UPPER COST shows the range in activity gross margins for which the current basis remains optimal.

Sorghum has a gross margin of 1523 pesos per hectare and at that level 4.491 hectares enter the optimal solution. However, sorghum would remain in the basis even if its gross margin were reduced by 151 pesos (the entry in the UNIT COST column) to 1372 pesos (the LOWER COST entry). If the gross margin for sorghum fell below 1372 pesos per hectare then corn would become sufficiently attractive relative to sorghum to enter the basis (see the LIMITING PROCESS column). On the upside, the gross margin for sorghum could increase to 4874 pesos per hectare before a change in basis would occur. At that level of profitability, sorghum would displace peanuts in the optimal solution and the slack activity for the market constraint would enter the basis.

Table 6.7 also shows that peanuts are a stable crop. Their gross margin would have to fall from 4874 to 1523 pesos per hectare before they would leave the optimal basis (they would be replaced in the basis by the slack activity for the market constraint). On the other hand, the optimal basis remains unchanged for any sized increase in their gross margin as shown by the infinity entry in the UPPER COST column. This is because of the market constraint on peanuts.

Note that in all these results it is always assumed that only one coefficient is changed at a time from its initial value. RANGE cannot be used to analyze the stability of the solution with respect to simultaneous changes in more than one coefficient.

6.4 POSTOPTIMALITY ANALYSIS USING PARAMETRIC PROGRAMMING

Parametric programming is one of the most important methods of further analysis after an initial optimum solution is obtained. This option allows one to consider the impact of a sequence of incremental changes in any of the model coefficients. Any a_{ij}, b_i, or c_j can be varied systematically and changes in the optimal solution studied. The technique is useful for additional sensitivity analysis of an optimal solution where it is desired to explore the stability of the solution for values of coefficients lying beyond critical values identified in the MPSX RANGE analysis. Also, by varying c_j or b_i coefficients, it is a powerful way to derive product supply or resource demand functions from a farm or sector model. It is also a useful tool in risk analysis as we saw in Chapter 5.

There are five basic options available in the MPSX package:

PARAOBJ	Considers changes in one or more coefficients in the objective function, others held constant
PARARHS	Considers changes in one or more RHS value, all other coefficients held constant
PARARIM	Considers simultaneously changes for coefficients in both the objective function and the RHS
PARACOL	Considers changes in one or more a_{ij} coefficients for a specific activity
PARAROW	Considers changes in one or more a_{ij} coefficients for a specific constraint

The first two of these options are the ones most commonly used in applied economic analysis, and they form the subject of our present discussion.

6.4.1 Basis Changes or Fixed Intervals

There are two ways to approach problems using the parametric options. One is to establish a fixed interval of change for the coefficient parameterized and then determine the optimal solution for successive values of that coefficient. This is the fixed interval option. For example, the amount of land available to the Mayaland farmer is 5.0 hectares. One could look at the optimal solutions obtained as the available land increases by intervals of 0.1 hectares. In this case the computer would solve and print for the initial supply of 5.0 hectares, and then successively provide solutions for 5.1 hectares, 5.2 hectares, and so on to an upper limit of say 10.0 hectares, all other coefficients held constant at their original values.

The second option is to instruct the computer to print solutions only at basis changes when a coefficient is parameterized. Using the example just considered, the available land is initially set at 5.0 hectares. After printing the initial optimal solution, the program prints a solution only when a change in basis occurs. That is, when the increase in the available land induces a new activity to enter the basis. A range of land areas is specified initially and the procedure stops when the maximum land area has been reached.

It is this second alternative which is of greatest value in economic analysis using linear programming. A particularly useful property of such parametrically derived solutions is that the solution corresponding to any noncritical value of the parameterized coefficient can always be derived by linear interpolation between adjacent basic solutions. Thus the set of basic solutions provides complete information about the optimal solution corresponding to any value of the coefficient within the prescribed range. With the fixed interval approach however, solutions corresponding to basis changes are only captured by chance, and linear interpolations between adjacent solutions may prove misleading. In the examples that follow the suggested program calls for printing solutions at basis changes only.

6.4.2 An Example Using PARAOBJ

Suppose that in the Mayaland farm example it is desired to know the supply function of sorghum for the farm over a range of possible price changes. To obtain this supply function we must systematically vary the price of sorghum over the relevant range of prices.

The initial gross margin for sorghum (Table 2.2) is 1523 pesos per hectare. Suppose this gross margin is derived as follows:

Yield of sorghum = 1.00 ton/hectare
Price of sorghum = 1760 pesos/ton
Revenue = (1.0)(1760) = 1760 pesos/hectare
Direct production costs = 237 pesos/hectare
Gross margin = 1760 − 237 = 1523 pesos/hectare

Suppose also that the range of prices of interest is 1000 to 5000 pesos per ton. If we substitute this latter price into the gross margin calculation, then the range of gross margins we need to consider is 763 to 4763 pesos per hectare.

To obtain the required solutions for the Mayaland farm using the basis change approach, the program and data cards are set up as shown in Table 6.8. If one compares this set of cards with those in Table 6.1 for the standard linear programming problem, many of them are the same. The new program and data cards and their function are described below.

Card	*Comment*
`MOVE(XCHROW,'CHSOR')`	This card identifies the name of a change row; in this case CHSOR. The name is optional and may consist of up to 8 characters. The chosen name must also be used on cards now added to, or changed, in the DATA deck. For this example a total of three changes are introduced into the data deck. In the ROWS section following the objective function OBJ, a new card is introduced: N CHSOR. In the columns section, the coefficient for sorghum in the OBJ row is changed to 763.0. This is the new initial gross margin. In addition, a unit coefficient is entered for sorghum in the CHSOR

	row. The positive sign on this coefficient indicates that the gross margin for sorghum is to be parameterized over increasing values. A −1.0 coefficient would lead to the gross margin being successively reduced.
`XPARDELT=0.0`	This card fixes the interval at which solutions are to be printed. It is always set equal to 0.0 when solutions are required at basis changes only.
`XPARMAX=4000.0`	This card defines the maximum increase to be considered for the coefficient being parameterized. In our example we wish the gross margin for sorghum to be increased from 763 pesos to 4763 pesos, or a total increment of 4000 pesos.
`XPARAM=0.0`	This card indicates that parameterization should start at the initial value for the coefficient.
`XFREQ1=1`	This card alerts the computer to carry out the command that follows at intervals of one (for each change in basis).
`MVADR(XDOFREQ1,PRINTSOL)`	This is the key program card for printing solutions at basis changes.
`PARAOBJ`	This is the program card calling for the set of parametric options to be performed using all the previous commands.
`PRINTSOL SOLUTION`	This calls for printing each solution following the PARAOBJ option.

6.4.3 Interpreting the Computer Output for PARAOBJ

The output for parametric programming has the same format as discussed for the standard linear programming run in Section 6.2. The first solution printed is the optimal one for the initial set of coefficients in the objective function. Note though that in the example we have reduced the initial gross margin for sorghum to 763 pesos per hectare. A sequence of optimal solutions are then printed, each corresponding to a basis change as the gross margin for sorghum is increased to the next critical value. The value of the gross margin for sorghum at which the basis change occurs is also indicated. The last solution printed is always the solution for the final value established (PARMAX) for the parameterized coefficient.

Table 6.8 Mayaland Example of Use of PARAOBJ Program and Data Cards for MPSX

```
(JCL cards)
          PROGRAM
          INITIALZ
          TITLE('MAYALAND CHANGING SORGHUM PRICE')
          MOVE(XDATA,'MAYALAND')
          MOVE(XPBNAME,'PROB2')
          MOVE(XOBJ,'OBJ')
          MOVE(XRHS,'B1')
          MOVE(XCHROW,'CHSOR')
          XPARDELT=0.0
          XPARMAX=4000.0
          XPARAM=0.0
          CONVERT('SUMMARY')
          SETUP('MAX')
          PRIMAL
          SOLUTION
          FREQ1=1
          MVADR(XDOFREQ1,PRINTSOL)
          PARAOBJ
PRINTSOL  SOLUTION
          EXIT
          PEND
(JCL cards)
NAME           MAYALAND
ROWS
 N   OBJ
 N   CHSOR
 L   LAND
 L   LABOR
 L   MULES
 L   MARKET
COLUMNS
    CORN      OBJ         1372.        LAND          1.
    CORN      LABOR          1.42      MULES         1.45
    BEANS     OBJ         1219.        LAND          1.
    BEANS     LABOR          1.87      MULES         1.27
    SORGHUM   OBJ          763.        CHSOR         1.
    SORCHUM   LAND           1.        LABOR         1.92
    SORGHUM   MULES          1.16
    PEANUTS   OBJ         4874.        LAND          1.
    PEANUTS   LABOR          2.64      MULES         1.45
    PEANUTS   MARKET         0.983
RHS
    B1        LAND           5.        LABOR        16.5
    B1        MULES         10.        MARKET         .5
ENDATA
(JCL cards)
```

Table 6.9 Basic Solutions for Changes in the Price of Sorghum, Mayaland Farm Example

	Initial solution	First basis change	Final solution
Price of sorghum (pesos / ton)	1000	1609	5000
Gross margin for sorghum (pesos / ha)	763	1372	4763
Total gross margin (pesos)	8641.28	8641.28	23871.46
Crop areas (ha)			
Corn	4.491	—	—
Beans	—	—	—
Sorghum	—	4.491	4.491
Peanuts	0.509	0.509	0.509
Slack activities			
Land (ha)	—	—	—
Labor (months)	8.779	6.534	6.534
Mules (months)	2.750	4.052	4.052
Market (tons)	—	—	—

Table 6.9 shows the sequence of solutions obtained for the Mayaland example in Table 6.8. In this case there is only one basis change as the sorghum price is parameterized from 1000 to 5000 pesos per ton. This change in basis occurs when the gross margin of sorghum is 1372 pesos per hectare; equivalent to a sorghum price of 1609 pesos per ton. At that price sorghum displaces corn from the solution. Since there are no further basis changes, the last solution printed is the same as the previous one.

6.4.4 An Example Using PARARHS

Instead of parameterizing one or more coefficients in the objective function, it is sometimes useful to study the impact of changing the availability of one or two key resources or restrictions listed among the RHS. For example, in the Mayaland example, it may be desired to study the impact on the optimal farm plan of changes in the amount of land available.

The procedure to institute parametric programming on a key RHS coefficient is analogous to that for PARAOBJ, but now a change column is introduced instead of a change row. A new card for the change column is added in the RHS section of the input deck. The name of the change column may be described using up to eight characters, but it must conform exactly with the name used in the program deck.

The new data card to change the available land in the Mayaland example might read as:

CHLAND LAND 1.0

where CHLAND is the name of the change column and is punched starting in

Table 6.10 Mayaland Example of Use of PARARHS Program Cards for MPSX

```
         PROGRAM
         INITIALZ
         TITLE('MAYALAND CHANGING LAND RHS')
         MOVE(XDATA,'MAYALAND')
         MOVE(XPNAME,'PROB3')
         MOVE(XOBJ,'OBJ')
         MOVE(XRHS,'B1')
         MOVE(XCHCOL,'CHLAND')
         XPARDELT=0.0
         XPARMAX=5.0
         XPARAM=0.0
         CONVERT('SUMMARY')
         SETUP('MAX')
         PRIMAL
         SOLUTION
         XFREQ1=1
         MVADR(XDOFREQ1,PRINTSOL)
         PARARHS
PRINTSOL SOLUTION
         EXIT
         PEND
```

column 5. This new card is inserted in the RHS section of the deck immediately after all the B1 cards.

The program cards for the Mayaland example are given in Table 6.10. This program parameterizes the available land from its initial value of 5.0 hectares to a maximum of 10.0 hectares. The program also calls for solutions to be printed at each change in basis.

NOTES

1 IBM publishes its own manuals describing how to use the MPSX package.
2 Activities may enter at nonzero limit levels when the MPSX option BOUNDS is used.