NATURAL ENVIRONMENT RESEARCH COUNCIL

INSTITUTE of HYDROLOGY

REPORT No 24 August 1974

CONCEPTUAL MODELLING IN HYDROLOGY

by

J R DOUGLAS

INSTITUTE OF HYDROLOGY
HOWBERY PARK
WALLINGFORD
BERKSHIRE

CONTENTS

FIGURES

TABLES

INSTITUTE OF HYDROLOGY

Report No 24 May 1974

CONCEPTUAL MODELLING IN HYDROLOGY

by

J. R. Douglas

ABSTRACT

This report has been written in three parts. The first part describes a programming package which has been developed to assist the system modeller examine and improve the performance of his model. The second part of the report describes a specific model, developed at the Institute of Hydrology to predict the flow hydrograph for any of the Institute's research catchments, given precipitation and other meteorological inputs. The third and final section of the report presents results of some applications of the model previously described on a number of British catchments. Ways in which this model might be used to solve various hydrological problems are suggested. Computer program listings are included as appendices to the report.

PART 1 A COMPUTER PROGRAM PACKAGE FOR DEVELOPING SYSTEM MODELS

1.1 An introduction to modelling

For many years scientists have been using mathematical modelling as a tool for increasing their knowledge of physical systems. A system is described as closely as possible by a series of equations, such that an applied input generates a system response, or output. If the model is a good representation of the physical system, the output from the model will approximate closely to the output from the physical system for the same input. However, a model always involves some simplification, so that it is only partially representative of the workings of the physical system.

The hydrological cycle is one such system. It can be represented in terms of a flow diagram (such as that shown in Figure 1.1) which itself can be expressed as a set of equations governing sizes of storage and fluxes between them. In many hydrological studies, however, such as a study of the generation of streamflow from precipitation, only parts of the whole system need be considered, while studies of particular hydrological processes may require treatment in much greater detail. This progression from an original, very simple representation of the system, to a considerable amount of detail demanded by specific problems is familiar to all modellers.

This report describes a computer program which helps the model builder examine the workings of his model, to estimate and improve the efficiency of its performance. Perhaps the most fundamental criterion is that it should be possible to judge performance by comparing the outputs generated by the model with those produced by the physical system in response to the identical set of inputs. The error of prediction should not only be observed, but should also be expressed numerically in some way, so that any improvement in model performance can be seen as a reduction in the measured error.

The computer package cannot itself make any changes to the structure of the model to improve the efficiency of prediction. It is, therefore, important that the modeller should use his experience to produce a satisfactory major structure of the model from the beginning. It is also important that he should be able to recognise when the structure of his model is inadequate. However, within this framework the modeller can give himself a considerable amount of flexibility. He can do this by introducing any constant within his model as a parameter. These parameters may represent quantities such as storage sizes or limiting temperatures. Then, if within the model structure, a segment of the model is not required in a particular application, it can be eliminated by manipulation of appropriate parameter values. For example, a storage element might be removed by setting its size to zero. Although functional relationships within the structure may be expressed very generally, it is still necessary to give the form of the relationship.

Using the computer package, a number of operations can be performed on the model parameters. It is possible to find the optimum set of values of the parameters; that is, the set which, in combination, produces the smallest error of estimate of the output data sequence. The package also contains facilities allowing the effects of specified sets of parameter values to be examined, and allows investigation of the influence that the value of one parameter might have on the value of another.

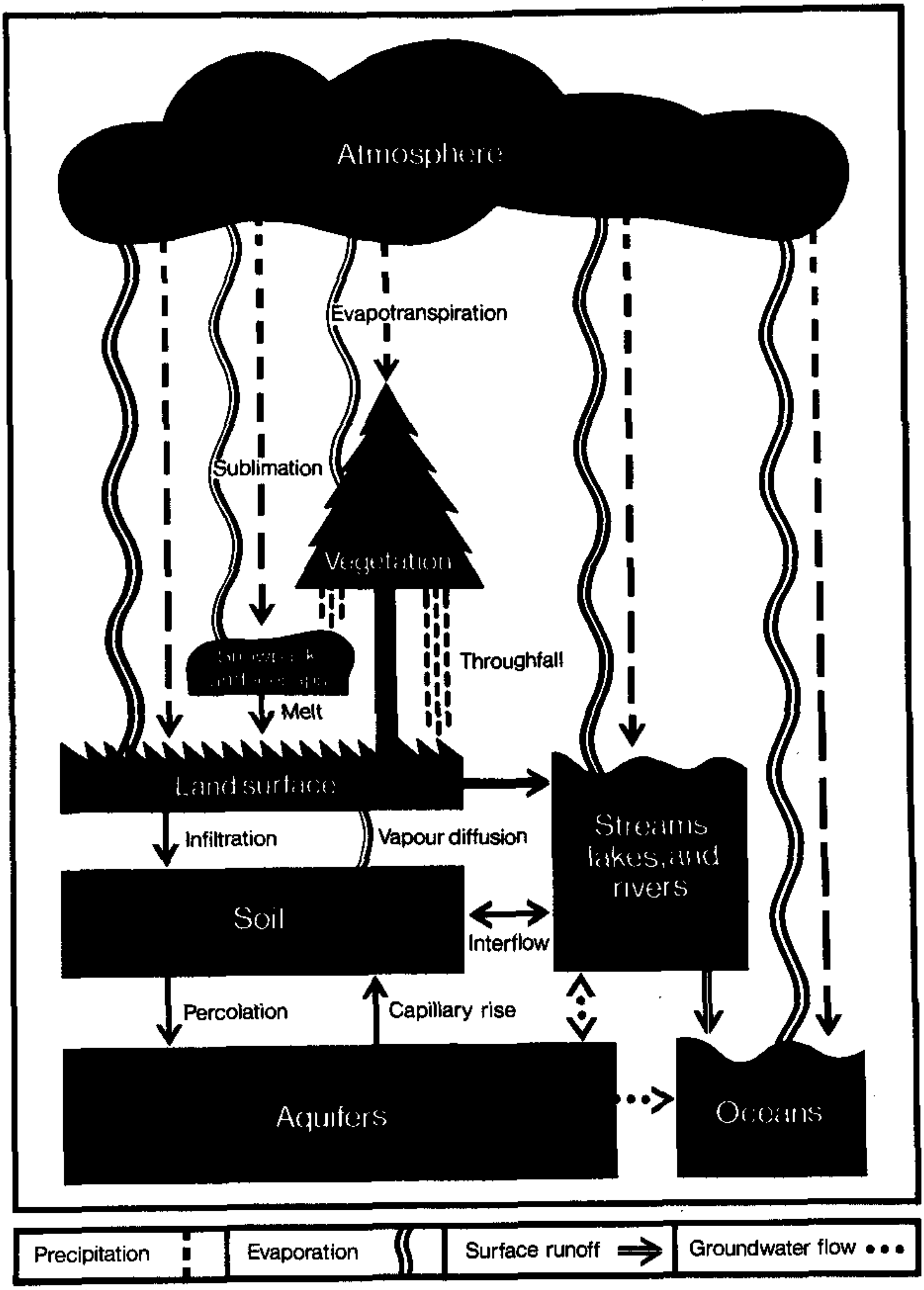


Figure 1.1 The hydrological cycle

1.2 Definition and calculation of error function

A fundamental concept of this modelling package is the definition of the error function. This is the term which calculates the error of a particular model run. We define a model run to be the process whereby an observed sequence of 'input'data (which may be precipitation data, evapotranspiration data or both) is 'passed through' the model for a particular combination of model parameters, and thereby transformed into an 'output' sequence (which for example may be a streamflow or a soil moisture sequence). This output sequence may be compared with the observed output sequence to obtain some measure of the error of estimate of each value in the output sequence. The error of a model run may then be calculated from the errors associated with the individual readings in the output sequence.

To illustrate this and subsequent points, a simple hydrological model is presented. This purports to predict daily streamflow, by budgetting daily rainfall, evaporation and soil moisture deficit, (SMD). It is suggested that daily flow is given by a proportion,

$$\exp^{-0.05 \times SMD}$$

of the daily rainfall. The current SMD is calculated by budgetting all inputs and outputs as each day's data are presented. Even with such a simple model there is a problem in defining the error function, the measure of error and model performance. Three basic choices are available: to compare observed flows with those predicted by the model; to compare observed soil moisture deficits with deficits predicted by the model; or to make both these comparisons. This choice is made simpler by an examination of the purpose of the model. If the model is solely concerned with predicting streamflows, then perhaps the quality of the predicted flows alone needs examination. If the model is intended to predict the state of soil moisture storage, it is the SMDs which should provide a measure of modelling efficiency.

Having decided what to compare, there is the major problem of how to quantify the comparison. The error of estimate must be related to the difference between observed and predicted outputs: whether it is best given by the absolute differences, the squares of the differences, logs of differences or some other measure depends again on the problem.

A measure of error frequently used in hydrological modelling is the sum of squares of differences between observed and predicted outputs, such that the error for each data point is given by

$$ERROR_t = (\text{observed output}_t - \text{predicted output}_t)^2$$

and the error, F, for the model run by

$$F = \sum_{t=1}^{n} ERROR_t$$

Table 1.1 A Simple Hydrological Model

Day	Rain R (mm)	Evaporation E (mm)	Predicted Flow, Qp* (mm)	Observed Flow, Qo (mm)	Predicted S.M.D., Dp* (mm)	Observed S.M.D., Do (mm)
0					10.0	10.0
1	0.0	4.2	0.0	0.2	14.2	
2	25.8	2.1	12.7	12.0	3.7	
3	0.0	5.2	0.0	1.5	8.9	
4	0.0	4.8	0.0	0.6	13.7	18.0
5	6.0	3.1	3.0	2.8	13.8	
6	20.6	2.7	10.3	12.4	6.2	
7	50.3	0.2	36.9	30.5	- 7.0	
8	10.6	3.4	15.0	18.7	0.8	2.0
9	0.0	5.2	0.0	3.2	6.0	
10	0.0	4.6	0.0	1.1	10.6	
11	0.0	4.4	0.0	0.6	15.0	
12	0.0	5.8	0.0	0.2	20.8	
13	0.0	5.2	0.0	0.1	26.0	
14	10.4	2.9	2.8	2.9	21.3	24.0

* $Qp_t = R_t \times e^{-0.05} \times Dp_{t-1}$ and $Dp_t = Dp_{t-1} - R_t + E_t + Qp_t$

Model Performance

	(a) Flow prediction	(b) Soil moisture prediction
Mean	$\bar{Q}o = \frac{\sum_{t=1}^{14} (Qo_t)}{14} = 6.2$ mm	$\bar{D}o = \frac{\sum_{n=1}^{4} (Do_n)}{4} = 12.25$ mm
Sum of squares	$Fo_Q = \sum_{t=1}^{14} (Qo_t - \bar{Q}o)^2 = 1070$ mm^2	$Fo_D = \sum_{n=1}^{4} (Do_n - \bar{D}_o)^2 = 248$ mm^2
Model Error	$F_Q = \sum_{t=1}^{14} (Qp_t - Qo_t)^2 = 74$ mm^2	$F_D = \sum_{n=1}^{4} (Dp_n - Do_n)^2 = 9.2$ mm^2
Efficiency	$RE_Q = 100 \frac{Fo_Q - F_Q}{Fo_Q} = 93.1\%$	$RE_D = 100 \frac{Fo_D - F_D}{Fo_D} = 96.3\%$

This measure is analogous with the least squares criterion used commonly in linear statistics. One of its features is that it places greater emphasis on individual large errors than on a series of relatively small ones. This is a benefit if estimates of peak flows are required, but may be undesirable if, for example, a model is required only to predict baseflow in a groundwater-fed stream. It might, in this latter case, be appropriate to attach equal weight to equal percentage errors throughout the range of outflow rates.

The problem of definition of the error function is clearly somewhat subjective and is left to the user. Some points have been outlined above, but more comprehensive reviews of the subject have been made by Aitken (1973) and Clarke (1973). The only requirement of this modelling package in this respect is that each model run calculates the value of an error function, F.

1.3 Efficiency of a model run

The magnitude of the model error, F, is in itself, not a complete statement of error.

Its magnitude is dependent on the goodness of fit of the model, but also on the magnitude and variation of the observed output data. Nevertheless when obtained from two model runs on the same set of data, the run giving the smaller value of F is the run with the better fit. To express the error estimate in a form allowing comparison between sets of data, the error can be regarded as that part of the sum of squares Fo, of the observed output data, which is explained by the model, where;

$$Fo = \sum_{t=1}^{n} \left(OBS.OUTPUT_t - \frac{\sum_{t=1}^{n} OBS.OUTPUT_t}{n}\right)^2$$

When the sum of squares error criteria is adopted, the modelling efficiency can be expressed as the percentage of the initial sum of squares explained by the model, such that the efficiency, RE, is given by

$$RE = 100 \frac{(Fo - F)}{Fo} \%$$

The value of this index can range from -∞ up to + 100%. A negative value indicates that the model produces a worse estimate of the output than simply using the mean output. An efficiency of 100% indicates that there is no error, and that the output computed by the model is then exactly equal to that observed.

Comparisons of efficiency are, however, not always very meaningful, because RE values still tend to reflect the type of data being modelled, as well as the performance of the model. This problem revolves around the amount of data being presented which is normally expressed in years, or months. A more meaningful measure of the amount of data might be the number of peaks or events being fitted. In a hydrological system, this might vary from one per year in a spring snowmelt or glacier fed

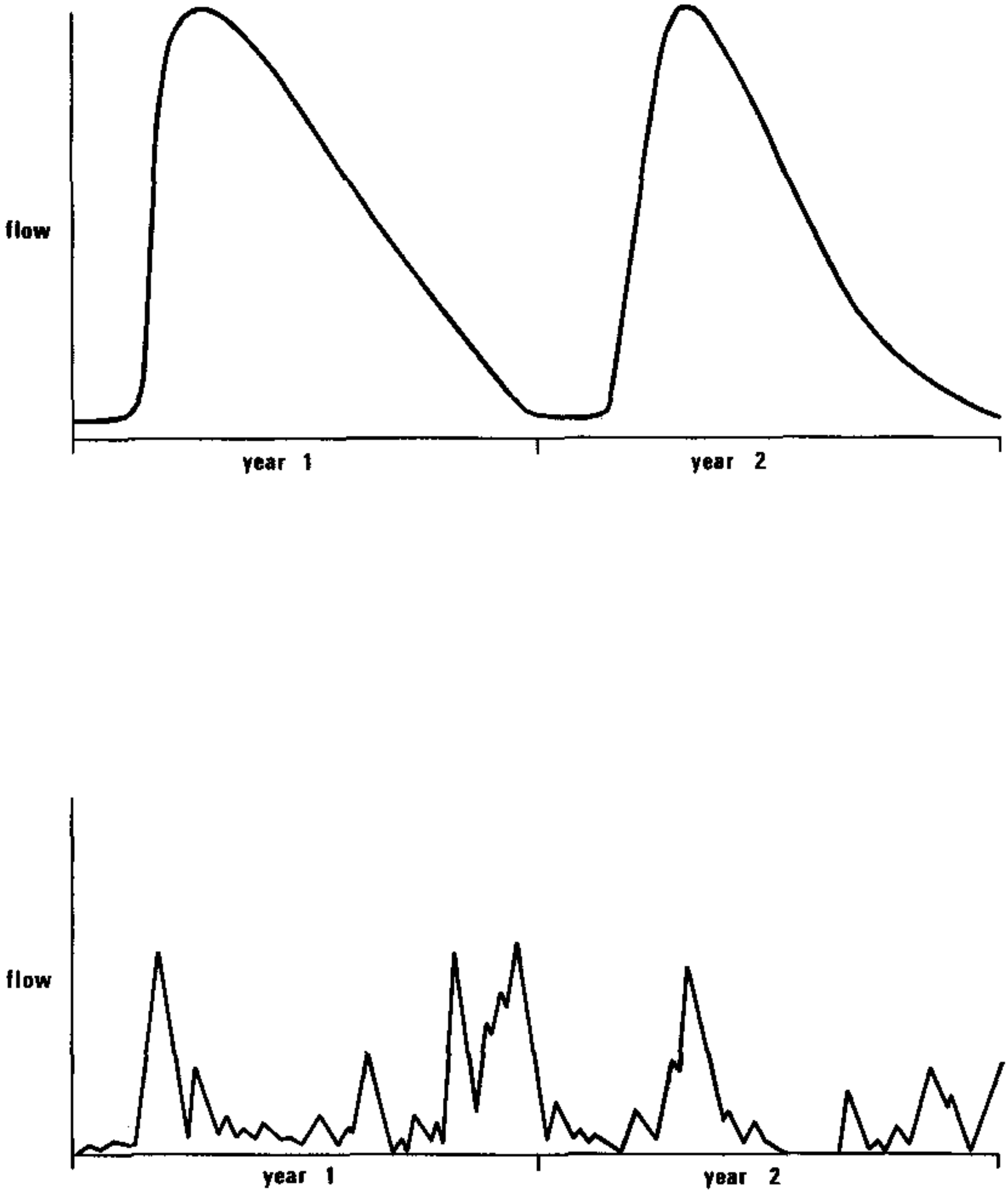


Figure 1.2 Contrasting patterns of annual streamflow

catchment, to upwards of 100 in a small, mountainous catchment in a temperate climate. Figure 1.2 illustrates the two cases, and shows that the initial variance of the two data sets might be very similar. It would appear that 90% explained variance would be much more commendable in the second case than in the first.

1.4 Model parameters and parameter optimisation

Within the equations and logic of a model, there are usually unknown parameters which must be estimated by fitting the model to the data. The example shown earlier had only one parameter, ie the 0.05 in the equation

$$FLOW_t = RAIN_t \times \exp^{-0.05 \times SMD_{t-1}}$$

It is clear that if this multiplication factor had been given a different value, the predicted flows would have differed and there would have been a consequent change in the error for each data point, and in the total error for the run. By testing different values of this parameter, it would be possible to find the value which gave the lowest total error, ie the minimum value of F. This is illustrated in Figure 1.3, where the error resulting from each run of the model is plotted against the value of the multiplier.

If the error of a model run is dependent on the values of more than one parameter, the problem of finding the best combination of values becomes increasingly difficult. The problem is illustrated in Figure 1.4, where the error resulting from running a model with various combinations of values of two parameters are plotted. Contours have been drawn linking points with the same error, producing the error function surface for these two parameters.

A similar picture could be visualised in 3 dimensions, representing the error produced by running the model with different combinations of values of 3 parameters. With more than 3 parameters, it becomes difficult to visualise the error function surface although there is no limit to the number of parameters whose multi-dimensional surface can be studied mathematically. With one or two parameters (for which the error function surface could be expressed graphically), it is a simple job to find the combination of parameter values which give the smallest error - the optimum parameter set. With more parameters, trial and error methods of finding the optimum, even by the calculation of errors from a regular grid system of parameter values, becomes very inefficient. Mathematical techniques are available for locating the optimum parameter set, and one such technique is included in this model program package. The technique chosen is that described by Rosenbrock (1960) and recommended for use with hydrological models by Ibbitt (1972).

Briefly, the modeller specifies a starting set of parameter values, and the ranges of values within which each parameter is allowed to vary. The program takes each parameter in turn, holding all the remainder constant while finding the best value of the first parameter (that is the value of the parameter which produces the lowest total error in a

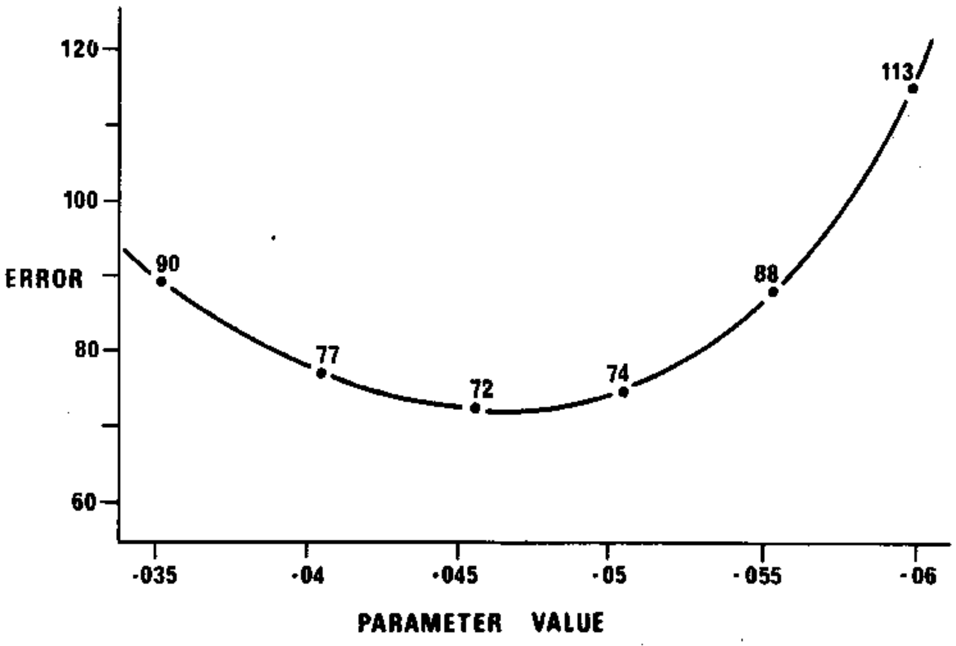


Figure 1.3 The optimum value of one parameter

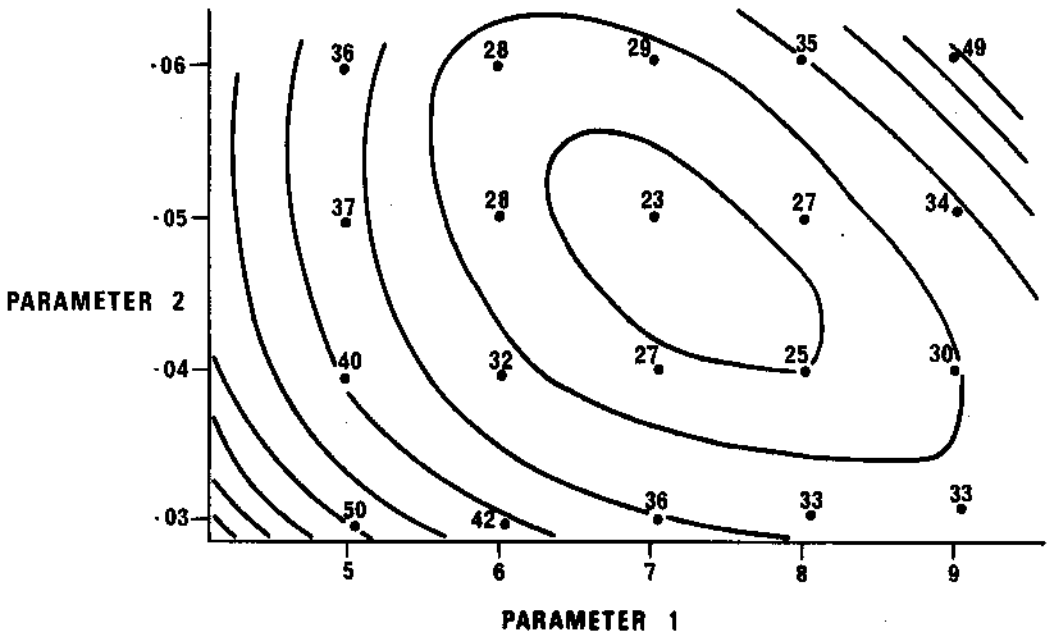
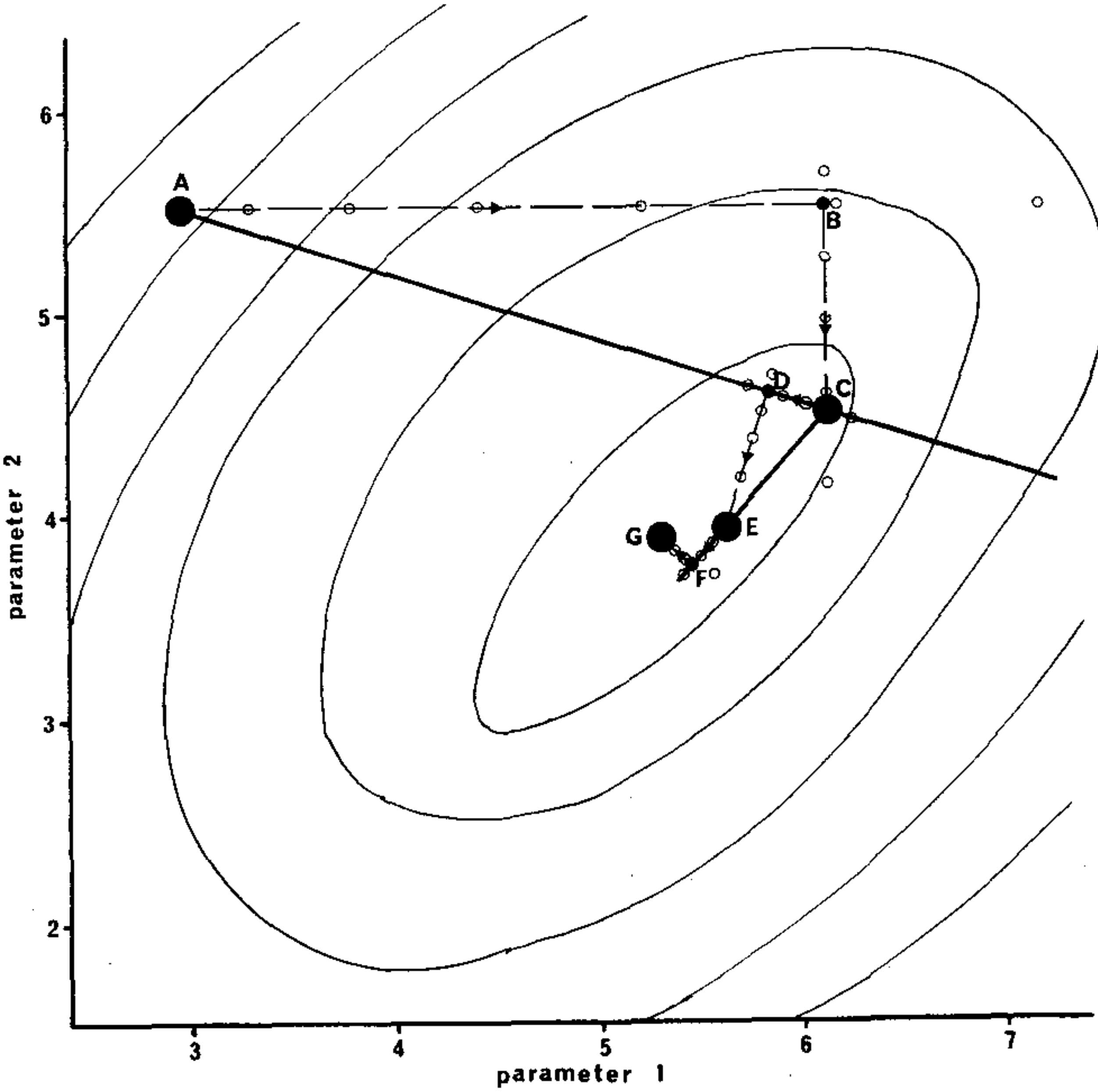


Figure 1.4 Error function surface for two parameters



A – starting point; P1 = 3, P2 = 5.5

C – end of iteration 1 ; P1 = 6.3, P2 = 4.4

E – - - - - - - - - - - 2 ; P1 = 5.6, P2 = 3.9

G – - - - - - - - - - - 3 ; P1 = 5.3, P2 = 3.9

—— ➤—— —— direction of search

———————— new axial directions

Figure 1.5 Locating the optimum values of two parameters model run). It then holds the first parameter at this value and all other parameters at their starting values, except the second parameter, for which a best value is found. This procedure is repeated for each parameter: when completed, this is the end of an iteration of search.

From its experience in the first iteration, the program defines the best direction of search as a line joining the starting point to the point reached at the end of the iteration. It uses this direction at the first axis of search in the second iteration, subsequent axes being defined orthoganally to this. At the end of each iteration, this re-orientation of axes is made, so that the search is always made in the most likely direction. The technique is illustrated for the two parameter case for 3 iterations in Figure 1.5. Although the error surface contours have been drawn on the figure, it should be remembered that the shape of the surface will not be known before using the optimisation program. The program finds the optimum by calculating the model error for given combinations of parameter values, selecting the combination of values by experience gained from previous combination in the manner described above.

1.5 Problems of parameter optimisation

When a model has only one parameter, it is clear that it will not be difficult to find the value of this parameter for which the model most efficiently predicts the output sequence. The only condition which would render this impossible would occur if the output sequence was unaffected by the parameter, so that a change in the value of the parameter would not cause any change in the output sequence predicted by the model. When there are two or more parameters, there may be an additional problem: that of interdependence of parameters. This problem has been discussed by Plinston (1971). Briefly the problem can arise when the output sequence is similarly influenced by either of two parameters. An example is shown in Figure 1.6, where the relationship between output and SMD is given by the two parameter relationship.

$$\text{OUTPUT}_t = \left(1 - \frac{a \times \text{SMD}_{t-1}}{b}\right) \times \text{RAIN}_t$$

The error function surface associated with this model might be that shown in the figure. There is no unique optimum parameter set, as the model is equally efficient with a = 1 and b = 100 as with a = 50 and b = 5,000. An automatic optimisation might locate any point along the valley, depending only on the point at which the optimisation was started.

It could be argued, of course, that this interdependence is not a problem, since any of the pairs of values on the valley is an optimum and the resulting output sequence is none the worse for the interdependence. However, if any meaning is to be attached to individual parameter values - if, say, parameter values are to be correlated with catchment characteristics - the values obtained from such an optimisation would be meaningless.

The above example is extreme, and could be avoided by expressing a/b as a single parameter, or by fixing a, say, at a = 1. The more usual case, illustrated by Figure 1.7, is interdependence which hinders optimisation,

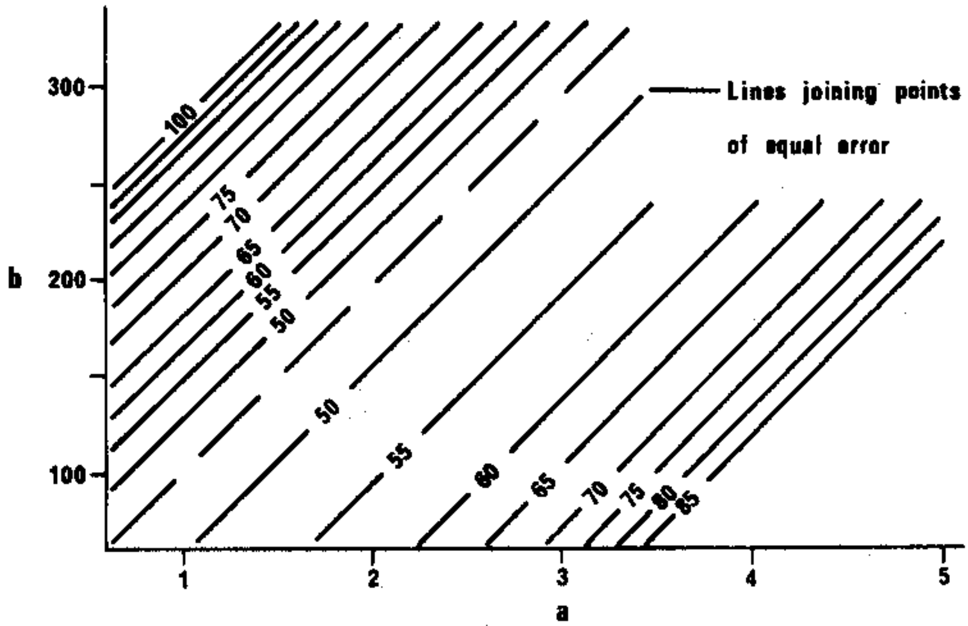


Figure 1.6 Two interdependent parameters

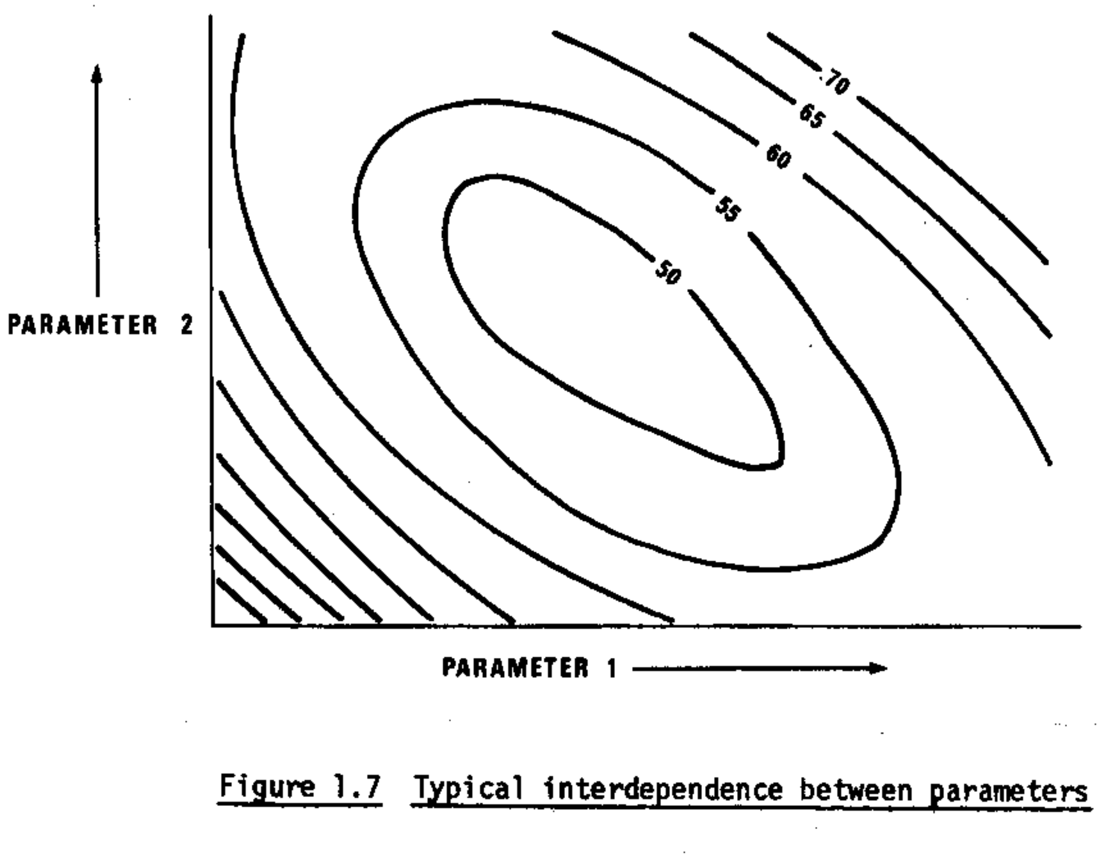


Figure 1.7 Typical interdependence between parameters

rather than completely preventing it. If the hindrance is severe, one of the same two tactics might be used to overcome it. Since in any well planned model, two parameters will not be functionally related to one another, it would be most common to fix the value of one parameter, and then find the optimum value of the other.

To assist the modeller the program package contains a section which calculates the model error for a grid of points on a parameter plane. Using this, a contour map, such as Figure 1.7 can be drawn up and suitable values chosen. Whilst this choice is bound to be somewhat subjective, if the parameters concerned relate to some physical process, it may be possible to measure the value, or to find likely values in the available literature.

Having located the best parameter set, and the optimum in the multi-dimensional error function space, it is possible to quantify the inter-dependence between parameters. Plinston's method gives a range of inter-dependence from 0, for totally independent parameters, to 1 for functionally related parameters.

The extent to which interdependent parameters can be tolerated in a model depends on the philosophy of the modeller. In most models of complete systems, including those representing the hydrological cycle, inter-dependence is unavoidable to some degree. This does not necessarily invalidate the model. A study of such interdependence does, however, provide an alternative means by which the model structure or sub-structure can be examined. For instance, if the results of fitting the model indicate that two parameters thought to be independent are in fact inter-dependent, then some structural change could be necessary.

1.6 The model as a predictor

The model can be fitted to a data sequence by the optimisation and surface mapping techniques described above. However, the real test of whether the model adequately reproduces system behaviour comes with applying the model to a sequence of data from the same system, not used in the calibration process. If the system is unchanged, and the model representative of it, then model performance on this independent data should approach that on the data used for calibration. Furthermore, if parameter optimisation is performed on the independent data batch, there should be little tendency for parameter values to depart significantly from the optimum located using the calibration data. If there is such a tendency, the reason for it should be ascertained. Possible explanations may be that:

- (i) two or more parameters are moving along an axis of inter-dependence;
- (ii) a parameter inadequately utilised in the calibration period suddenly becomes more critical in the independent data;
- (iii) there is a change in characteristics of the system.

In the first case, the differences, after being investigated by surface mapping, can probably be ignored, or dealt with in the manner described in Section 1.5. The second case implies that not enough data were used to calibrate the model. The third case could indicate that systematic errors are present in the data for one or other period: alternatively that there has been a significant change in the system. If none of these reasons for parameter variation is indicated, then one must suspect that the model is not a good representation of the system. Although it is flexible enough to produce reasonable output during calibration, it will only maintain its efficiency by constant re-calibration, and is, therefore, useless as a prediction tool.

1.7 The computer package

The computer package which performs the functions outlined above has been written in the FORTRAN language and used on ICL 1900 series computers. However, as far as was possible, it was written in standard Fortran, and should require only very minor modification to run on other types of computer. The following description of the computer program assumes some knowledge of the Fortran language. The program has a dummy MAIN routine, whose sole purpose is to call in the remainder of the package.

1.7.1 SUBROUTINE CONTROL, as its name implies, controls the program run. It first reads in three cards, which give details of the model being used, and the data to be used to model the system. The contents of these, and subsequent control cards, are detailed in the program listing and in Table 1.2. They are also displayed as a heading for each run of the modelling package. A card is then read containing indices which control the mode of operation of the model, the input and output, and so on.

A final control card is read, which gives information about the data to be modelled, then subroutine OPTION is called, to take charge of the manipulation of the model.

At the end of each operation, whether it be an optimisation run, an interdependence check or whatever, control passes back to the beginning of subroutine CONTROL, and another operation can be started. The program stops if it has reached the end of its input data cards, with the message 'MODEL RUN COMPLETED'.

1.7.2 SUBROUTINE OPTION contains the Fortran coding for the optimisation, model prediction, parameter interdependence, surface mapping and initial data variance modes of operation. Before starting one of these options, parameter values are read according to the index II (3) (see Table 1.2) in the format outlined in Table 1.3, and except when II(4) = 0, they are displayed on the lineprinter. Control then passes to the appropriate section of the subroutine according to the index II(1).

(a) Optimisation. No further data cards are required to complete an optimisation run. The technique described in Section 1.4 and Figure 1.4 is used. Before each call to SUBROUTINE MODEL current values of each parameter are assigned, within the range specified by the parameter cards, and using experience gained from previous

Table 1.2 Control cards for modelling package

Card	Symbol	Format	Columns	Comments
1	JJ(1)-(5)	5A4	1-20	Description of model
2	JJ(6)-(10)	5A4	1-20	Description of data source (catchment or location)
3	JJ(11)-(15)	5A4	1-20	Describes the duration of the data
4	II(1)	I1	4	Mode of operation: value is 1 for optimisation 2 for prediction 3 for interdependence 4 for surface mapping 5 for initial variance
	II(2)	I1	8	Controls reading of data: value 0 causes no data to be read 1 data is read
	II(3)	I1	12	Controls reading of parameter cards: value 1 reads new parameter cards 2 uses previously optimised values 3 no active parameters
	II(4)	I1	16	Controls details of lineprinter output: value 0 summary, 1 or 2 for more detail
	II(5)	I1	20	Controls graphical output: value 0 no graphs, otherwise up to user
	II(6)	I4	21-24	For optimisation only - maximum number of iterations
	II(7)	I4	28	Controls choice of error function. Value up to user
5	MM(1)	I4	1-4	Frequency (readings/day) of data source
	MM(2)	I4	5-8	Frequency (readings/day) of modelling
	MM(3)	I4	9-12	Julian day number at start of modelling
	MM(4)	I4	13-16	Number of months to be modelled
	MM(5)	I4	17-20	Number of first month to be modelled, relative to the start of the data file
	ANY OTHER CARDS REQUIRED BY REMAINDER OF PROGRAM RESUME AT CARD 1 ABOVE			

Table 1.3 Format of cards assigning values to model parameters

Card	Contents	Format	Columns	Comments
1	N	I4	1-4	Number of parameters
	NN	I4	5-8	Number of active parameters
then, for each parameter, a card containing:				
2 etc	NAMES	A8	1-8	Name of parameter as used in MODEL
	KK	I2	9-10	Order number for active parameters only
	YI	F10.0	11-20	Starting value of parameter
	B	F10.0	21-30	Optimisation runs only. Lower limit of range within which parameter may move
	C	F10.0	31-40	Optimisation runs only. Upper limit of range

N.B. Parameter cards must be presented in the order in which they are assigned in SUBROUTINE MODEL. The parameter on the first card should be assigned to YI(1), that on the second card to YI(2) and so on.

parameter sets. SUBROUTINE AUG is used in the conversion of previous experience into new parameter estimates.

Output from the model depends on the value of the index II(4), but the only output from OPTION is produced whatever the value of II(4): this summarises each iteration of optimisation, giving the number of runs through MODEL, and the lowest modelling error up to this point. If any iteration fails to show an improvement over the preceding one, the optimisation is terminated, and the best parameter set up to this point is assumed to be the optimum set.

Otherwise optimisation continues until either the maximum number of iterations is completed, or until no parameter changes more than 0.1% of its given range from the end of one iteration to the end of the next.

At the end of optimisation, the 'end of optimisation' indicator, IFIN, is given the value 2, and MODEL is called a final time with the best parameter set found. At this time, details of the performance of the model can be checked at its supposed optimum. Graphical output and any additional information required by the user can be obtained.

(b) Model Prediction

The model is run with the index, IFIN set at 2 (as at the end of optimisation) and details of the performance with the set of parameter values given is checked, graphically if required. Parameter values may either be included in the model, or as starting values on parameter cards.

(c) Parameter interdependence calculations

Interdependence is calculated between pairs of parameters, although it is permissible to submit 3 parameters, in which case all 3 combinations of two parameters are examined. Before calculating the interdependence of any parameters, the optimum point on the error function surface should be located. The assumptions incorporated in the program may not apply at points distant from the optimum. The method used is to study the second derivatives of the error function along the cross sections whose locations are shown in Figure 1.8. All conclusions are drawn from these cross-sections.

The difference in shape between section A-A and B-B is a reflection of the different relative sensitivity of the error function to the given changes in the respective parameters. If the sections are similar, then the effect on the model error of changing parameter 1 by DP(1) is the same as changing parameter 2 by DP(2). If the sections A-A and B-B are not similar, they can be made similar by changing either DP(1) or DP(2), which is equivalent to changing the scale on one of the axes shown in Figure 1.9. The values of DP used are read in the format given in Table 1.4, the cards coming immediately after the last parameter card. All other parameters must be set at their optimum values; the optimum values of the active parameters is

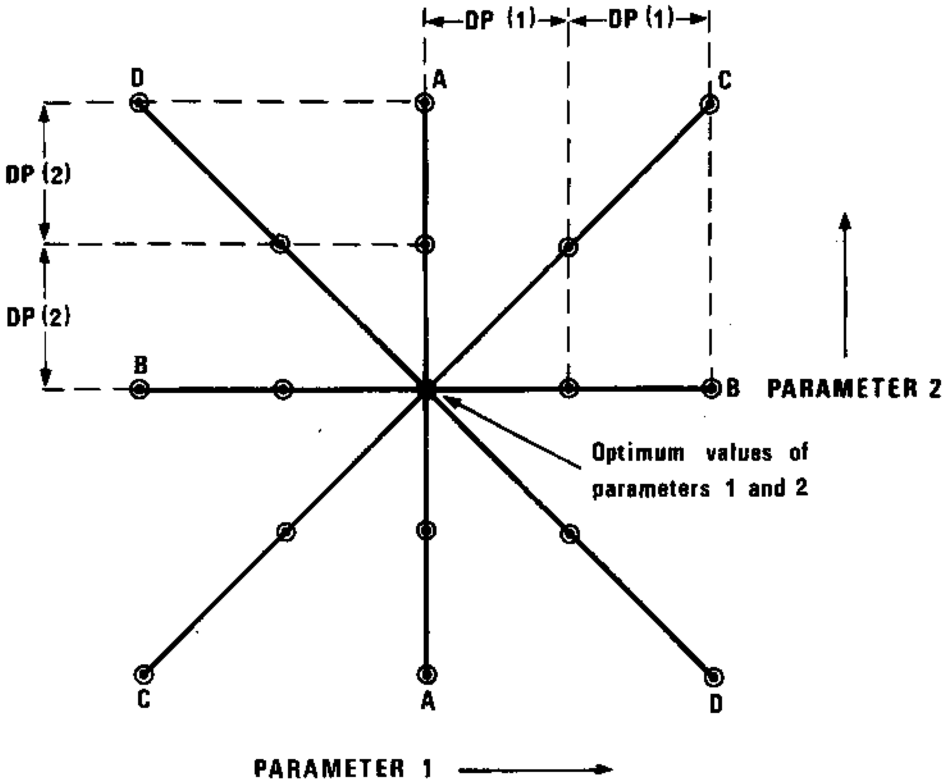


Figure 1.8 Grid of parameter values used in interdependence calculations

Table 1.4 Sensitivity and interdependence calculations - cards defining parameter step sizes

Card	Contents	Format	Columns	Comments
1	(DP(I),I=1,N)	3F10.0	1-30	Parameters must be given in the same order as was defined on the parameter cards, explained in table 1.3 DP is defined in Figure 1.8 II(3) must be 1; N must be 3 or 2

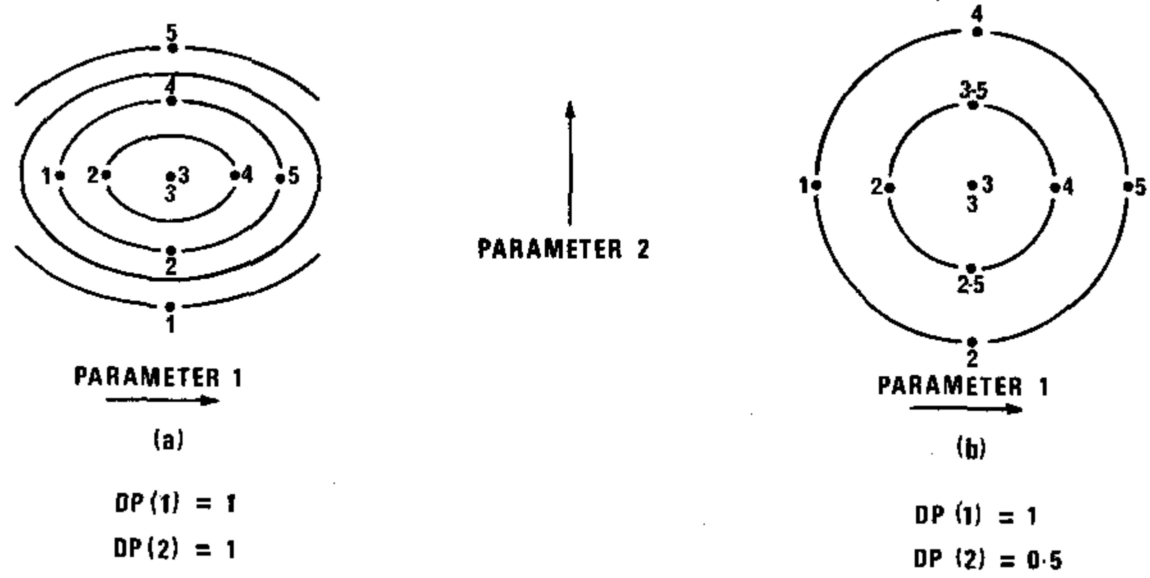


Figure 1.9 Scaling for equal parameter sensitivity

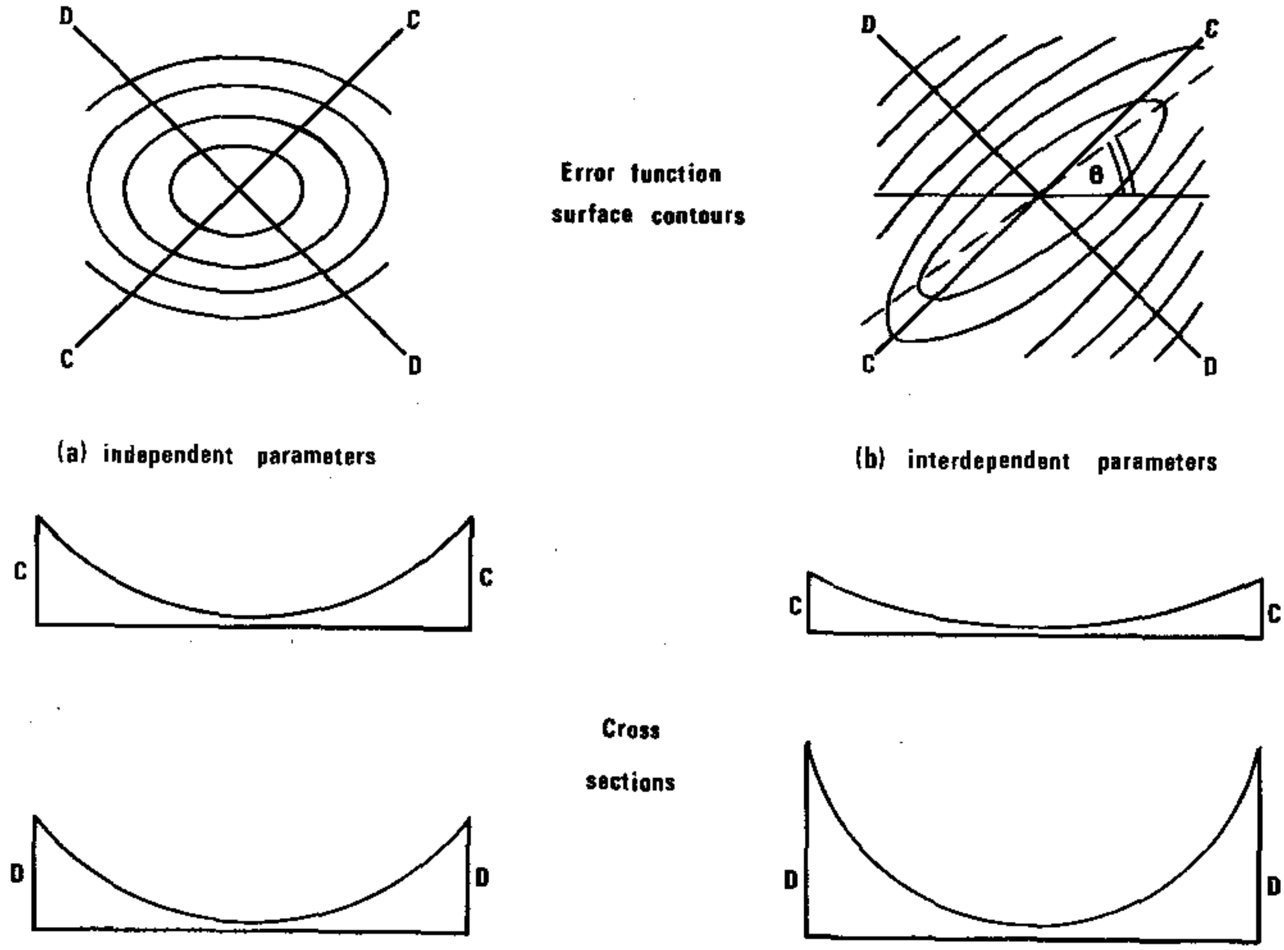


Figure 1.10 Interdependence: diagonal second derivatives

given by the starting values on the parameter cards.

In the case of two interdependent parameters, it is assumed that near the optimum a contour of the error function surface can be represented by an ellipse. If there were parameter interdependence the axis of the ellipse would lie at some angle θ, to the parameter axes, as shown in Figure 1.10(b). The difference in shape between sections C- C and D-D in this figure is a reflection of interdependence between parameters. No amount of rescaling will make C-C and D-D the same shape, nor will it reduce θ to zero. Plinston showed how ρ, akin to the coefficient of correlation between the two parameters, can be expressed in terms of the second derivatives of F in the axial and diagonal directions. The routine calculates F at points on each of the cross sections, and displays the matrix. It then prints out the second derivatives in axial, then diagonal, directions, and finally, estimates of the relative sensitivity of the two parameters at the given step lengths, and of the interdependence ρ, between the two parameters.

(d) Error function surface mapping. The mapping routine is included to assist the modeller in studying an error function surface, particularly when difficulty is experienced in optimisation. Up to 5 parameters can be studied at a time, and up to 9 values specified for each parameter. The mapping routine calls for a run of the model for each possible combination of values of these active parameters. (Because of this, restraint should be exercised in its use, since 9^5 runs through a model is likely to be expensive!) The program outputs a list of the runs through the model, and the error resulting from each run. The user can then select any two-dimensional plane he wishes, and draw up the surface associated with it. All parameters not being mapped should be set within the model to reasonable values, normally to their optimum values if these are known. Parameter cards should be given for the active parameters, although these need not specify starting value or range. A card should then be presented for each active parameter, giving a list of the values it should be assigned in the mapping grid. Details of these cards are given in Table 1.5.

(e) Initial variance of output data. To calculate the efficiency of a model from a knowledge of the error of a model run, the error may be expressed as a proportion of the initial variance of the output data. The program calculates the mean output per modelling period, and then sums the squares of differences from this mean through the duration of the data, giving the initial sum of squares, Fo as

$$Fo = \sum_{t=1}^{n} (\text{observed output}_t - \text{mean output})^2$$

Depending on the output detail index, II(4) (see Table 1.2, card 4), the program either outputs month by month details of the summation of output, then of the summation of the squares, or when II(4) = 0, just the average output and total sum of squares.

Table 1.5 Error function surface mapping - cards defining parameter values to be included in the mapping grid

CARD	CONTENTS	FORMAT	COLUMNS	COMMENTS
1	MNI (I)	I2	1-2	Parameters dealt with in same order as given by KK on the parameter cards: II(3) must be 1: Number of values to be assigned to parameter I
	(PM(I,M),M=1,MNI(I))	9F8.0	3-74	Values to be assigned to parameter I
	NN cards - one for each parameter			

Table 1.6 Cards required by modelling package

CARD	CONTENTS	DESCRIBED IN	WHEN REQUIRED
1	JJ(1) - (5)	table 1.2	always
2	JJ(6) - (10)	table 1.2	always
3	JJ(11) - (15)	table 1.2	always
4	II(1) - (7)	table 1.2	always
5	MM(1) - (5)	table 1.2	always
6	N,NN	table 1.3	when II(3) = 1
7*	NAMES, KK, Y, B, C	table 1.3	when II(3) = 1. B and C only need to be defined when II(1) = 1
8	DP	table 1.4	when II(1) = 3
9*	MNI, PM	table 1.5	when II(1) = 4

* there may be more than one of these cards

At the completion of whichever operation was undertaken, control returns to SUBROUTINE CONTROL, to begin the next operation or alternatively, to end the run. Table 1.6 summarises all the cards that may be required for input to a modelling package run.

1.8 User-supplied subroutines

Two subroutines, namely MODEL and READER, are called by OPTION within the modelling package, and are therefore obligatory, although their contents are in the hands of the user. The user may supply any additional routines for his own convenience. One such, namely SUBROUTINE GRAPH, is suggested, as a means of presenting output.

1.8.1 SUBROUTINE MODEL contains the Fortran coding of the logic of the model being tested, and is called by OPTION to produce a measure, F, of the error of the model run with each set of parameter values selected by OPTION. The composition of MODEL is up to the individual user, but should be constructed around the skeleton shown in Table 1.7, and expanded below.

The first 3 COMMON blocks are obligatory, as they are used in CONTROL and OPTION. The other COMMON blocks only pass information between the user-supplied routines, and may be included or omitted at the discretion of the user. Assignment of parameters to elements in the array of YI should be performed with great care, ensuring that the elements assigned are in the same order as the parameter cards. In other words, YI(1) will be given the starting value from the first parameter card. No check is made on the NAMES item on the parameter card, so the correct order is essential. The order of parameters to be optimised, as well as the limits within which optima may be found, are given on the parameter cards, as detailed in Table 1.3. Initial conditions of parts of the model may be read as parameters, and along with the error function count, F, must be reset for each run through the model.

Having performed all the required initiation, the data to be modelled should be entered. The following rules governing the arrangement and handling of data are included to keep the size of the program to a minimum. Data arrays have been given a maximum dimension of 250 elements. In order to run a model through a considerable amount of data, the data must be broken down into sections, called 'months' in this report. A 'month' consists of IFREQ x IDAYS readings, where IFREQ is constant, and was read as MM(1) on the fifth data card; IDAYS is variable, and should be read as a header to each monthly batch of data. If data is not actually in days and months, these two items should be sufficiently flexible to cope with whatever has to be presented. Each month is read separately and the model fitted to it. It is then discarded, to be replaced by data from the following month. Subroutine READER handles this input of data. There will be no attempt here to describe the body of the model, as this is completely in the hands of the user of this programming package. However, one example of a model which has been used is described in section 2 of this report. Somewhere within the body of the model, an output should be calculated for each data point, and this should be compared with the corresponding observed output, and the error term, F, accumulated. This is the term which is returned to the optimisation. Its definition was discussed in section 1.2.

The remaining coding at the end of the model performs various output functions, including a call to the graphing subroutine, depending on the output indices, II(4) & II(5), and the index, IFIN (see 1.7.2(a) and (b)).

Unless II(4) is set to zero, MODEL will output a heading naming all active parameters and for each run through the model, the total error and the current values of these active parameters. On the final run through MODEL, with IFIN set to 2, a month by month summary of model performance can be obtained. With II(4) set to zero, the only output is the error and active parameter set for the final run through the model.

1.8.2 SUBROUTINE READER is another user-supplied routine, and handles the input of data for use by the model. Table 1.8 details its essential features.

One 'monthly' data batch should be read each time READER is called. The data should be held in COMMON blocks 4 and 5: block 4 for the observed output data, and block 5 for input data. The COMMON blocks must be compatible with those in MODEL. The 'monthly' data is read sequentially, until the final batch has been read: reading will then be resumed at the beginning of the data at the start of the following run through model. Some means of resetting the data file to the beginning, either by the Fortran REWIND statement, or by use of direct access data files, should be available.

Since the data are transferred in and out of the computer core very frequently, it is advisable to store the date on a fast peripheral device, disc or drum for example, rather than on magnetic tape.

1.8.3 SUBROUTINE GRAPH may be used for obtaining graphical output from the final version of a model. When graphical output is required, the index, II(5) should be set to a positive value. GRAPH will be called once for each monthly data batch, as soon as IFIN, the end of optimisation index, has been set.

Since the routine is supplied by the user, it can be written to use any graph plotting device that is available. It should be remembered that the lineprinter can produce plots very cheaply, and give as much detail as many users would require. As techniques for using the lineprinter as a plotter may not be familiar, a possible method is included in Table 1.9, which also gives the skeleton structure of the subroutine. Possible output from this routine is illustrated in Figure 1.11.

1.9 RUNNING THE PACKAGE

The modelling package is a normal Fortran program and its running should provide few difficulties. A few points should be borne in mind.

Table 1.7 Skeleton structure of SUBROUTINE MODEL

```
      SUBROUTINE MODEL                          < no dummy arguments
      COMMON/B1/II(7),JJ(15),MM(5)              < transfer from CONTROL to MODEL
      COMMON/B2/NOPS,NITS,NTIM,IFIN,NM,IFREQ    <       OPTION to MODEL
      COMMON/B3/N,NN,PNOM(20),Y(20),YI(50),IQUIV(50,20),F
      COMMON/B4/OBSOUT(250)                     < transfer from READER to OPTION
      COMMON/B5/                                < transfer from READER to MODEL
      COMMON/B6/                                < transfer from MODEL to GRAPH
      DIMENSION                                 < arrays local to MODEL
      NTIM=NTIM+1                               < increment count of runs through
C ** DETERMINE WHICH PARAMETERS ARE TO BE MANIPULATED.            MODEL
C     PARAMETERS NOT BEING TESTED ARE HELD IN THE ARRAY 'YI'
C     CURRENT VALUES FOR THOSE BEING TESTED ARE IN THE ARRAY
C     'Y', AND MUST BE TRANSFERRED TO THEIR SLOT IN 'YI'
C     BEFORE PARAMETER VALUE ASSIGNMENT
C
      DO 500 J=1,NN
      DO 500 K=1,N
      IF (IQUIV(J,K).GT.0) YI(J)=Y(K)
  500 CONTINUE
C ** ASSIGN VALUES TO ALL PARAMETERS, STARTING VALUES TO
C     STORAGES, ETC.
      parameter  1=YI(1)
      parameter  2=YI(2)
                  .
                  .
      parameter n =YI(n)
      storage  1=YI(n +1)
      storage  2=YI(n +2)
      etc.
C ** SET INITIAL CONDITIONS OF ERROR COUNT,INTERVAL COUNT
      F=0.0
      IM=1                                      < month count
      JK=1                                      < overall period count
  510 CALL READER (IM,IDAYS)                    < read data for next monthly batch
                                                < arguments as used in OPTION
C ** RUN THROUGH MODEL FOR THIS MONTHLY BATCH
      IK=IDAYS*IFREQ
      DO 520 KK=1,IK
            .
     perform modelling operations
            .
            .
```

Table 1.7 Continued

.
.

```
C ** CALCULATE OUTPUT PREDICTED BY MODEL,THEN ERROR FOR
C    THIS INTERVAL
      CALCOUT(KK)=
      ERROR=(CALCOUT(KK)-OBSOUT(KK))      < save error in array if required
      F=F+(ERROR*ERROR)                   < summation of error for model run
                                          < using sum of squares in this example
                                          < or use II(7) to select error criterion
      JK=JK+1                             < increment absolute interval count
  520 CONTINUE
C ** CHECK WHETHER GRAPH REQUIRED
      IF (II(5).GT.0.AND.IFIN.EQ.2) CALL GRAPH(IM,IDAYS)
      IF (II(4).GT.0.AND.IFIN.EQ.2) WRITE(1,200)
  200 FORMAT(       )
C ** CHECK WHETHER THIS IS LAST MONTH
      IF (IM.EQ.NM) GO TO 530
      IM=IM+1                             < increment month count
      GO TO 510
C ** END OF DATA
  530 IF (NTIM.EQ.1) WRITE(1,201)(PNOM(K),K=1,N)
                                          < write parameter heading
  201 FORMAT(14H0   ERROR       ,20(1X,A8)/)
      IF (II(4).GT.0.OR.IFIN.EQ.2) WRITE(1,202)F,(Y(I),I=1,N)
  202 FORMAT(1H ,F9.2,3X,20F9.4)
      RETURN                              < to SUBROUTINE OPTION
      END
```

Table 1.8 Skeleton structure of SUBROUTINE READER

```
      SUBROUTINE READER (IM, IDAYS)       < arguments required by OPTION
      COMMON/B1/II(7),JJ(15),MM(5)
      COMMON/B2/NOPS,NITS,NTIM,IFIN,NM,IFREQ
      COMMON/B4/OBSOUT(250)
      COMMON/B5/                          < input data arrays
      DIMENSION                           < arrays local to READER
C **  SUBROUTINE READS IN DATA TO BE MODELLED. EACH CALL TO THIS
C     SUBROUTINE CAUSES ONE BATCH (IM) OF DATA TO BE READ,CONTAINING
C     (IDAYS*IFREQ) DATA INTERVALS. IDAYS IS STORED ON THE DATA FILE,
C     IFREQ IS GIVEN AS MM(3) IN THE CONTROL DATA. THE PRODUCT
C     (IDAYS*IFREQ) MUST NOT EXCEED 250. THE ROUTINE IS CALLED NM
C     TIMES IN EACH FUNCTION CALCULATION.
C **  IF MODELLING IS NOT TO START AT THE FIRST MONTH ON THE DATA
C     FILE, SKIP MONTHS NOT TO BE MODELLED.
      IF (IM.GT.1) GO TO 505
      IF (MM(5).EQ.1) GO TO 505
      MMM=MM(5)
      DO 500 K=2,MMM
      READ (ch,format  ) IDAYS
      DO 500 I=1,IDAYS
      READ (ch,format  )
  500 CONTINUE
C
C **  A CHECK IS THEN MADE ON THE RATIO OF DATA FREQUENCY TO THE
C     DESIRED MODELLING FREQUENCY
  505 MINPUT=MM(1)
      IBULK=MM(1)/MM(2)
      IF (IBULK-1) 520,510,510
  510 FBULK=FLOAT(MINPUT)/FLOAT(MM(2))
      IF (FBULK-FLOAT(IBULK)) 520,530,520
  520 WRITE (1,200)
  200 FORMAT(1H0,43HINCOMPATIBLE DATA AND MODELLING FREQUENCIES )
      STOP
  530 READ (ch,format  ) IDAYS
      IOUT=IFREQ*IDAYS

< then the user should cause his data to be read from the input stream into the
< appropriate COMMON areas:  the following is an example of how this might be done

      DO 1 I=1,IDAYS
      READ ( ch,fm) (A((I-1)*MINPUT+K),Z((I-1)*MINPUT+K),K=1,MINPUT)
    1 CONTINUE
      DO 3 I=1,IOUT
      OBSIN(I)=0.0
      OBSOUT(I)=0.0
      DO 2 K=1,IBULK
      OBSIN(I)=OBSIN(I)+A((I-1)*IBULK+K)
    2 OBSOUT(I)=OBSOUT(I)+Z((I-1)*IBULK+K)
    3 CONTINUE
      IF (IM.EQ.NM) REWIND ch
      RETURN
      END
```

Table 1.9 Skeleton structure of SUBROUTINE GRAPH

```
      SUBROUTINE GRAPH(IM,IDAYS)
      COMMON/B1/II(7),JJ(15),MM(5)
      COMMON/B2/NOPS,NITS,NTIM,IFIN,NM,IFREQ
      COMMON/B4/OBSOUT(250)
      COMMON/B5/                               < input data arrays
      COMMON/B6/                               < model output arrays
      DIMENSION SYMBOL(120)
      DATA SA,SB,SC,SD,SE/1HI,1HO,1H.,1H*,1H /
      ID=IDAYS*IFREQ
C ** CHECK THAT ALL POINTS LIE WITHIN RANGE OF GRAPH
      YMIN=0.0
      YMAX=100.0
      DO 510 I=1,ID
      IF (OBSOUT(I).LT.YMIN) OBSOUT(I)=YMIN
      IF (OBSOUT(I).GT.YMAX) OBSOUT(I)=YMAX
      IF (PREDOUT(I).LT.YMIN) PREDOUT(I)=YMIN
      IF (PREDOUT(I).GT.YMAX) PREDOUT(I)=YMAX
  510 CONTINUE
C ** WRITE HEADING
      WRITE (1,200)
  200 FORMAT(     )
C ** SET UP LOOP TO WRITE ONE LINE FOR EACH DATA INTERVAL
      DO 540 N=1,ID
C ** SET ALL 120 CHARACTERS ON LINE TO BLANKS
      DO 520 I=1,120
  520 SYMBOL(I)=SE
C ** ARRANGE PAGE LAYOUT
      SYMBOL(1)=SC
      SYMBOL(31)=SC
      SYMBOL(61)=SC
      SYMBOL(91)=SC
C** DRAW HISTOGRAM OF INPUT DATA
      IINP=OBSIN(N)*120.0/100.0
      DO 530 M=1,IINP
  530 SYMBOL(M)=SD
C ** SCALE AND PLOT OBSERVED AND PREDICTED OUTPUTS
      IPOS=OBSOUT(N)*120.0/100.0
      SYMBOL(IPOS)=SB
      IPOS=PREDOUT(N)*120.0/100.0
      SYMBOL(IPOS)=SA
C ** WRITE LINE FOR EACH DATA INTERVAL
      WRITE (1,201) SYMBOL
  201 FORMAT(1H ,120A1)
C ** END INTERVAL LOOP
  540 CONTINUE
      RETURN                                   < to SUBROUTINE MODEL
      END
```

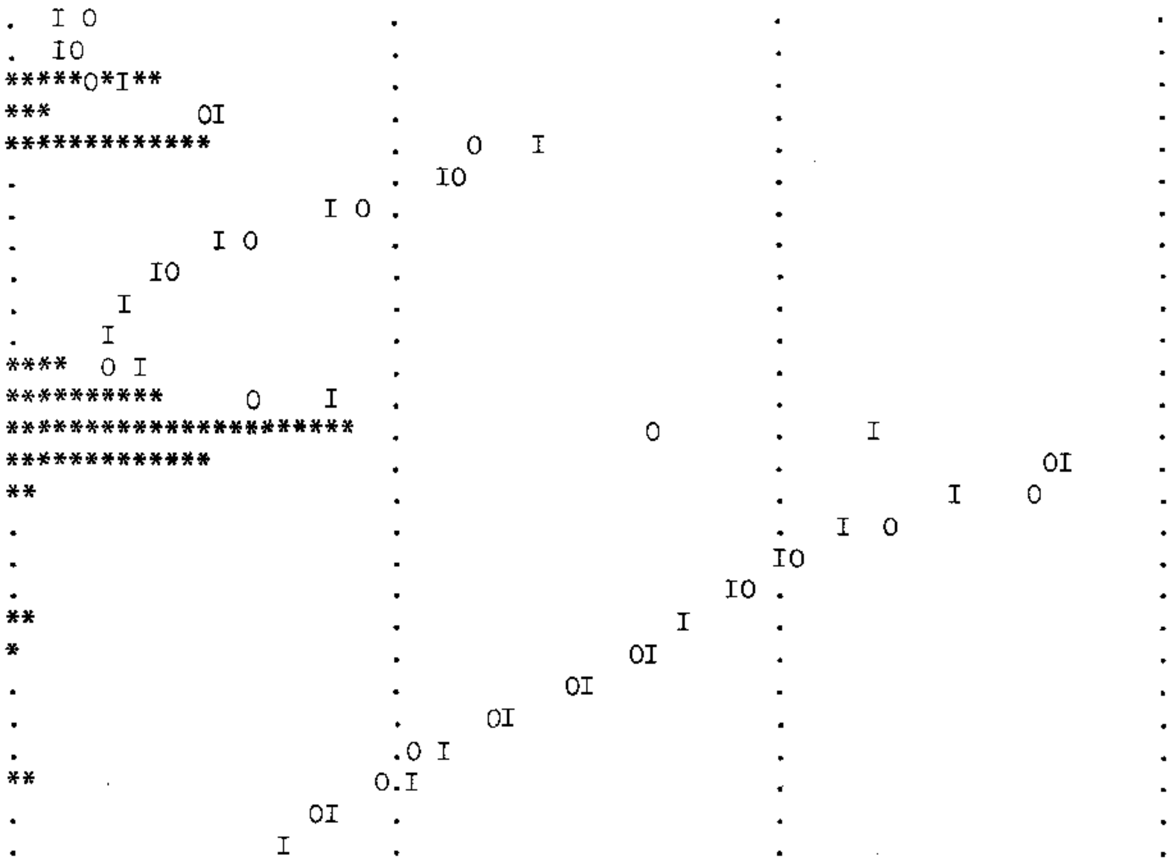


Figure 1.11 An example of graphical output produced by lineprinter

(a) Fortran input/output channel numbers

The following channels have been used:
Channel 1 for lineprinter output
Channel 10 for control card input

Any other channel use, for example the input of the data to be modelled, or the graphical output, is in the hands of the user.

(b) Computing time

An optimisation run can be very expensive in terms of computing time. As a rough guide, the model will be run about 5 times per active parameter per iteration.

(c) Input of data to be modelled

The choice of input device for the reading of the modelling data will be influenced by the necessity to read through the data during each run through the model. There are two implications:

- as there may be a large number of data transfers, a fast device should be chosen

- as, at the beginning of the second and subsequent runs through the model, data must be read from the beginning of the data file, some facility for reinitiating the data file is necessary.

PART 2 CONCEPTUAL MODELLING OF CATCHMENT HYDROLOGY

2.1 Background

In the past few years, several models for predicting catchment behaviour have been produced at the Institute of Hydrology.

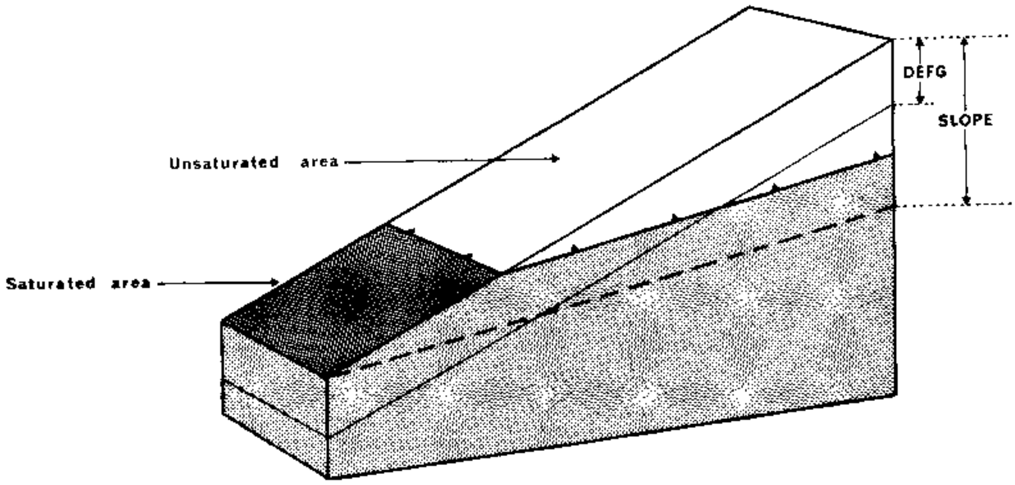
Nash and Sutcliffe (1969) reported on a model which considered a catchment to be analagous to a vertical stack of soil layers, all layers having equal water storage at field capacity. Evaporation took place at the potential rate from the top layer. On exhaustion of the top layer, evaporation was allowed from the second layer, at a factor, C, of the potential rate; then, with its subsequent exhaustion, from the third layer at C^2 of potential, and so on. Runoff was generated as a factor, h, of the rainfall excess (rainfall less potential evaporation) in any time interval. The remainder of the rainfall filled up the soil layers, starting with the top layer, and working down the stack as successive layers became full. Any surplus rainfall after the entire stack had been brought to field capacity, formed runoff.

Approaching the problems from a different angle, Mandeville and others (1970) hypothesized a model which considered spatial variation in vegetation, rather than vertical variation within the soil profile, to be important. They represented each vegetation type as a single storage, into which rainfall excess was applied and evaporation removed. Evaporation took place at a rate which was the product of a crop factor and the potential rate, until the crop's wilting point was reached, after which it was reduced to a rather arbitrary 10% of potential rate. Runoff was produced when a storage overflowed. The vegetation types considered were deep-rooted trees, which covered about 20% of the study area at Grendon Underwood, and agricultural crops, both arable and pasture, which made up the remainder, and in which wilting was known to occur during prolonged dry spells. Early runs with this model produced runoff only by overflow of storages.

Observing that storms were capable of producing flow even when the soil in both zones was relatively dry, a further vegetation zone was introduced. Known as the 'riveraine area', this was kept completely wet, so that any rain falling on it immediately became streamflow. The size of the riveraine area was found by optimisation, and comprised 14% of the agricultural land on the Ray catchment at Grendon Underwood.

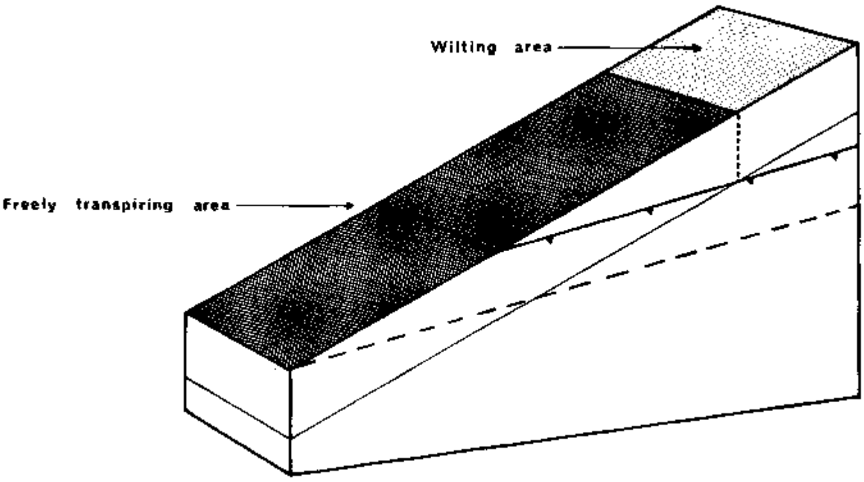
2.2 A variable saturated area

A natural extension of this model was the idea that the riveraine area might not be a constant proportion of the catchment area. As the catchment became wetter, so the area of saturation would become progressively larger, until when the entire catchment storage system was full it could be considered as 100% riveraine area. This model and its operation can be considered schematically as shown in Figure 2.1. It is clear that there is some relationship between the proportion of rainfall forming streamflow and the soil moisture status. This function is of the form illustrated by Figure 2.2(a). It will further be noticed that the



a. REACTION OF A UNIT SECTION OF CATCHMENT TO RAINFALL

- rain falling on saturated area forms direct runoff
- rain falling on unsaturated area reduces moisture deficit, raises saturation surface and increases saturated proportion of catchment



b. EVAPOTRANSPIRATION MECHANISM OF THE MODEL

- evapotranspiration increases moisture deficit, reduces saturated area and increases wilting area
- evapotranspiration occurs at a minimal rate from the wilting area

Figure 2.1 A model of catchment behaviour with variable saturated area

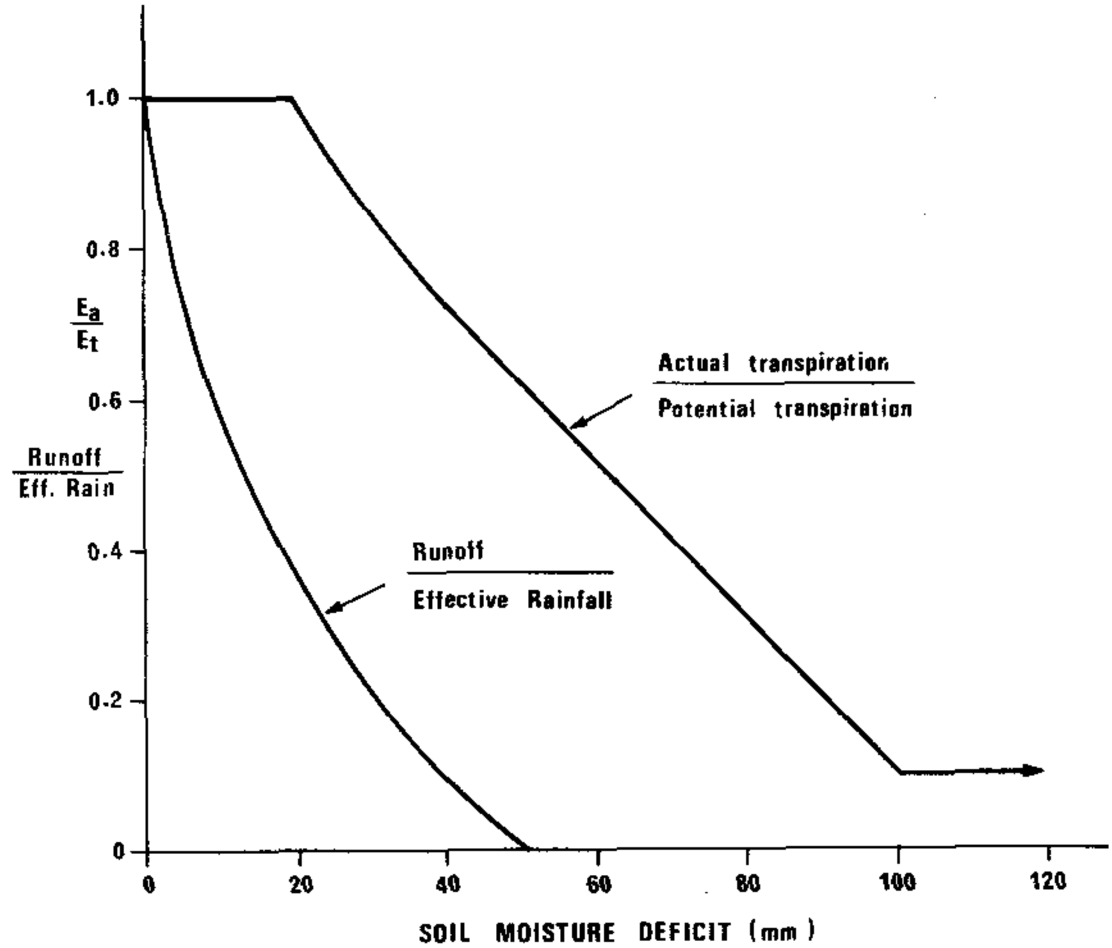


Figure 2.2 Relationships embedded in the variable saturated area model

Table 2.1 A comparison of results from previous models

Models run on data from the R. Ray catchment at Grendon Underwood, from 9 November 1963 to 24 November 1967.

Sum of squares statistics are obtained from 204 precipitation events although the models all used a data interval of 3 hours.

Total precipitation	2729 mm
Open water evaporation	2658 mm
Observed runoff, Qo	777 mm
Initial sum of squares, Fo	9967 sq mm

Model	Error, F (mm^2)	Efficiency (%)	Predicted runoff, Qp (mm)	Error in volume prediction (Qp-Qo)/Qo
(1) 'LAYER' model SM.1, ZHI/T/CY	1204	87.9	860	+ 10.7%
(2) 'AREA' model without riveraine area SM.2, ADI/T/HY	1191	88.1	780	+ 0.4%
(3) 'AREA' model with riveraine area SM.2, ADHI/T/Y	985	90.1	849	+ 9.3%
(4) 'VARIABLE SATURATED AREA' model, ADSTI//Y	701	93.0	840	+ 8.1%

ratio of actual evaporation to potential evaporation is also a function of the soil moisture status, as shown in figure 2.2 (b). These two important functional relationships are defined by just 2 parameters, namely DEFG and SLOPE shown in Figure 2.1.

In order to compare the performance of this model with those outlined earlier, it was fitted to the same data from the Institute of Hydrology experimental catchment of the River Ray at Grendon Underwood. Table 2.1 shows comparable results from the 4 models. The data used in each case was approximately 4 years in duration at a constant 3-hourly time interval. The error function and initial variance, were, however, not the simple sum of squares of the 3-hourly errors. Instead, the data were broken up into storm events, of which there were 204 in the 4 year period. The volume of streamflow resulting from each storm event was calculated from the observed data, making adjustments at the beginning and end of events for that water in storage in the system. These figures were then compared with predicted volumes resulting from applying the model for the 204 events and the error F given by

$$F = \sum_{n=1}^{204} (Qo_n - Qp_n)^2$$

where Qo_n is the observed streamflow resulting from the precipitation in period n and Qp_n is the predicted streamflow resulting from that precipitation.

The effect of using such an error criterion is to eliminate any problems with the shape of the hydrograph. As can be seen, the increase in efficiency of this model over that of previous models was quite marked.

2.3 The mathematical function approach

It might appear that the move towards modelling with mathematical functions is automatically a move away from conceptual hydrological modelling and towards a 'black box' statistical approach. This is not the case.

The operation of all the models described in sections 2.1 and 2.2 could be expressed in terms of relationships between the soil moisture status, evaporation reduction factors and runoff percentages. Figure 2.3 shows the same relationships as they are embedded in the logic of the Nash and Sutcliffe 'layer' model. (Fig 2.3(b) is an approximation, as it does not take account of the distribution of moisture in the soil profile, as allowed by the model.) The dashed line on Figure 2.3(a) might result if spatial variation of the soil moisture deficit about the catchment were considered in this model. Different parts of the catchment would reach field capacity at different times resulting in a more gradual change in runoff percentage from h to 1 than in the original 'lumped' version of the model. Figure 2.3(b) could easily be replaced by the dashed line shown, as this would be the effect of reducing the capacity of each layer until it approached zero. The effect of spatial variation superimposed on this would be to accentuate the curve.

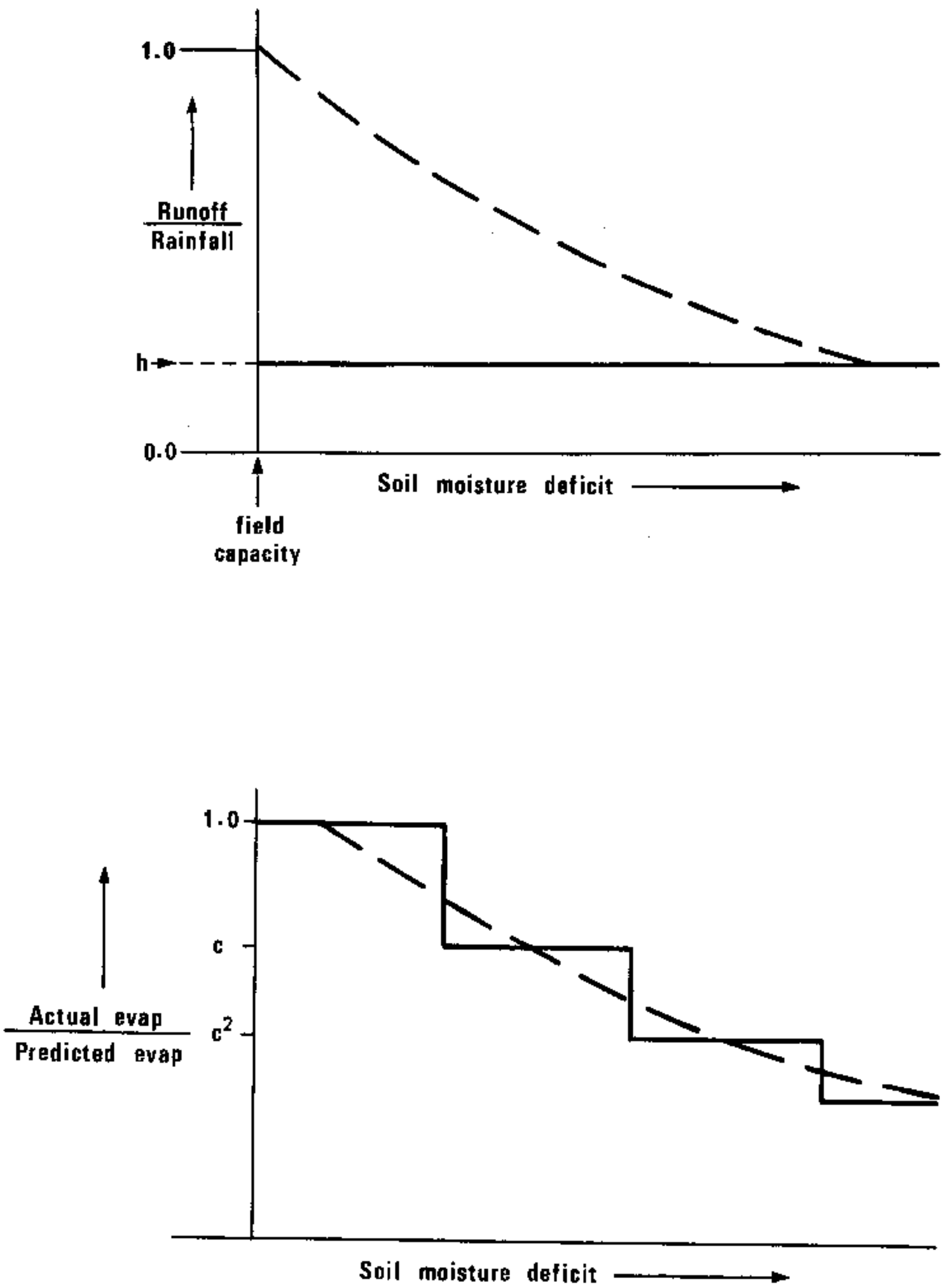


Figure 2.3 Relationships in the 'layer' model

Having been able to express all three of the models mentioned in terms of two basic relationships, albeit using different reasoning and nomenclature, it becomes clear that these models were just three cases of a very general model. In this the same two functional relationships may be expressed in more general terms, which has two important advantages:

(a) the form of the individual relationships are no longer rigid

(b) the form and position of the runoff/SMD relationship is independent of the form and position of the evaporation/SMD relationship.

The breakdown of these models into this general, functional form is the basis of the general conceptual model which has since been used to model all of the Institute's experimental basins. Several sections have been added to the soil moisture model to cater for the various conditions arising within these basins. These comprise an interception storage element, a groundwater storage and snow accumulation and melt routines.

2.4 The general model

The model is shown schematically as Figure 2.4. It comprises a maximum of five storage elements for any 'lump' of a basin. If a basin is characterised by two or more hydrologically distinct sub-areas, this distinction can be made by duplicating parts of the model. The storage elements have been labelled as follows:

Interception store
Snowpack
Soil moisture store
Groundwater store
Channel routing reservoir

Fluxes between storages are shown in the figure. The methods of calculation of these various fluxes are described for each storage unit below.

2.4.1 Interception store

This is a shallow storage element, whose capacity, SS, is normally less than 10 mm of water, into which all rain falls. Only when the store is full, that is when the contents CS are equal to SS, does any incoming rain overflow into the soil moisture store. The effective rainfall, ERAIN, is the amount passed on to the soil moisture store, and is calculated as

$$ERAIN = RAIN - (CS-SS)$$

Water is lost from the storage by evaporation. The evaporative demand (an estimate of open water evaporation produced by Penman's equation) may be enhanced by a factor, FS, in order to calculate the losses, ES, from the storage. For grassland, the value of FS would be expected to be in the order of unity as the system is acting like an open water surface. For forest cover, however, Stewart and Thom (1973) suggest

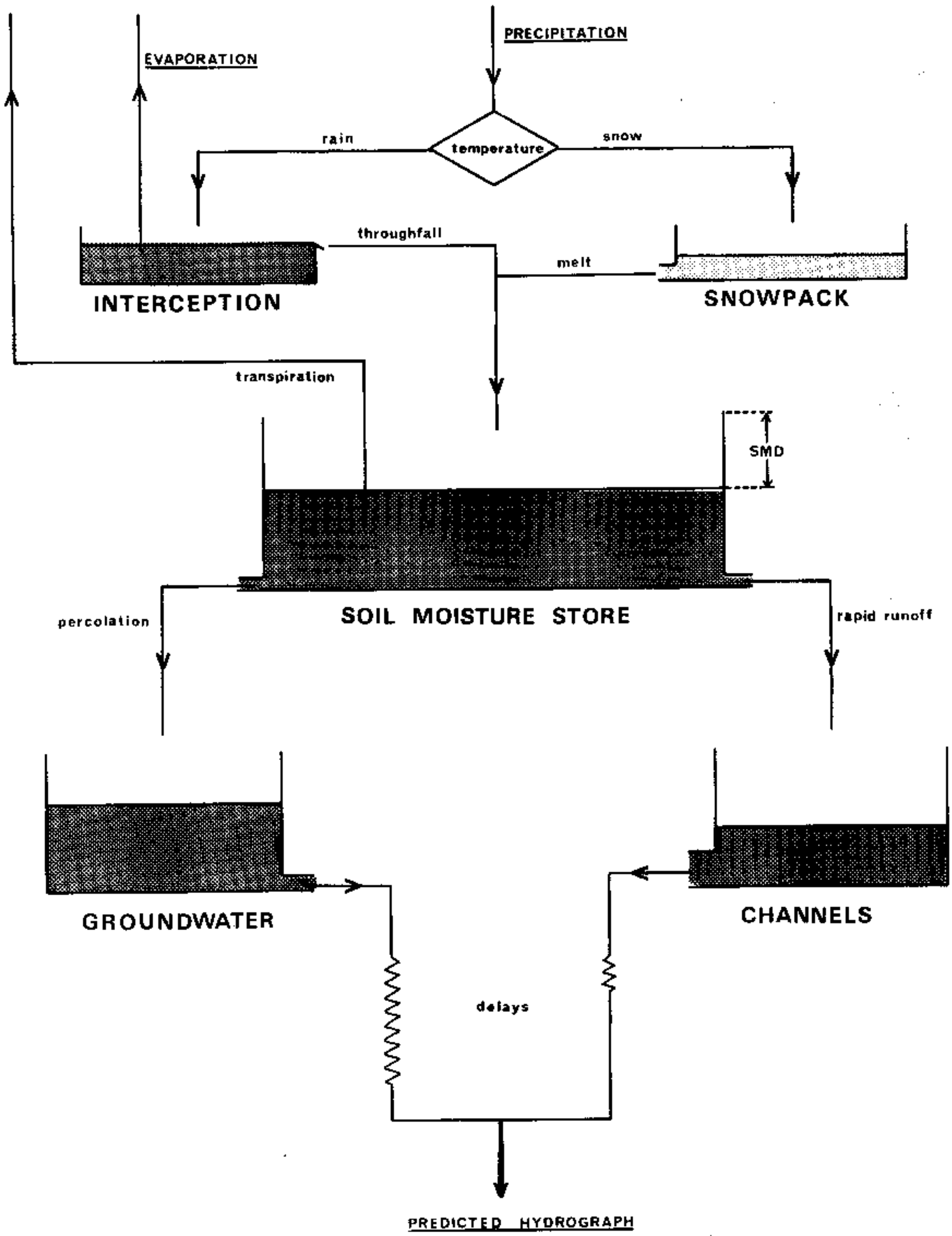


Figure 2.4 Schematic representation of the general catchment model

that, because of increased roughness of the taller vegetation, FS might take a value of 4 or 5. If any evaporative demand remains when the storage has been emptied, it is passed on to the soil moisture storage as EEVAP. To eliminate the effects of FS in the calculation of EEVAP, it is given as:

$$EEVAP = EVAP - \frac{ES}{FS}$$

The outputs from this storage unit may, therefore, be summarised as:

(i) Losses to evaporation, ES = EVAP x FS
(ii) Effective rainfall, ERAIN = RAIN - (SS - CS)
(iii) Residual evaporative demand, EEVAP = EVAP - ES/FS

2.4.2 Snowpack

When precipitation falls at a temperature less than a limiting temperature, SLIM, it is considered to be snow. It is either added to an already existing snowpack, or if none exists, it forms a new pack. The size of the pack is expressed in mm of water equivalent of the snow, rather than as a depth of snow. (To convert this to snow depth requires a knowledge of snow density, which might be around 0.1 gm/cc for fresh snow, rising to about 0.5 gm/cc for melting snowpack). Snow is lost from the pack as melt, which is calculated solely as a function of temperature, once a limiting termperature, TLIM, has been passed, such that MELT = TFAC (TEMP - TLIM). This melt is applied directly to the soil moisture store by-passing the interception storage, and deducted from the pack.

Evaporation is known to take place from a snow surface, but rates are very low (due to the energy required to overcome the latent heats of fusion and vaporisation) and have been ignored in the model for the sake of simplicity.

One factor which was thought likely to be important was the variability of the areal snow cover. This becomes significant once parts of the basin become bare, as these parts can no longer contribute snowmelt to the soil moisture store. A simple relationship has been included which gives the proportion of the basin covered, APACK, as a function of the mean snow pack content, as shown in figure 2.5. Melt is multiplied by APACK before being transferred to the soil moisture store, and deducted from the pack. This very simple approach to snow accumulation and melt is not likely to reproduce snow behaviour very accurately. In the typical absence of very much data, notably of accurate precipitation estimates during periods of snow and of detailed temperature measurements, there is little point in adding complications. A more thorough approach to the problem of snowmelt is given by the US Army, Corps of Engineers (1956).

2.4.3 Soil moisture store

The soil moisture store is the hub of all the hydrological activity of the model. It accepts effective rainfall and snowmelt as inputs, against which it balances losses to transpiration, streamflow and as deep percola-

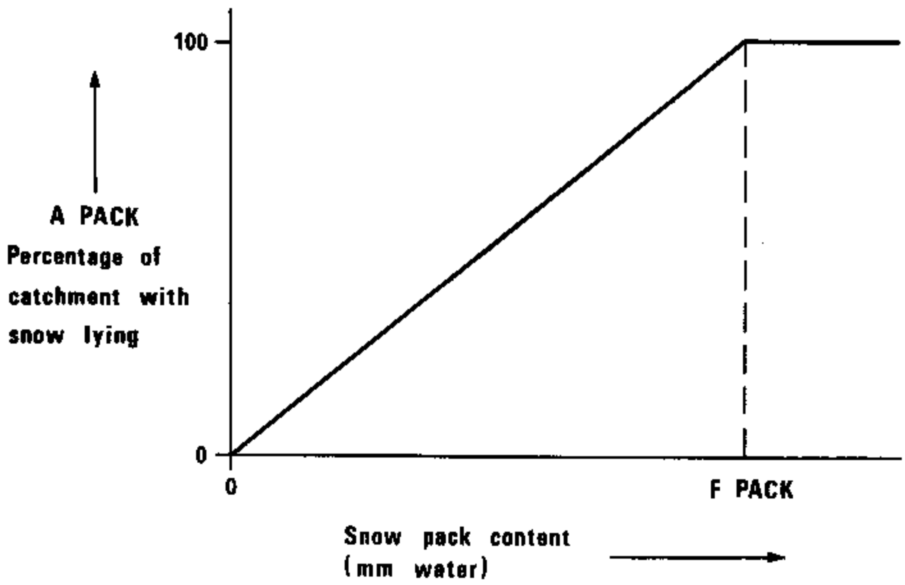


Figure 2.5 Relationship between areal extent of snow cover and mean snow depth

tion to groundwater. The rates of all these losses are assumed to be related to the contents of the store. These contents are recorded as soil moisture deficits below field capacity, and are monitored within the model as DC.

Perhaps the most important output from a hydrologist's point of view is the outflow to the river channel. This is considered to be a function of both the contents of the store and the combined effective rain and snowmelt inputs, such that the outflow, ROFF, is given as:

$$ROFF = ROP \times (ERAIN + MELT)$$

and ROP as some function of the soil moisture deficit, DC. Input which is not immediately lost in this way, ie (1 - ROP) x (ERAIN + MELT) reduces the deficit, DC.

Transpiration, or water use by vegetation, is then calculated. It is assumed to take place at a rate related to the residual evaporative demands, EEVAP (that not satisfied by the interception store) and the current moisture deficit, DC. The actual transpiration, ET is given by: ET = ECP x EEVAP, where ECP is again related to DC. If the transpiration calculated in any period is greater than an upper limit, ECL, it is reduced to this value, which might represent the maximum rate at which the crop is capable of transpiring water.

The third outflow from this soil moisture storage is deep percolation to a groundwater storage. This occurs at a rate GPR, which is also assumed to be related to the contents of the soil moisture store, DC.

2.4.4 Groundwater store

The groundwater store receives water by percolation from the soil store, at a rate GPR, and this is added to the contents of the store. The only loss from the store is as baseflow to the stream, at a rate which is a function of the contents of the groundwater store, GS, and the rate of inflow, GPR. Baseflow, GRO, is given as:

$$GRO = \frac{\left(\frac{GS}{GSM} - GF.GPR\right)}{(1 - GF)}$$

This equation is equivalent to a Muskingum routing procedure, described by McCarthy (1938). Clearly, by setting the factor GF to zero, this becomes a simple linear reservoir. Having produced the contribution to streamflow, it can be delayed in time by an interval, GDEL.

2.4.5 Channel routing reservoir

The rapid outflow, ROFF, produced by the impact of effective rainfall and melt on the soil moisture storage, enters a channel routing system. This is conceived simply as a non-linear reservoir, output from which is related to the reservoir contents by the equation $RO = RK \times (RSTORE^{RX})$. No losses are allowed from this reservoir. The output from the channel routing reservoir is delayed by a constant time interval, RDEL, and then added to the baseflow output by the groundwater store. This sum is the predicted outflow at the catchment outlet.

2.5 Appraisal of the general model

The system of storages suggested would appear to be sufficient to model the hydrological activity in the majority of cases. Local conditions might necessitate the addition of fluxes between storage elements. For example, in situations of extreme heat, or where a high proportion of a catchment consists of an open water surface, losses from the channel reservoir to evaporation might be significant. In the latter case, direct input of precipitation to this reservoir would have to be considered. In some circumstances, losses from the channel reservoir to the groundwater system might necessitate the inclusion of a flux between the two.

In the event of a catchment containing two or more sections requiring individual modelling, it would be appropriate to duplicate the interception, snowpack and soil moisture storages, though not necessarily the groundwater or channel reservoirs.

On the whole, it would appear that the model described should be sufficiently flexible to handle streamflow generation on most small catchments. Extension of its use to large catchments would necessitate additional consideration of the channel routing component, spatial variation of input data and so on.

PART 3 APPLICATIONS OF THE GENERAL MODEL

3.1 General

The model described in Part 2 has been used to try to increase our knowledge of the hydrology of the Institute's experimental catchments. There have been three main aims. The first, at which most effort has been directed to date, is to anticipate the effects of changes in land use on streamflow. The second aim is to produce a model which, after a minimal period of calibration, will enable a short streamflow record to be extended by consideration of historic climatological data. The final aim, one which requires a great deal more work, is to be able to predict the streamflow from a catchment solely from climatological data, and without the use of historical hydrological data for model calibration.

This section is confined to a discussion of progress on the first of the aims outlined above, except in the final subsections, where some preliminary remarks are made on the other two objectives.

3.2 The model used

The model used is exactly as described in section 2.4. The parameters, which have been listed and explained in Table 3.1 are mostly referred to in that section. The only exceptions to this are the parameters which appear in the functions relating to the soil moisture store. In section 2.4, the forms of these functions was not described. It was suggested however that the percentage runoff, the actual evapotranspiration and the percolation rate were all related to soil moisture deficit. Figure 3.1 shows the form of the relationships that were used.

3.2.1 The runoff percentage function

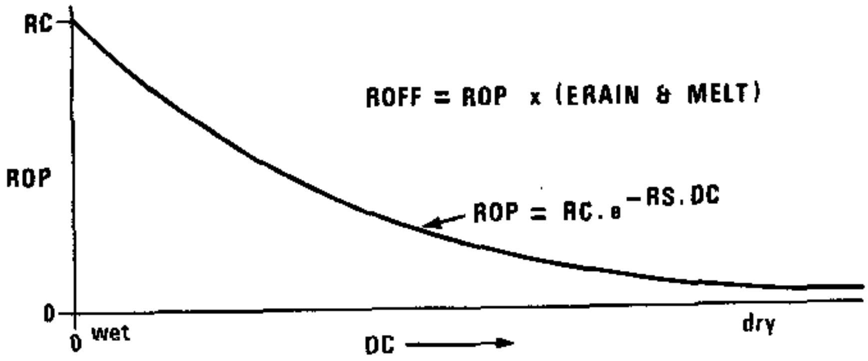
The proportion, ROP, of the input to the soil moisture store which forms rapid runoff is seen to decrease as the catchment dries out. The first relationship tested was the simplest possible, a linear decrease in ROP with increasing deficit. Although this worked reasonably well, a function giving a rapid reduction in ROP as the catchment dried from saturation, and a much slower reduction for a similar drying at higher soil moisture deficits, would be more satifactory on the whole. The exponential decay function used satisfies this requirement, is simple, and is efficient in parameters. There are two parameters: RC, the ROP at zero deficit, and RS, governing the rate of decay (see Figure 3.1(a)).

3.2.2 The actual transpiration function

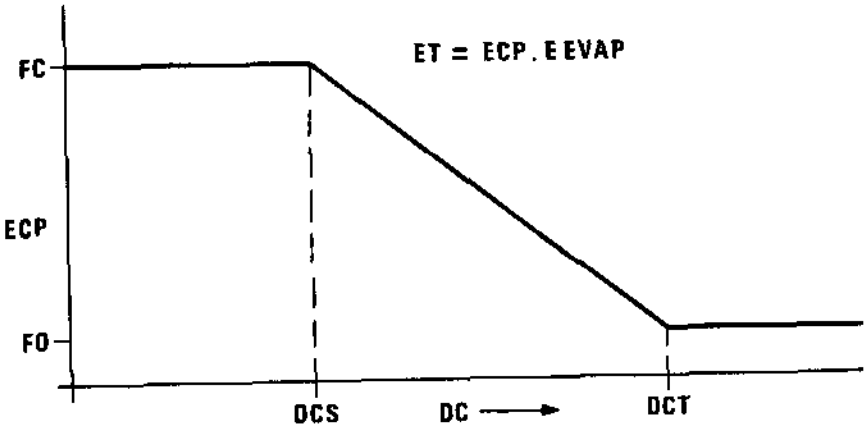
The process of transpiration is a complex relationship between plant, climate and soils. Each imposes constraints on the rate of transpiration at any one time. Penman (1948) has formulated relationships between climatic factors and potential evaporation but in order to predict the moisture actually transpired by plants from his figures some estimate of the resistance to moisture transfer must be made. It is assumed that there is an upper limit to transpiration which is the maximum rate

Table 3.1 Parameters in the general catchment model

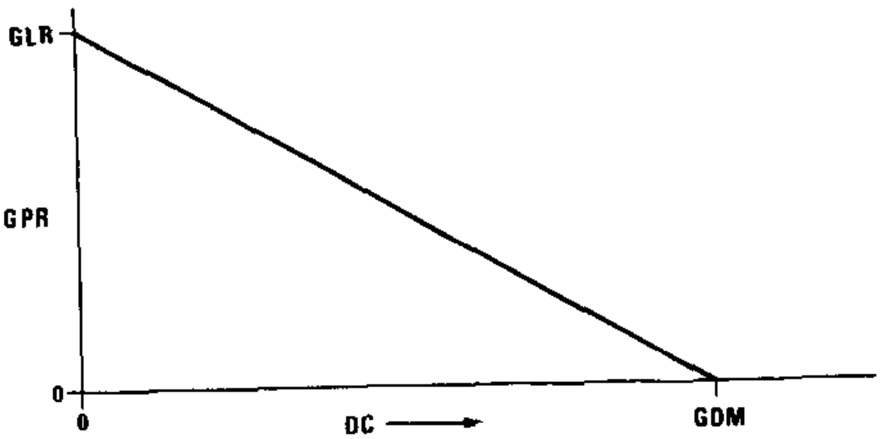
FORTRAN SYMBOL	EXPLANATION OF PARAMETER
(1) Interception parameters	
SS	Size (in mm) of interception store
FS	Factor by which open water evaporation estimate is multiplied for estimating evaporation from interception store.
(2) Snowfall parameters	
TLIM	Temperature (^{o}C) below which precipitation assumed to be snow
SLIM	Temperature (^{o}C) above which melt occurs
TFAC	Factor converting temperature excesses into melt
FPACK	Snowpack (mm W.E.) above which complete snow cover assumed
(3) Transpiration parameters	
ECL	Maximum rate of transpiration (mm/interval)
FC	Factor by which open water evaporation estimate is multiplied for estimating potential transpiration
FO	Factor by which open water evaporation estimate is multiplied for estimating minimum transpiration rate at soil moisture deficits greater than DCT mm
DCS	Soil moisture deficit (mm) below which FC applies
DCT	Soil moisture deficit (mm) above which FO applies
(4) Runoff parameters	
RC	Proportion of effective rainfall forming runoff at zero S.M.D
RS	Parameter defining rate of reduction of runoff proportion with increasing soil moisture deficit
RK	Output constant for non-linear runoff routing reservoir
RX	Exponent for non-linear runoff routing reservoir
RDEL	Time delay (intervals) in runoff routing
(5) Groundwater parameters	
GLR	Percolation rate (mm/interval) at zero S.M.D
GDM	Soil moisture deficit (mm) above which percolation zero
GSM	Output constant for linear groundwater reservoir
GF	Factor for including input to groundwater in calculation of output.
GDEL	Time delay (intervals) in groundwater routing



(a) Loss to streamflow



(b) Loss to transpiration



(c) Loss to groundwater

Figure 3.1 Functional relationships within the soil moisture component of the model at which the plant can release water in the most favourable circumstances. The parameter, ECL, is expressed as millimetres of water per time interval. Apart from this constraint, transpiration is assumed to occur at a proportion of the potential rate, that proportion being dependent on the soil moisture deficit, as shown in Figure 3.1(b). The upper limit, FC, could be considered as a crop factor, and the deficit, DCS, the greatest that the plant can withstand without showing stress and cutting down its transpiration. At deficits greater than DCT mm, there is only a base level of transpiration of a proportion FO of the potential rate.

3.2.3 The percolation function

Percolation from the soil moisture store to groundwater is assumed to be solely dependent on the soil moisture deficit. It varies from a maximum of GLR mm per time interval at zero deficit, to zero percolation at a deficit of GDM mm (see Figure 3.1(c)).

3.3 Modelling changes in catchment characteristics

Much of the research effort of the Institute goes into the study of the hydrological response to changing catchment conditions. At Plynlimon in Central Wales, two adjacent catchments, similar in most respects but with contrasting land use, have been densely instrumented, with a view to predicting the effects of afforestation on catchment yield, flood peaks and minimum flows. At Coalburn, in Northumberland, the hydrology of a single small catchment has been monitored for five years prior to afforestation. Changes caused by afforestation and its associated drainage programme will be observed over the next few years. On the site of the new city of Milton Keynes in Buckinghamshire, the Institute has set up two small catchments. One of these will remain undeveloped for many years, whilst the other is in one of the first areas to be developed. Studies will be made of changes brought about by the introduction of storm drains and the large proportion of impervious surface.

Differences in hydrological behaviour caused by variations in land use, either in space or time, should be evident from the values of a few crucial parameters in the model. As discussed in section 1.4, the values of parameters for a particular model application are found by an objective fitting of the model to data. Although parameters will thus have been evaluated objectively for each condition, the warnings given in section 1.5 should be borne in mind when reviewing results.

3.3.1 The Plynlimon catchments

Some quantitative assessment of the significance of vegetation type on catchment water yield was made by applying the general model to the two neighbouring Plynlimon catchments. It was expected that different optimum values of a few key parameters would indicate parts of the hydrological cycle which differed between the forested Severn catchment and the hill pasture of the Wye catchment. An estimate would be made of the magnitude of these differences, when viewed on a catchment scale.

The results of fitting the model to data from the two catchments for the 15 month period October 1971 to December 1972 are included in Table 3.2.

Table 3.2 A comparison of the behaviour of the Severn and Wye catchments

	Severn	Wye
SS	1.43	0.362
FS	4.79	1.00
TLIM	0.0+	0.0+
SLIM	0.7+	0.7+
TFAC	0.1+	0.1+
FPACK	20.0+	20.0+
ECL	2.0+	2.0+
FC	0.594*	0.594*
FO	0.2+	0.2+
DCS	2.66	85.2
DCT	100.0+	100.0+
RC	0.722	0.800
RS	0.037*	0.037*
RK	0.083*	0.083*
RX	1.544*	1.544*
RDEL	0.5+	0.5+
GLR	1.45*	1.45*
GDM	25.6*	25.6*
GSM	44.2*	44.2*
GF	0.0+	0.0+
GDEL	8.0+	8.0+
F	341.1	406.3
Fo	3580.0	4331.5
Efficiency	90.5%	90.6%
ΣQo	2026.0	2396.4
ΣQp	2200.6	2394.4
Volume error	+ 8.6%	- 0.1%

Notes:

+ value not optimised

* parameter optimised but constrained to adapt the same value in both catchments.

Values of some of the parameters in the model have not been found by optimisation. For example, there was insufficient snow in this period to obtain reliable values for the four snow parameters, nor was the summer of 1972 dry enough to invoke the parameters at the 'wilting' end of the transpiration function. Then, to simplify the comparison, some other parameters were constrained to optimise to the same value in both catchments. This left just four key parameters which were allowed to find a different optimum value in each catchment. These four were the two surface store parameters, SS and FS; the maximum runoff percentage, RC; and the soil moisture deficit at which transpiration reduction started, DCS.

Results of the optimisation suggest that there are differences between the interception and transpiration mechanisms of the two catchments, though apparently not in the generation of runoff.

A more thorough investigation on these lines is continuing to test more parameters and on a longer sequence of data. This work will be reported at a later date.

3.3.2 Land use changes with time

Plynlimon is an example of spacial variation of catchment characteristics. Another type of variation which we might require to model is change through time. At Coalburn in Northumberland, the 155 hectare catchment which was peat bog in 1967 is now a very youthful pine forest. For five years from 1967 to 1971 the natural conditions were monitored by the Institute with the assistance of the Cumberland River Authority. The general model has been fitted to the data from the catchment in this initial state. As data from the modified catchment comes in, it will also be modelled, and changes in catchment characteristics will, it is hoped, manifest themselves as changes in previously stable parameters.

As an exercise in testing the stability of the model through time, it was applied to data from the catchment of the R. Ray at Grendon Underwood. As there has been no observed systematic change in catchment characteristics during the period since October 1963 when data collection began, it was hoped that there would be no significant shifts in optimised parameter values with time. The model was fitted separately to 4 sets of 24 months' data, as shown in table 3.3. The quality of the model fit is remarkably consistent throughout the eight years. However, the optimised values of certain parameters do not show the same consistency. The problem is mainly confined to the evapotranspiration phase of the model, and is caused by the severe interdependence of parameters in this area. As a test of the significance of any of the parameter variation, a further run was made, this time holding all parameters to the mean of their four optimised values. The results of this run are given in the final column of Table 3.3. The overall efficiency of the model is reduced by about 5%, from 89% to 84%, indicating that the variation in parameter values between the 4 sets of data was more than simply a problem of interdependence: in fact that each period requires a slightly different model. The final efficiency of 84% would however, be high enough to warrant this average model's use for many predictive purposes.

Table 3.3 Model Calibration at Grendon Underwood

Parameter	Oct 1963 -Sep 1965	Oct 1965 -Sept 1967	Oct 1967 -Sep 1969	Oct 1969 -Sep 1971	Oct 1963 -Sep 1971
SS	5.6	13.7	7.7	9.0	9.0
FS	1^+	1^+	1^+	1^+	1
TLIM	2.1	1^+	0.8	1.6	1.5
SLIM	- 4.78	- 4.84	- 4.5	- 2.14	- 4.1
TFAC	0.8	0.1^+	1.0	1.2	1.0
FPACK	10.9	20^+	21.5	11.5	14.6
ECL	∞	∞	∞	∞	∞
FC	0.806	0.643	0.576	0.866	0.723
FO	0.366	0.276	0.478	0.324	0.361
DCS	2.35	44.3	76.1	78.5	50.3
DCT	48.5	48.2	76.1	79.9	63.2
RC	0.677	0.874	0.762	0.868	0.795
RS	0.0319	0.0375	0.0323	0.0292	0.0327
RK	0.0511	0.0493	0.0469	0.0393	0.0466
RX	1.679	1.574	1.587	1.550	1.598
RDEL	2.41	2.31	2.26	1.63	2.15
F	19.2	32.5	59.4	16.9	184.7
FO	214.5	270.0	517.5	159.8	1162
Efficiency	91.0%	88.0%	88.5%	89.4%	84.1%
ΣQo	245.8	502.6	498.6	305.3	1552.3
ΣQp	244.8	495.4	499.2	324.5	1485.1
Volume error	- 0.4%	- 1.4%	+ 0.1%	+ 6.3%	- 4.3%

3.4 Extending streamflow records using climatological data

An early approach to the comparison of the different land use of the two Plynlimon catchments was made using a lengthy sequence of precipitation data to test the long term effects of afforestation. Daily precipitation measurements were available for a fifty year period from a gauge at Blaenau Ffestiniog, in an area of similar, though somewhat wetter, climate to that at Plynlimon. At the time of this study, the Institute had collected data from the Wye catchment for nearly three years, but had very little useful data from the Severn catchment.

The general model was fitted to daily data from the Wye catchment in order to obtain estimates of the values of the various parameters in the model. This model was then applied to the 50 year sequence of data, and 50 years of daily streamflows were estimated.

To test the sensitivity of the daily streamflows to manipulation of land use, the model was re-run with different sizes of interception storage, and with different transpiration/SMD functions (these being the aspects of the catchment hydrology in which change would be expected following afforestation).

Results of these tests, summarised in Table 3.4, show how sensitive the catchment water yield (particularly in the summer months) is to the size of the interception storage in this area of frequent rainfall. They also show how insensitive the catchment yield is to changes in the transpiration properties of the crop.

3.5 Model parameters and catchment characteristics

Because we have only applied the general model to a handful of catchments so far, it is impossible to find even tentative relationships between basin features and parameters. Table 3.5 does, however, list values obtained by fitting the model to five catchments, and suggests a few physical characteristics of these catchments to which the parameter values might be related.

If we are intent on making a real contribution in the direction of parameter value prediction, much work remains to be done on applying the model to a large number of very diverse catchments. Only when these data have been assembled will it be possible to estimate the reliability with which the hydrograph might be synthesised for an ungauged catchment.

3.6 Summary

This section of the report has been kept deliberately brief. It does not attempt to do more than outline the progress made to date, and the possibilities for the future, of applications of the general hydrological model. More detailed accounts of aspects of this work have been, or will be, reported elsewhere.

However it does illustrate the types of problem that the computer package described in Section 1 and the general model described in section 2 have been used to solve.

Table 3.4 Long term effects of afforestation on catchment water yield

	1	2	3	4	5
	Optimised model (pasture conditions)	increasing interception			increased transpiration
SS (mm)	1.2	2.0	4.0	8.0	2.0
FS	1.0	3.0	3.0	3.0	3.0
DCS (mm)	10.0	10.0	10.0	10.0	400.0
FC	0.7	0.7	0.7	0.7	0.7
mean annual flow (mm)	2329	2168	2019	1850	2150
mean summer flow (mm) (May-September)	773	708	600	468	690
Reduction in annual flow (%)	-	6.9*	13.3*	20.6*	0.8+
Reduction in summer flow (%)	-	8.4*	22.4*	39.4*	2.5+

* Reduction compared to Run 1

\+ Reduction compared to Run 2

Table 3.5 Parameter values and catchment characteristics

		Coalburn	Severn	Wye	Ray	Cam
SS		3.5	1.4	0.4	9.0	9.5
FS		1.0	4.8	1.0	1.0	1.5
FC		0.60	0.59	0.59	0.72	0.60
DCS		43.	3.	85.	50.	17.
Land* use	(urban	-	-	-	-	5%
	(arable	-	-	-	16%	70%
	(pasture	-	15%	92%	64%	18%
	(trees	-	70%	1%	20%	7%
	(peat	100%	15%	7%	-	-
RC		1.0	0.72	0.80	0.80	0.24
RS		0.035	0.037	0.037	0.033	0.029
GLR		-	1.4	1.4	-	0.14
Soils	(permeable	-	-	-	-	46%
	(medium	-	-	-	-	54%
	(impermeable	100%	100%	100%	100%	-
RK		0.052	0.083	0.083	0.047	0.22
RX		1.68	1.54	1.54	1.60	1.0
RDEL		0.6	0.5	0.5	2.2	4.6
Catchment area (km^2)		1.5	8.7	10.6	18.6	198.
Slope index		25.5	63.5	37.2	4.8	2.2
GSM		-	44	44	-	560
Bedrock		Carbonif. Limestone /Shales	Silurian/ Ordovician Shales		Oxford clay	Chalk

* approximate figures

Although applications described in this report have all been on the Institute's research catchments, there is reason to believe that the extension to other basins, for example, those maintained by the various Water Authorities, will be quite straightforward. It is anticipated that this model will prove a useful tool for the hydrological study of most basins of, say, 250 km^2 or less, provided that reliable estimates of precipitation and potential evaporation are available. Although most applications of the model at the Institute have used a 3 hourly modelling interval, it is anticipated that a daily model would be more useful in many studies. The model has, therefore, been designed to accept data at intervals ranging from 3 hours to 1 day. In trials shortly to be started, the model will be fitted to hydrographs from several Water Authority catchments, using precipitation and meteorological data provided by the Met. Office.

4 REFERENCES

Aitken, A. P. 1973. Assessing systemmatic errors in rainfall-runoff models. *J. Hydrol., 20*, 121-136

Clarke, R. T. 1973. Mathematical models in hydrology. Irrig. and drainage paper 19, FAO Rome pp 153-154

Dickinson, W. T. and Douglas, J. R. 1972. A conceptual runoff model for the Cam catchment. *Inst Hydrol Report No 17*

Eagleson, P. S. 1970. Dynamic Hydrology. McGraw-Hill. 462p.

Ibbit, R. P. 1970. Systemmatic parameter fitting for conceptual models of catchment hydrology. Unpubl. Ph D. thesis, Civil Engn. Dept, Imperial College, London

Mandeville, A. N., O'Connell, P. E., Sutcliffe, J. V. and Nash, J. E. 1970. River flow forecasting through conceptual models. Pt III - The Ray catchment at Grendon Underwood. *J. Hydrol., 11*, 109-128

McCarthy, G. T. 1938. The unit hydrograph and flood routing. Unpubl. manuscript, North Atlantic Division, Corps. of Engineers, U.S. War Dept.

Nash, J. E. and Sutcliffe, J. V. 1969. Flood wave formation. Hydrol. forecasting. Tech. note 92 WMO. 147-155

Penman, H. L. 1948. Natural evaporation from open water, bare soil and grass. *Proc. roy. Soc. Lond. (A) 193*, 120-195

Plinston, D. T. 1971. Parameter sensitivity and interdependence in hydrological models. *Brit. Ecol. Soc. 12th Symposium, Grange-over-Sands, England 237-247*

Rosenbrock, H. H. 1960. An automatic method of finding the greatest or least value of a function. *Comp. J. 3*, 175-184

Stewart, J. B. and Thom, A. S. 1973. Energy budgets in pine forest. *Quart J. R. met. Soc. 99*, 154-170

U.S. Army, Corps. of Engineers. 1956. Snow Hydrology. Portland, Oreg. North Pacific Division, Corps of Engineers, 437 p

5. APPENDIX - LISTING OF PROGRAM

There follows a listing of the MASTER routine, and subroutines CONTROL, OPTION and AUG, which were described earlier in Section 1.7.
This version of the program was developed for use on an ICL 1900 series computer, using the GEORGE 3 operating system. Minor modifications have since been made to enable the program package to run under the Exec 8 operating system on a Univac 1108 computer.

```
FORTRAN COMPILATION BY #XFIH MK 1C    DATE  16/01/74  TIME  21/21/05

          LIST
          PROGRAM(JRDX)
          INPUT 10 = CR0
          INPUT 4 = MT1/UNFORMATTED(COARTQE)
          OUTPUT 1 = LP0
          OUTPUT 6 = MT2/UNFORMATTED(RESIDS)
          END



          MASTER CONMODS
          CALL CONTROL
          END
```

```
      SUBROUTINE CONTROL
      COMMON/B1/II(7),JJ(15),MM(5)
      COMMON/B2/NOPS,NITS,NTIM,IFIN,NM,IFREQ
C ***THIS ROUTINE READS DATA CARDS WHICH CONTROL THE WAY THE
C    MODELLING PACKAGE IS USED, AS WELL AS DESCRIBING THE
C    MODEL IN USE, THE CATCHMENT AND THE DURATION AND FREQUENCY
C    OF THE DATA.
C ***THE ARRAY 'JJ' CONTAINS DETAILS OF MODEL,CATCHMENT AND DATA
C          ELEMENTS 1-5 (CARD 1,COLS 1-20) DESCRIBE THE MODEL
C          ELEMENTS 6-10(CARD 2,COLS 1-20) GIVES THE CATCHMENT
C          ELEMENTS 11-15(CARD3,COLS 1-20) GIVES THE DURATION OF
C                                                  THE DATA
      NOPS=0
    1 READ(10,100,END=99)(JJ(K),K=1,15)
      NOPS=NOPS+1
C ***THE ARRAY 'II' CONTAINS THE VALUES OF VARIOUS INDICES, USED
C    TO CONTROL THE MODE OF OPERATION, INPUT AND OUTPUT, ETC.
C          ELEMENT 1 CONTROLS THE TYPE OF ANALYSIS:
C    (CARD 4,COL 4) VALUE 1 FOR OPTIMISATION
C                   VALUE 2 FOR PREDICTION
C                   VALUE 3 FOR SENSITIVITY/INTERDEPENDENCE
C                   VALUE 4 FOR SURFACE MAPPING
C                   VALUE 5 FOR INITIAL VARIANCE
C          ELEMENT 2 CONTROLS THE READING OF HYDROLOGICAL DATA
C           (COL 8)    VALUE 0 CAUSES NO DATA TO BE READ (THIS MAY BE
C                              USED WHEN DATA HAS ALREADY BEEN READ BY
C                              AN EARLIER PART OF THE RUN.)
C                   VALUE 1 CAUSES HYDROLOGICAL DATA TO BE READ
C          ELEMENT 3 CONTROLS THE READING OF PARAMETER VALUES
C           (COL 12)   VALUE 1 INDICATES NEW PARAMETER VALUES
C                   VALUE 2 CAUSES PREVIOUS OPTIMUM VALUES TO BE USED
C                   VALUE 3 INDICATES NO PARAMETER VALUES TO BE ASSIGNED
C          ELEMENT 4 CONTROLS THE DETAIL OF LINEPRINTER OUTPUT
C           (COL 16)  VALUE 0 FOR SUMMARY, 1 OR 2 FOR MORE DETAIL
C          ELEMENT 5 CONTROLS GRAPHICAL OUTPUT
C           (COL 20)   VALUE 0 FOR NO GRAPHS, OTHERWISE UP TO USER
C          ELEMENT 6 (USED FOR OPTIMISATION RUNS ONLY) GIVES THE MAXIMUM
C     (COLS 21-24) NUMBER OF ITERATIONS TO BE PERFORMED DURING OPT'N.
      READ (10,101)(II(K),K=1,7)
C ***THE ARRAY 'MM' CONTAINS INFORMATION ABOUT THE DATA
C          ELEMENT 1 GIVES FREQUENCY (READINGS/DAY) OF DATA SOURCE
C          ELEMENT 2 GIVES DESIRED TIME INTERVAL OF MODEL (READINGS/DAY)
C          ELEMENT 3 GIVES JULIAN DAY NUMBER AT START OF DATA
C          ELEMENT 4 GIVES THE NUMBER OF MONTHS TO BE MODELLED
C          ELEMENT 5 GIVES THE NUMBER OF THE STARTING MONTH
      READ(10,102)(MM(K),K=1,5)
      WRITE(1,200) JJ,MM(2)
      IFREQ=MM(2)
      NM=MM(4)
      CALL OPTION
      WRITE(1,201)
      GO TO 1
   99 STOP
  100 FORMAT((5A4))
  101 FORMAT(7I4)
  102 FORMAT(5I4)
  200 FORMAT(1H1 /// 2( 5X,60(1H*)/), 5X,3H***,16X,22HINSTITUTE OF HYDRO
     1LOGY,16X,3H***/ 5X,3H***,13X,28HCONCEPTUAL MODELLING PACKAGE,13X,
     23H***/ 5X,3H***,54X,3H***/ 5X,3H***,12X,18HMODEL BEING USED: ,5A4,
     34X,3H***/ 5X,3H***,10X,20HLOCATION/CATCHMENT: ,5A4,4X,3H***/ 5X,
     43H***,12X,18HDURATION OF DATA: ,5A4,4X,3H***/ 5X,3H***,14X,16HDATA
     5 FREQUENCY: ,I2,19H READING(S) PER DAY,3X,3H***,2(/ 5X,60(1H*))//)
  201 FORMAT( // 30X,19HMODEL RUN COMPLETED)
      END
```

```
      SUBROUTINE OPTION
      COMMON/B1/II(7),JJ(15),MM(5)
      COMMON/B2/NOPS,NITS,NTIM,IFIN,NM,IFREQ
      COMMON/B3/N,NN,PNOM(20),Y(20),YI(50),IQUIV(50,20),F
      COMMON/B4/OBSOUT(250)
      DIMENSION A(20,20),B(20),C(20),CH(20),DP(20),FN(2),FUN(5,5)
     1,PM(5,10),PP(5),QQ(5),X(20),XK(20),XOLD(20),Z(3,20),IP(5)
     2,MN(5),P(5),KK(50),NAMES(50)
C
C **INITIATE COUNTS AND INDICES
C
      NTIM=0
      IFIN=0
      NITS=0
C
C **INPUT OF PARAMETERS, AS REQUIRED, FIRST CARD GIVES THE NUMBER OF
C   PARAMETERS, SUBSEQUENT CARDS GIVE PARAMETER NAME, STARTING VALUE,
C   AND, FOR OPTIMISATION, THE LOWER AND UPPER LIMITS OF SEARCH.
      I=II(3)
      GO TO (901,902,905),I
  901 READ(10,100) N,NN
      DO 801 J=1,NN
      DO 801 K=1,N
      X(K)=0.0
      IQUIV(J,K)=0
  801 CONTINUE
      WRITE(1,203)
      GO TO 903
  902 WRITE(1,204)
  903 WRITE(1,200)
      DO 904 J=1,NN
      IF (I.EQ.1) READ(10,103) NAMES(J),KK(J),YI(J),BB,CC
      K=KK(J)
      IF (K.EQ.0) GO TO 804
      IF (I.EQ.2) GO TO 802
      CALL COPY8(PNOM(K),NAMES(J))
      Y(K)=YI(J)
      B(K)=BB
      C(K)=CC
      IQUIV(J,K)=1
  802 IF(II(1).EQ.1)X(K)=ASIN(SQRT((Y(K)-B(K))/(C(K)-B(K))))
      IF (II(4).GE.1) WRITE (1,221) PNOM(K),B(K),C(K),Y(K),K
      GO TO 904
  804 WRITE (1,222) NAMES(J),YI(J)
  904 CONTINUE
C
C **SELECT MODE OF OPERATION
C
  905 I=II(1)
      GO TO (1,30,40,60,80),I
C
C **ROSENBROCK OPTIMISATION
C   ***********************
    1 WRITE(1,220) II(6)
          DO 3 I=1,N
          Z(1,I)=0.04
          Z(2,I)=1.0
          Z(3,I)=X(I)
              DO 2 J=1,N
    2         A(J,I)=0.0
    3     A(I,I)=1.0
      DO 907 J=1,N
  907 Y(J)=B(J)+(C(J)-B(J))*SIN(X(J))**2
      CALL MODEL
      F1=F
    4     DO 11 L=1,N
          AA=0.5*Z(1,L)
          BB=0.0
          CALL AUG (AA,X,Z,A,L)
      DO 908 J=1,N
  908 Y(J)=B(J)+(C(J)-B(J))*SIN(X(J))**2
      CALL MODEL
          IF (F1.GT.F) GO TO 6
          CALL AUG (-2.*AA,X,Z,A,L)
          F2=F
```

```
      DO 909 J=1,N
  909 Y(J)=B(J)+(C(J)-B(J))*SIN(X(J))**2
      CALL MODEL
          IF (F1.GT.F) GO TO 5
          CALL AUG (AA,X,Z,A,L)
          TOP=F-F2
          BOT=2.0*(F+F2-2.0*F1)
          GO TO 7
 5              AA=-AA
 6              F2=F1
                F1=F
                BB=BB+AA
                AA=1.5*AA
      DO 910 J=1,N
  910 XK(J)=X(J)
                CALL AUG (AA,X,Z,A,L)
      DO 911 J=1,N
  911 Y(J)=B(J)+(C(J)-B(J))*SIN(X(J))**2
      CALL MODEL
                IF (F1.GT.F) GO TO 6
                CALL AUG (-AA,X,Z,A,L)
                AA=AA/1.5
                Z(1,L)=AA
                BOT=3.0*F2-5.0*F1+2.0*F
                TOP=2.25*F2-1.25*F1-F
 7        IF (BOT.LE.1.E-10) GO TO 9
          D=AA*TOP/BOT
          CALL AUG (D,X,Z,A,L)
      DO 912 J=1,N
  912 Y(J)=B(J)+(C(J)-B(J))*SIN(X(J))**2
      CALL MODEL
          IF (F.GT.F1) GO TO 8
      DO 913 J=1,N
  913 XK(J)=X(J)
          Z(1,L)=AA+D
          F1=F
          BB=BB+D
          GO TO 10
 8              CALL AUG (-D,X,Z,A,L)
 9              IF (BB.NE.0.0) GO TO 10
                BB=AA/10.0
 10       DO 11 J=1,N
 11       A(J,L)=BB*A(J,L)
C
C
C **END OF ITERATION - CHECK PROGRESS OF OPTIMISATION
C
      IF (N.EQ.20) GO TO 914
      NNB=N+1
      DO 12 M=NNB,20
      CH(M)=0.0
   12 X(M)=0.0
  914 NITS=NITS+1
      WRITE (1,201) NITS,NTIM,F1
      DO 13 M=1,N
      IF (NITS.GT.1) CH(M)=ABS(XOLD(M)-XK(M))
   13 XOLD(M)=XK(M)
      IF (NITS.EQ.1) GO TO 916
      XMAX=AMAX1(CH(1),CH(2),CH(3),CH(4),CH(5),CH(6),CH(7),CH(8),CH(9),
     1CH(10),CH(11),CH(12),CH(13),CH(14),CH(15),CH(16),CH(17),CH(18),
     2CH(19),CH(20))
      IF (XMAX.LT.1.E-3) GO TO 14
  916 IF (NITS.LT.II(6)) GO TO 15
 14   IFIN=2
   15 IF (IFIN.LT.2) GO TO 17
C
C ** END OF OPTIMISATION - RUN OPTIMUM IN PREDICTION MODE
C
      DO 16 J=1,N
 16   Y(J)=B(J)+(C(J)-B(J))*SIN(XK(J))**2
      CALL MODEL
      RETURN
C
C ** RE-ORIENTATE AXES AND CONTINUE OPTIMISATION
C
   17 DO 18 J=2,N
                L=N-J+1
                DO 18 I=1,N
 18             A(I,L)=A(I,L)+A(I,L+1)
                DO 23 J=1,N
                IF (J.EQ.1) GO TO 21
                I=J-1
                DO 20 L=1,I
                AA=0.0
```

```
                  DO 19 K=1,N
   19             AA=A(K,L)*A(K,J)+AA
                  DO 20 K=1,N
   20             A(K,J)=A(K,J)-AA*A(K,L)
   21             AA=0.0
                  DO 22 K=1,N
   22             AA=A(K,J)**2+AA
                  AA=1.0/SQRT(AA)
                  DO 23 K=1,N
   23             A(K,J)=AA*A(K,J)
                  GO TO 4
C
C **HYDROGRAPH PREDICTION**
C
   30 IFIN=2
      WRITE(1,218)
      CALL MODEL
      RETURN
C
C **SENSITIVITY ROUTINE**
C
   40 IF (N.GT.3) N=3
      READ (10,101) (DP(I),I=1,N)
      WRITE(1,205)(PNOM(I),DP(I),I=1,N)
      DO 915 J=1,N
  915 P(J)=Y(J)
      CALL MODEL
      FOPT=F
C
      NNC=N-1
      DO 46 I=1,NNC
      DO 46 M=1,NNC
      MI=M+I
      IF (MI.EQ.4) GO TO 46
      DO 917 J=1,N
  917 Y(J)=P(J)
      DO 41 K=1,5
      QQ(K)=P(I)+(K-3)*DP(I)
   41 PP(K)=P(MI)+(K-3)*DP(MI)
      DO 43 J=1,5
      Y(I)=P(I)+(J-3)*DP(I)
      DO 42 K=1,5
      IF (J.GT.1.OR.K.GT.1) ICS=0
      Y(MI)=P(MI)+(K-3)*DP(MI)
      IF ((K.EQ.1.OR.K.EQ.5).AND.(J.EQ.2.OR.J.EQ.4)) GO TO 42
      IF ((K.EQ.2.OR.K.EQ.4).AND.(J.EQ.1.OR.J.EQ.5)) GO TO 42
      IF (I.EQ.1.AND.M.EQ.2.AND.K.EQ.3) GO TO 42
      IF (I.EQ.2.AND.J.EQ.3) GO TO 42
      CALL MODEL
      FUN(J,K)=F-FOPT
   42 CONTINUE
   43 CONTINUE
      WRITE (1,206) PNOM(MI),(PP(K),K=1,5),PNOM(I)
      WRITE (1,207) ((QQ(J),(FUN(J,K),K=1,5)),J=1,5)
      WRITE (1,208)
      FN(2)=(-FUN(3,5)+FUN(3,4)*16.0-FUN(3,3)*30.0+FUN(3,2)*16.0
     1         -FUN(3,1))
      FN(1)=(-FUN(5,3)+FUN(4,3)*16.0-FUN(3,3)*30.0+FUN(2,3)*16.0
     1         -FUN(1,3))
      WRITE (1,209) PNOM(I),FN(1),PNUM(MI),FN(2)
      FNB=SQRT(FN(2)/FN(1))
      FNC=2.0*SQRT(FN(2)*FN(1))
C
      FN(1)=(-FUN(5,5)+FUN(4,4)*16.0-FUN(3,3)*30.0+FUN(2,2)*16.0
     1         -FUN(1,1)) / 2.0
      FN(2)=(-FUN(5,1)+FUN(4,2)*16.0-FUN(3,3)*30.0+FUN(2,4)*16.0
     1         -FUN(1,5)) / 2.0
      WRITE (1,211) FN(1),FN(2)
      FNA=(FN(2)-FN(1))/FNC
      WRITE (1,219) FNA,FNB
   46 CONTINUE
      RETURN
```

```
C
C **MAPPING ROUTINE**
C
   60 IF (N.GT.5) N=5
      WRITE(1,216) N
      DO 61 I=1,5
      MN(I)=1
   61 CONTINUE
      DO 62 I=1,N
      READ (10,102) MNI,(PM(I,M),M=1,MNI)
      MN(I)=MNI
   62 CONTINUE
      MN1=MN(1)
      DO 64 IP1=1,MN1
      MN2=MN(2)
      DO 66 IP2=1,MN2
      MN3=MN(3)
      DO 64 IP3=1,MN3
      MN4=MN(4)
      DO 64 IP4=1,MN4
      MN5=MN(5)
      DO 66 IP5=1,MN5
      IP(1)=IP1
      IP(2)=IP2
      IP(3)=IP3
      IP(4)=IP4
      IP(5)=IP5
      DO 63 NP=1,N
      Y(NP)=PM(NP,IP(NP))
   63 CONTINUE
      CALL MODEL
   64 CONTINUE
      RETURN
C
C *** INITIAL VARIANCE CALCULATIONS ***
C
   80 F=0.0
      SOUT=0.0
      NPER=0
      IM=MM(5)
      NTIM=1
      WRITE(1,217)
   81 CALL READER(IM,IDAYS)
      ID=IDAYS*IFREQ
      DO 82 I=1,ID
      SOUT=SOUT+OBSOUT(I)
   82 CONTINUE
      NPER=NPER+ID
      IF (IM.GE.NM) GO TO 83
      IF (II(4).GE.2) WRITE(1,212) IM,SOUT,NPER
      IM=IM+1
      GO TO 81
   83 AOUT=SOUT/NPER
      WRITE(1,215)NM,SOUT,AOUT
      IM=MM(5)
      NTIM=2
   84 CALL READER(IM,IDAYS)
      ID=IDAYS*IFREQ
      DO 85 I=1,ID
      F=F+(OBSOUT(I)-AOUT)**2
   85 CONTINUE
      IF (IM.EQ.NM) GO TO 86
      IF (II(4).GE.1) WRITE(1,213) IM,F
      IM=IM+1
      GO TO 84
   86 WRITE(1,214) F
      RETURN
```

```
C
C ** FORMAT STATEMENTS **
C
  100 FORMAT(2I4)
  101 FORMAT(3F10,0)
  102 FORMAT(I2,9F8,0)
  103 FORMAT(A8,I2,3F10.0)
  200 FORMAT(1H /11X,44HLOWER       UPPER      PARAMETER      OPTIMISING  /
     111X,43HLIMIT       LIMIT         VALUE          SEQUENCE  /  )
  201 FORMAT(1H /21X,I4,37H ITERATIONS OF OPTIMISATION COMPLETED / 21X,
     1I4,29H FUNCTION CALCULATIONS SO FAR / 23X,21HFUNCTION VALUE IS NOW
     2,F15.8 / 23X,36(1H*) // )
  203 FORMAT(1H0,10X,24H**NEW PARAMETER VALUES** )
  204 FORMAT(1H0,10X,27H**PREVIOUS OPTIMUM VALUES** )
  205 FORMAT( // 11X,38HPARAMETER INTERDEPENDENCE CALCULATIONS / 11X,
     138(1H*) // 15X,21HPARAMETRIC INTERVALS( // 3(20X,A4,F10,5/))
  206 FORMAT( /  25X,20(1H*) // 11X,58HERROR FUNCTION MATRIX  (DIFFERENC
     1ES FROM SUPPOSED OPTIMUM) // 12X,A4,1X,5(F9,5,2X) // 3X,A4)
  207 FORMAT(1H0,5(/F10,5,7X,5(F9,5,2X)))
  208 FORMAT( // 31X,20(1H*) // 21X,18HSECOND DERIVATIVES )
  209 FORMAT(1H0,20X,3HIN ,A4,12H DIRECTION =,F16,8 / 21X,3HIN ,A4,
     112H DIRECTION =,F16,8 )
  211 FORMAT(1H0,13X,24HIN DIAGONAL DIRECTIONS =,F16,8 / 38X,F16,8 )
  212 FORMAT(1H ,5X,8HBY MONTH,I3,23H  ACCUMULATED OUTPUT IS,F10,2, 5H
     1IN ,I5,9H PERIODS,)
  213 FORMAT(1H ,5X,8HBY MONTH,I3,25H  ACCUMULATED VARIANCE IS,F10.2)
  214 FORMAT(1H0,5X,37HTOTAL VARIANCE FOR COMPLETE RECORD IS,F12,3 /
     16X,49(1H*) // )
  215 FORMAT(1H0,5X,7HFOR ALL,I3,24H MONTHS, TOTAL OUTPUT IS,F10.2/15X,
     127H  MEAN OUTPUT PER PERIOD IS,F8,4 ///)
  216 FORMAT(1H0,10X,13HMAPPING OF A ,I1,32H-DIMENSIONAL ERROR FUNCTION
     1GRID / 11X,46(1H*) )
  217 FORMAT(1H0,10X,39HINITIAL (NO-MODEL) VARIANCE CALCULATION / 11X,
     139(1H*) // )
  218 FORMAT(1H0,10X,16HMODEL PREDICTION / 11X,16(1H*) // )
  219 FORMAT(1H0,20X,6HHENCE; / 24X,24HINTERDEPENDENCE VALUE IS,F8.5 /
     114X,45HRELATIVE SENSITIVITIES AT THESE STEP SIZES IS,F10,4 // )
  220 FORMAT(1H /// 58H OPTIMISATION (USING ROSENBROCKS METHOD) WITH NO
     1MORE THAN,I3,11H ITERATIONS /  1X,71(1H*) // )
  221 FORMAT (1H ,A8,F9,4,F10.4,F12,4,9X,I2 )
  222 FORMAT(1H ,A8,19X,F12,4)
C
      END
```

```
      SUBROUTINE AUG (G,X,Z,A,L)
      COMMON/B3/N,NN,PNUM(20),Y(20),YI(50),IQUIV(50,20),F
C
      DIMENSION X(20),A(20,20),Z(3,20)
C
      DO 1 I=1,N
      Z(2,I)=Z(2,I)+A(I,L)*G
    1 X(I)=Z(2,I)+Z(3,I)
      RETURN
      END
```