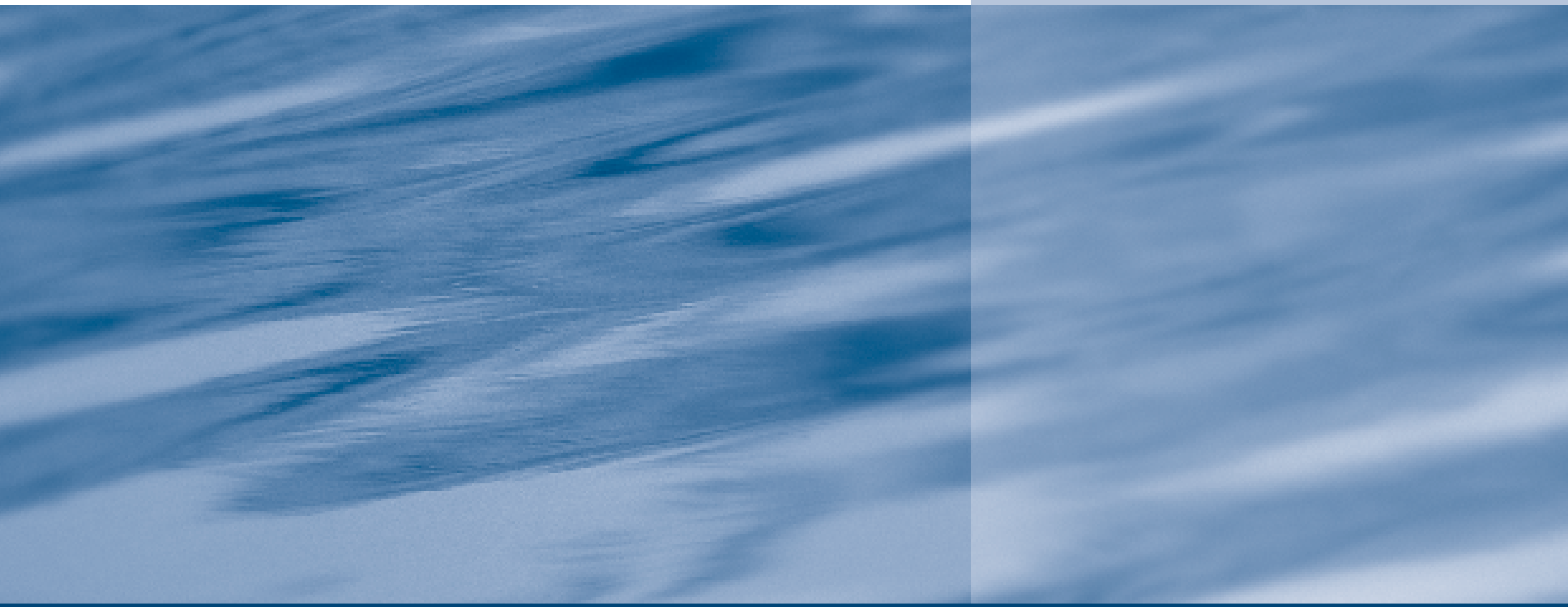


www.environment-agency.gov.uk

Enhancements to Modflow-User Guide for Modflow VKD–A modified version of modflow 96 to include variations in hydraulic properties with depth

National Groundwater & Contaminated Land Centre
Project NC/00/23

ENVIRONMENT AGENCY

The Environment Agency is the leading public body protecting and improving the environment in England and Wales.

It's our job to make sure that air, land and water are looked after by everyone in today's society, so that tomorrow's generations inherit a cleaner, healthier world.

Our work includes tackling flooding and pollution incidents, reducing industry's impacts on the environment, cleaning up rivers, coastal waters and contaminated land, and improving wildlife habitats.

This report is the result of research commissioned and funded by the Environment Agency's Science Programme.

Published by:
Environment Agency, Rio House, Waterside Drive, Aztec West, Almondsbury, Bristol, BS32 4UD
Tel: 01454 624400 Fax: 01454 624409
www.environment-agency.gov.uk

ISBN: 1 84432 352 8

This report is printed on Cyclus Print, a 100% recycled stock, which is 100% post consumer waste and is totally chlorine free. Water used is treated and in most cases returned to source in better condition than removed.

Further copies of this report are available from:
The Environment Agency's National Customer Contact Centre by emailing enquiries@environment-agency.gov.uk or by telephoning 08708 506506.

Author(s):
Water Mangement Consultants Ltd

Dissemination Status:
Publicly available

Keywords:
MODFLOW, VKD, modelling

Research Contractor:
Water Management Consultants Ltd
23 Swan Hill Shrewsbury
Shropshire
SY1 1NN
Tel: 01743 231793 Fax: 01743 232894

Environment Agency's Project Manager:
Paul Hulme

Project Board:
Paul Hulme, Paul Shaw, Andrew Longley, David Johnson

Science Project Number:
NC/00/23

Product Code:
SCHO0903BILY-E-P

Science at the Environment Agency

Science underpins the work of the Environment Agency, by providing an up to date understanding of the world about us, and helping us to develop monitoring tools and techniques to manage our environment as efficiently as possible.

The work of the Science Group is a key ingredient in the partnership between research, policy and operations that enables the Agency to protect and restore our environment.

The Environment Agency's Science Group focuses on five main areas of activity:

- **Setting the agenda**: To identify the strategic science needs of the Agency to inform its advisory and regulatory roles.
- **Sponsoring science**: To fund people and projects in response to the needs identified by the agenda setting.
- **Managing science**: To ensure that each project we fund is fit for purpose and that it is executed according to international scientific standards.
- **Carrying out science**: To undertake the research itself, by those best placed to do it - either by in-house Agency scientists, or by contracting it out to universities, research institutes or consultancies.
- **Providing advice**: To ensure that the knowledge, tools and techniques generated by the science programme are taken up by relevant decision-makers, policy makers and operational staff.

Professor Mike Depledge Head of Science

ENHANCEMENTS TO MODFLOW

USER GUIDE FOR MODFLOW-VKD –
A MODIFIED VERSION OF MODFLOW-96
TO INCLUDE VARIATIONS IN HYDRAULIC PROPERTIES WITH DEPTH

September 2003

User Guide-v24.DOC

(1621/R3/)

Prepared for:

National Groundwater and Contaminated Land Centre
Environment Agency
Olton Court
10 Warwick Road
Olton
Solihull B92 7HX
UK

Prepared by:

Water Management Consultants Ltd
23 Swan Hill
Shrewsbury
Shropshire SY1 1NN
UK

CONTENTS

TABLES

FIGURES

APPENDICES

GLOSSARY OF TERMS AND ABBREVIATIONS

Symbol	Description	Variable name in MODFLOW	Dimensions
BAS	Basic	Package: BAS	-
BCF	Block Centred Flow	Package: BCF	-
B_u	Thickness of upper part of VKD layer	VMID (first simulation when auto-conversion is active)	L
CV, CR, CC	Conductance terms	CV, CR, CC	L^2T^{-1}
E_{Bot}	Elevation of the base of the model layer	BOT	L
E_{Kmax}	Elevation where the hydraulic conductivity reaches its maximum value	EKMAX	L
E_{Mid}	Elevation of the change from constant to varying hydraulic conductivity (point of inflection)	VMID	L
E_{Top}	Elevation of the top of the model layer	TOP	L
f	Hydraulic conductivity gradient factor	VKGRAD	L^{-1}
F_s	Storage gradient factor	VSGRAD	L^{-1}
H	Groundwater head	HNEW (current head) HOLD (head in last time step)	L
HCOF	Head coefficient – contains terms for head dependant boundaries	HCOF	L^2T^{-1}
i, j, k	Row, column and layer numbers	I, J, K	-
K	Hydraulic conductivity	HY	LT^{-1}
K_{base}	Hydraulic conductivity of the lower part of a VKD layer in MODFLOW-VKD	HY (same as normal hydraulic conductivity)	LT^{-1}
K_{max}	Maximum hydraulic conductivity factor	VKMAX	-
LAYCON	Layer-type code	LAYCON	-

Symbol	Description	Variable name in MODFLOW	Dimensions
RHS	"Right hand side" – contains terms for sources/sinks and head dependant boundaries	RHS	L^3T^{-1}
S	Storage coefficient	SC1 or SC2	-
S_{base}	Storage coefficient of the lower part of a VSD layer	SC2 (or SC1 depending on value of LAYCON)	-
S_{max}	Maximum storage factor	VSMAX	-
S_s	Specific storage	(becomes SC1 when integrated over saturated thickness)	L^{-1}
Sy	Specific yield	SC2 (or SC1 depending on value of LAYCON)	-
t	Time	TOTIM	T
T	Transmissivity	- (converted to conductance and stored in CC and CR)	L^2T^{-1}
VKD	Variable hydraulic conductivity with depth	-	-
VSD	Variable storage with depth	-	-
W	Source/sink term (flow per unit volume)	(stored in RHS and HCOF)	T^{-1}
x, y, z	Cartesian co-ordinates	-	L
Δt	Time step length	DELT	T

1 INTRODUCTION

The modified version of the MODFLOW groundwater modelling code described in this user guide is based on the 1996 version of the USGS code: MODFLOW-96 (Harbaugh & McDonald, 1996), which in turn was based on an earlier version of the code: MODFLOW-88 (McDonald & Harbaugh, 1988).

This is a revised version of the User Guide originally issued in March 2002. The revised version includes some minor changes to the MODFLOW VKD code (currently MF-VKD7), the results of some further code testing, and an independent peer review of the code and documentation.

The main focus of the modifications to MODFLOW has been to include a representation of the variation of hydraulic conductivity and storage with depth commonly observed in chalk and limestone aquifers. These modifications provide a way to represent in MODFLOW the behaviour of groundwater in fractured systems, such as chalk and limestone formations. In such systems fissure frequency is high near to the surface thus enhancing hydraulic conductivity, specific yield and the hydraulic connection between the aquifer and surface water. The fracture frequency, and hence conductivity and specifc yield, reduce considerably with depth.

Traditionally in the UK, chalk and limestone aquifers have been represented in groundwater models as a thin layer representing only the 'active flow zone', which is usually assumed to be between 30 and 60 m thick, with no variation of hydraulic conductivity with depth. These constant hydraulic conductivity models give rise to a linear variation of transmissivity with depth, which is a poor approximation to the observed transmissivity variation. The limited representation of the conditions in the aquifer means that, in order to represent observed behaviour, the modeller may need to employ unrealistic or contrived values for other model parameters such as storage, river conductance, or temporal distribution of recharge. An additional problem with these types of MODFLOW models is that if a severe drought is being simulated, groundwater heads can fall below the normal active flow zone, resulting in cells becoming 'dry', and distorting the pattern of flow. If a low hydraulic conductivity layer is added below the active flow zone to enable 'wetting' of the layer above, the contrast in hydraulic conductivity between the two layers often results in numerical oscillations as cells change between wet and dry, reducing the likelihood that the model will converge.

The modification to MODFLOW where hydraulic properties are allowed to vary with depth within a single model layer provides a solution to these problems. This gives a non-linear relationship between transmissivity and groundwater level, improving the representation of field conditions. The underlying issues and implications of the changes introduced in MODFLOW are detailed in the Project Report NC/00/23 – *Enhancements to Modflow, variations in hydraulic conductivity and storage with depth* (Environment Agency, 2002).

The modifications to the MODFLOW code are based on a code developed at the University of Birmingham (UK) against which the modified MODFLOW code has been benchmarked (Environment Agency, 1999 & 2001). Other changes to the code have been made to aid comparison with the Birmingham University code, and to include a few useful new options.

The new features introduced in MODFLOW that are described in this User Guide can be grouped into those that relate to the implementation of variable hydraulic properties with depth within a layer (Group I) and others, which increase the flexibility and output options of the code (Group II). These features include:

Group I

a) Variation of horizontal hydraulic conductivity with depth (VKD).

b) Variation of storage coefficient with depth (VSD).

c) Auto-conversion option for converting standard MODFLOW-96 steady state models to include VKD and providing initial conditions for time variant simulations.

Group II

a) Spatially variable anisotropy (Ruskauff & Kladias).

b) Inter-nodal transmissivity option to allow greater control of model transmissivities and to allow direct comparison with mesh-centred codes.

c) Option to allow input of X- and Y-direction transmissivities independently.

d) Explicit transmissivity calculation option (for comparison with other codes), where transmissivity is updated every time step using heads from the previous time step.

e) Output of calculated transmissivity values to the listing file.

f) Allow use of direct access binary output files that can be read directly by Groundwater Vistas (Environmental Simulations Inc, 2001).

g) Progress monitor for the PCG solver package.

h) Allow convergence to be forced if the convergence criteria are met during the last inner iteration for a specified number of outer iterations (rather than during the first inner iteration as is usual). This is the same technique used to force convergence of the PCG solver in Groundwater Vistas.

i) Debugging option for the PCG solver.

j) Allow a specified discharge, abstraction or tributary inflow at any stream cell.

k) Correction to the cell-by-cell stream flow output, so that where more than stream reach is defined in a model cell, only the stream flow from the furthest downstream reach is used to compute the accreted stream flow at that point.

MODFLOW is the ‘Modular Three-Dimensional Finite-Difference Ground-Water Flow Model’ developed by the United States Geological Survey (McDonald & Harbaugh, 1988, Harbaugh & McDonald, 1996). The model is made up of a number of different modules (or packages) that handle different aspects of groundwater simulations, and a main program that handles the calls to the different routines in each of the modules.

The modifications undertaken for the implementation of variable hydraulic conductivity (VKD) and specific yield (VSD) with depth within a layer required changes to the main program and Block-Centred Flow (BCF) module, which defines the conductance and storage terms of the groundwater flow equations, based on the aquifer properties. This leads to changes or new options in the format of the BCF (and name) files required by MODFLOW. The modifications to the code and input files are described in Sections 2, 6 and 8. Section 8 also provides an introduction to a practical application of the code for modelling groundwater systems where parameters vary with depth.

The stream package, a module used to simulate external influences on the groundwater system due to surface water interactions, was also modified as part of this project. The changes to the stream package are outlined in Section 3.

Once the terms of the finite difference groundwater flow equations have been defined (by the BCF and other packages), a number of different solver modules are available to solve these equations. Some modifications were also made to the output options of one of these solver modules; namely, the Preconditioned Conjugate Gradient 2 (PCG2) module. These changes are documented in Section 4.

The Utilities module was also modified slightly. This module contains routines to read data from input files, and write data to output files. The modifications introduced to the Utilities module are described in Section 5.

Finally, a guide to compiling the code and post processing utilities for VKD MODFLOW are presented in Section 7 and 9, respectively.

Appendices A-E list the computer code of the MODFLOW packages where modifications were introduced.

Appendix F describes further testing for MODFLOW VKD (MF-VKD), which was carried out to ascertain that the VKD modifications do not affect the basic functionality of MODFLOW. A series of tests is presented, where the results generated by the unmodified version of MODFLOW 96 (MF96) and MODFLOW-VKD (MF-VKD) with the VKD function active and inactive, are compared to identify whether and why significant discrepancies emerge. The testing is based on a suite of models developed by the US EPA, the USGS, and two models developed for the Environment Agency, the former aiming at water resource assessment in the Wirral peninsula and the latter at the definition of groundwater protection zones for the Tadcaster brewery boreholes. In the test based on the Tadcaster model, MF96 and MF-VKD are coupled with MODPATH to check whether the two programs produce different particle-tracking results. A variation on one of the original test models used for testing MODFLOW-VKD is also employed to run the USGS optimisation and sensitivity analysis program, UCODE, in combination with MODFLOW-VKD. The purpose of this test is to check whether and how UCODE works with MF-VKD. In two of the tests, the results produced by unmodified MODFLOW and the executable of MF-VKD created with the FORTRAN Compaq compiler were compared to check that the use of a different compiler would not affect the model output. The modifications to the input datasets that are required to run MF-VKD and the results of each test are documented in a series of summary sheets. The implications of coupling MF-VKD with MODPATH to perform particle tracking are also discussed.

Appendix G is an independent peer review of the MODFLOW VKD code by Professor David Lerner, Groundwater Protection and Restoration Group, University of Sheffield.

Appendix H provides a summary of the MODFLOW VKD version history.

2 BLOCK-CENTRED FLOW (BCF) PACKAGE

2.1 Introduction

The Block-Centred Flow (BCF) package computes the conductance components of the finite-difference equation, which determines the flow between adjacent cells. It also computes the terms that determine the rate of movement of water to and from storage. To make the required calculations, it is assumed that the node is located at the centre of each model cell.

MODFLOW uses a finite difference form of the groundwater flow equation:

$$\frac{\partial}{\partial x}\left(K_{xx}\frac{\partial h}{\partial x}\right)+\frac{\partial}{\partial y}\left(K_{yy}\frac{\partial h}{\partial y}\right)+\frac{\partial}{\partial z}\left(K_{zz}\frac{\partial h}{\partial z}\right)-W=S_S\frac{\partial h}{\partial t}$$

Equation (2.1)

where

K_{xx}, K_{yy} and K_{zz} are values of hydraulic conductivity along the x, y and z coordinate axes, which are assumed to be parallel to the major axes of hydraulic conductivity [LT^{-1}];
h is the potentiometric head [L];
W is a volumetric flux per unit volume and represents sources and/or sinks of water [T^{-1}];
S_s is the specific storage of the porous material [L^{-1}]; and
t is time [T].

The finite difference form of equation (1) used in MODFLOW is (McDonald and Harbaugh, 1988):

$$CV_{i,j,k-½}\,h_{i,j,k-1} + CC_{i-½,j,k}\,h_{i-1,j,k} + CR_{i,j-½,k}\,h_{i,j-1,k}$$
$$+ (- CV_{i,j,k-½} - CC_{i-½,j,k} - CR_{i,j-½,k} - CR_{i,j+½,k} - CC_{i+½,j,k} - CV_{i,j,k+½} + HCOF_{i,j,k})\, h_{i,j,k}$$
$$+ CR_{i,j+½,k}\,h_{i,j+1,k} + CC_{i+½,j,k}\,h_{i+1,j,k} + CV_{i,j,k+½}\,h_{i,j,k+1} = RHS_{i,j,k}$$

Equation (2.2)

The *CV*, *CR* and *CC* coefficients are conductances between nodes, defined as the product of hydraulic conductivity and cross-sectional area of flow divided by the length of the flow path (in this case the distance between the nodes). The *HCOF* and *RHS* coefficients are composed of external source terms and storage terms (constant flows such as abstractions are added to *RHS*, whilst head dependent boundary conditions affect both the *HCOF* and *RHS* coefficients). The *i*, *j* and *k* terms relate to the row, column and layer numbers in the model respectively.Besides calculating the conductances and storage terms, the BCF package calculates flow correction terms that are added to *HCOF* and *RHS* when an underlying aquifer becomes partially unsaturated, effectively reducing the downward flow to a constant limiting value. Other functions of the BCF package include the desaturation (drying), and resaturation (wetting) of cells, and alternative formulations for calculating the conductances between cells.

2.2 Conceptualisation and implementation of new features

The following discussion of the conceptualisation and implementation of the new features included in the BCF package is divided into seven sections: Variation of hydraulic conductivity with depth (VKD), Variation of storage with depth (VSD), Initial conditions, Spatially variable anisotropy, Inter-nodal transmissivity, Explicit transmissivity calculation and Transmissivity output option.

2.2.1 Variation of hydraulic conductivity with depth (VKD)

The traditional MODFLOW layer types (LAYCON = 0 to 3) assume either that transmissivity remains constant, or that the hydraulic conductivity within the aquifer layer is constant, resulting in a linear relationship between groundwater head and transmissivity. For many aquifers hydraulic conductivity is a function of depth, and these formulations are inappropriate; especially for aquifers in fractured rock, such as chalk and limestone, where groundwater flow is primarily through fissures which have been enhanced due to groundwater flow near the water table.

Groundwater systems where hydraulic conductivity varies with depth can be simulated using a model with a number of traditional MODFLOW layers, however, if the water table fluctuates through these layers numerical problems associated with desaturation and resaturation of cells can become a problem.

This version of the BCF package introduces two new layer types to represent aquifers where hydraulic conductivity reduces with depth within the layer. The first of these (LAYCON = 4) can be used to represent unconfined layers only, the second (LAYCON = 5) can be used to represent layers that can convert between confined and unconfined depending on the groundwater head.

Conceptual model for variation of hydraulic conductivity with depth

The conceptual model for variation of hydraulic conductivity with depth is based on a modified version of that employed at the University of Birmingham (Rushton *et al,* 1980, 1982 & 1989) and is shown in Figure 2.1. For the lower part of the aquifer, the hydraulic conductivity takes a constant value, K_{base}. Above a reference level the hydraulic conductivity increases linearly, increasing at a rate of f per unit rise in groundwater head, until a maximum value of hydraulic conductivity is reached, K_{max}. f and K_{max} can either be defined in units of hydraulic conductivity or as a multiple of K_{base} depending on the preference of the user. The transmissivity is simply the integral of the hydraulic conductivity between the base and the water table (or between the base and the top if the layer is confined).

This conceptualisation has been tried and tested in many groundwater models of chalk and limestone aquifers constructed by the University of Birmingham and other industry consultants (Salmon *et al,* 1996).

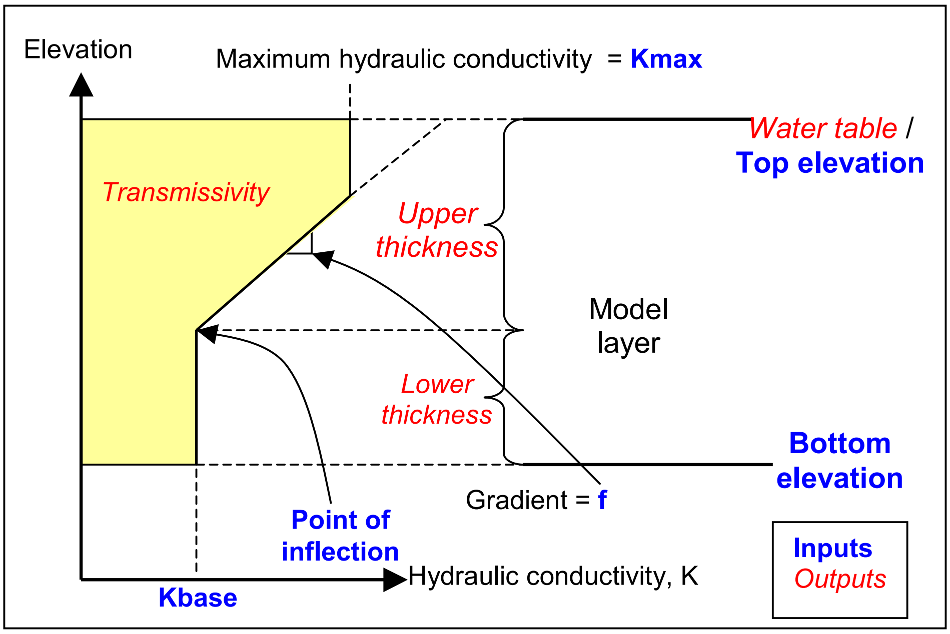


Figure 2.1 Conceptual model for the variation of hydraulic conductivity with depth

Implementation of VKD in MODFLOW

The transmissivity for each cell is calculated each iteration from the following properties:

Base hydraulic conductivity, K_{base} [LT^{-1}]
Bottom elevation, e_{bot} [L]
Elevation of point of inflection, e_{mid} [L]
Hydraulic conductivity gradient (or gradient *factor*), f [T^{-1}] (or [L^{-1}])
Maximum hydraulic conductivity (or max K *factor*), K_{max} [LT^{-1}] (or [-])
Top elevation, e_{top} [L]
Groundwater head, h [L]

In the equations below it is assumed that f and K_{max} are in units of hydraulic conductivity rather than as multiples of K_{base}.

The elevation (e_{Kmax}) where the hydraulic conductivity reaches its maximum value is given by

$$e_{k\max} = e_{mid} + \frac{K_{\max} - K_{base}}{f}$$

Equation (2.3)

The transmissivity is calculated in different ways depending on the elevation of the groundwater head within the hydraulic conductivity profile. If the head is below the point of inflection then the transmissivity, *T*, [L^2T^{-1}] is simply

$$T = K_{base}\left(h - e_{bot}\right)$$

Equation (2.4)

If the head is above the point of inflection but below the elevation where the hydraulic conductivity reaches its maximum, then the following relationship is used:

$$T = K_{base}\left(h - e_{bot}\right) + \frac{f\left(h - e_{mid}\right)^2}{2}$$

Equation (2.5)

If the head is above the elevation where the hydraulic conductivity reaches its maximum, the transmissivity is given by:

$$T = K_{base}\left(h - e_{bot}\right) + \frac{f\left(e_{K\max} - e_{mid}\right)^2}{2} + \left(K_{\max} - K_{base}\right)\left(h - e_{K\max}\right)$$

Equation (2.6)

If the groundwater head is above the top of the layer (for layer type 5 only) then Equation 2.6 is used with the groundwater head, *h*, replaced by the top elevation, e_{top}.

If the groundwater head is below the bottom elevation then the cell becomes inactive (dry) in the same way as for layer types 1 and 3.

Notice that Equation 2.5 includes a term with groundwater head (*h*) squared. Introducing the squared term makes the mathematical problem non-linear. Such problems are inherently harder to solve. In this case the transmissivity (and hence the conductance terms) can change dramatically as the groundwater head values are updated at each node during each iteration of the solver routines. This can have a detrimental effect on simulation convergence.

Issues relating to convergence

As a result of the non-linear relationship between groundwater head and transmissivity, simulations using VKD can be subject to instabilities. This is due to groundwater head values being over or underestimated as the transmissivity values are updated each iteration. This is especially true for steady-state simulations, where the initial conditions need to be close enough to the actual solution in order for the simulation to converge. The automatic conversion option (for converting standard MODFLOW-96 models to VKD models – see Section 2.2.3) has been developed to avoid these instability problems.

Whilst VKD produces a non-linear relationship between groundwater head and transmissivity, it also causes a reduction in the variation of groundwater heads (when heads are above the point of inflection) compared to simulations that use an average hydraulic conductivity. This reduction in head variations means that, when running time-variant simulations, the heads from the last time step are likely to be closer to the solution for the current time step, therefore requiring fewer iterations to converge. This coupled with the general damping effect of aquifer storage, means that time-variant VKD simulations are far more stable than steady-state VKD simulations.

If problems are experienced with time variant simulations, an option has been added that enables the code to use the groundwater head from the last time step in the transmissivity calculations (set IHOLD to 1 – see Section 2.2.6). This means that the transmissivity is held constant throughout a time step, which may improve convergence in time-variant simulations. This explicit formulation for the transmissivity calculations is not as accurate as the implicit formulation (default), and may even result in reduced stability if large time steps are used.

2.2.2 Variation of storage with depth (VSD)

Associated with the reductions in horizontal hydraulic conductivity, there is also evidence of reductions in storage with depth in fissured groundwater systems (Owen and Robinson, 1978, Southern Water Authority, 1979). Therefore, to complement the variation of hydraulic conductivity with depth, the new layer types in MODFLOW (LAYCON = 4 or 5) also allow variations in storage coefficient (specific yield) with depth (VSD). It should be noted that this feature can cause instabilities and problems with convergence and it is recommended that its use be avoided unless the field evidence suggests that it is an important feature of the catchment.

Conceptual model for S variation with depth

The conceptual model for variation of storage with depth is based on that for the variation of hydraulic conductivity with depth and is shown in Figure 2.2. For the lower part of the aquifer, the specific yield takes a constant value, S_{base}. Above a reference level (independent of that for hydraulic conductivity) the storage increases linearly, increasing at a rate of F_S per unit rise in groundwater head, until a maximum value of storage is reached, S_{max}. F_S and S_{max} can either be defined in units of specific yield or as a multiple of S_{base} depending on the preference of the user. The storage accessible from the aquifer depends on the level of the water table within this profile.

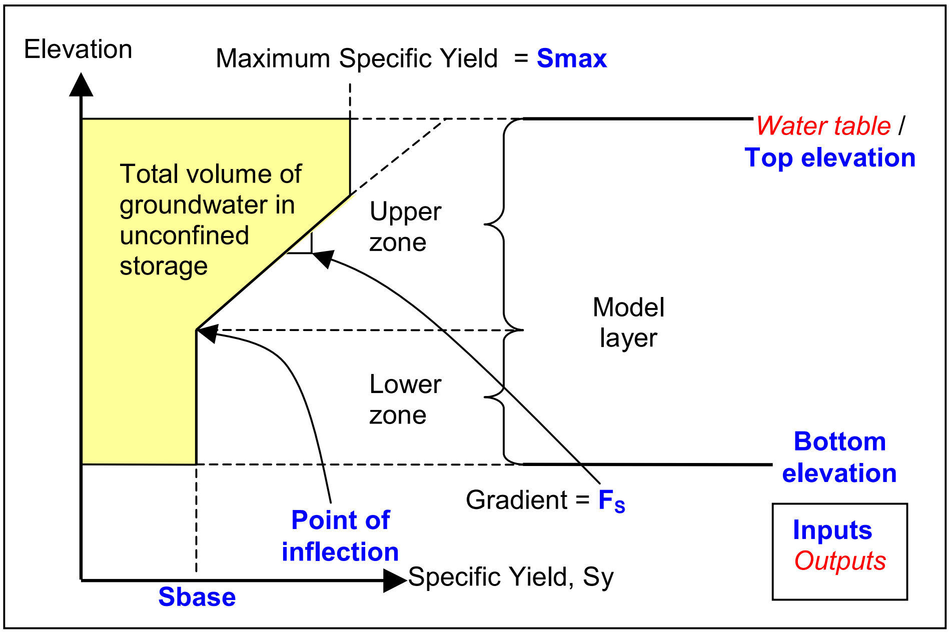


Figure 2.2 Conceptual model for the variation of storage with depth

A problem related to the use of VSD when both confined and unconfined storage coefficients are employed was discovered and corrected in January 2003. The problem emerged whenever the point of inflection of the VSD profile was set above the top elevation of the model layer, and consisted in not replacing specific yield with the confined storage coefficient when calculating flows to and from storage.

Implementation of VSD

The part of the groundwater flow equation (2.1) that relates to the flows to and from storage is the last term in Equation 2.1:

$$S\frac{\partial h}{\partial t}$$

Equation (2.7)

when S is constant, the finite difference form of this term can be written as:

$$S(h-h^*)/\Delta t$$

Equation (2.8)

Where S is the storage coefficient,
h is the head in the current time step,
h^* is the head from the previous time step, and
Δt is the time step length.

This can be substituted into the constant flux (*RHS*) and head dependent (*HCOF*) terms of equation 2.2 as:

$$RHS = -S\,h^*/\Delta t$$
$$HCOF = -S/\Delta t$$

Equation (2.9)

However, when the storage changes as a function of groundwater head (as in Figure 2.3 below), the flows to or from storage can be proportional to the square of the change in groundwater head. As terms relating to the square of groundwater head are not included in Equation 2.2, an approximation has to be made. The approximation assumes that when the solution has converged the change in groundwater head between iterations is small enough that the corresponding change in storage coefficient is negligible.

If the head is in the variable storage zone, the assumption that the change in storage coefficient is negligible is incorrect at the start of a time step. Therefore the storage coefficient used at the start of an iteration is inconsistent with the head at the end, leading to water a balance error. This means that the storage coefficient must be updated before the next iteration. The approximation is stable if the head has moved between two points in the constant S zone (as the storage coefficient at the beginning and end of the iteration will be equal). However, problems of convergence may occur if the head is in the variable storage zone, when the storage coefficient will change between iterations.

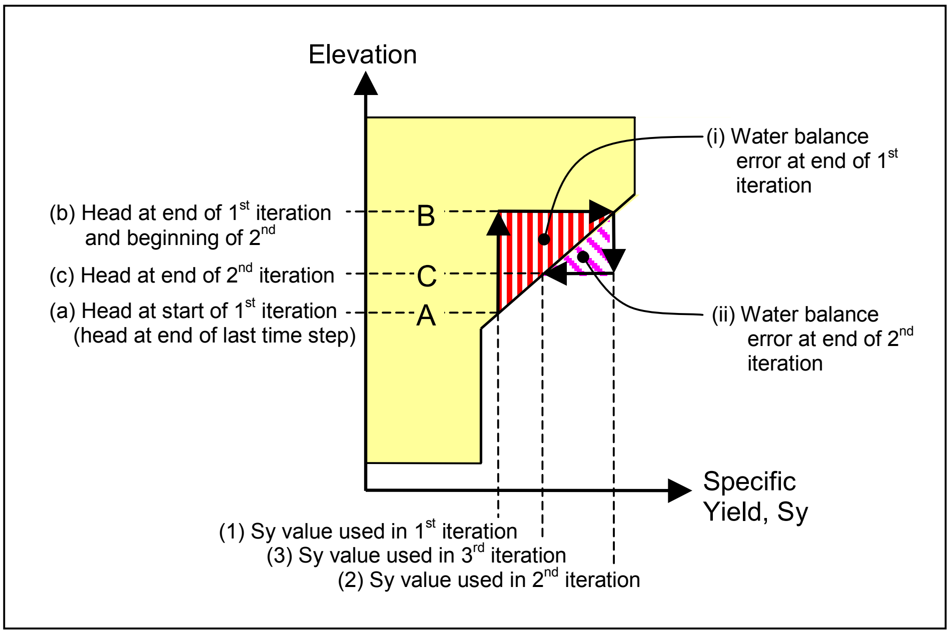


Figure 2.3 Approximation used in VSD calculations

The way the code implements the approximation is to calculate the total water in storage for the head values at the current iteration, and the previous time step (this is shown for the beginning of the 2nd iteration in Figure 2.3 as the shaded area below lines B and A respectively). The total change in storage (the shaded area *between* lines A and B) is then added to the *RHS* coefficient. The *HCOF* coefficient is then given a value reflecting the current storage coefficient (line 2 in Figure 2.3) at the current groundwater head (line B). Finally a correction term equal to the *HCOF* value multiplied by the current groundwater head value is added to the *RHS* coefficient.

This approximation means that the storage coefficient is held constant during each iteration, leading to a water balance error (shown as (ii) in Figure 2.3 for the end of the 2nd iteration). This error should gradually reduce as the changes in head between iterations become smaller. The approximation will converge on a solution once the change in groundwater head between iterations (and hence the change in storage coefficient) is small enough. However large changes in head can often lead to instabilities.

Issues relating to convergence

Problems with non-convergence arise when the storage coefficient changes between iterations. This can also happen when the groundwater head is at the top of the layer, and the storage coefficient changes between confined and unconfined values, but it is most likely to happen when the head is above the point of inflection in the zone of increasing storage. The problem occurs when the groundwater head is calculated using an initial value for the storage coefficient, which then changes based on the new head value. The situation can then arise where the groundwater head fluctuates between levels of different storage coefficient, without converging on a solution.

Simulations that include variations in storage with depth are particularly prone to numerical instabilities due to the approximation in the VSD calculations described above. For this reason it is suggested that this capability should be made inactive (by setting the maximum factor to one) unless the field evidence indicates that it is an important feature of the catchment. However, if the feature is required, a number of methods are available to combat the instabilities and non-convergence. These include:

- Try different combinations of solver parameters. The PCG2 solver is a reliable solver, and is especially good at solving non-linear problems when the maximum number of inner iterations is set to one, and/or the damping factor is set to a low value (see Section 4).
- Reduce the size of the time steps, which reduces the change in heads between time steps and hence minimises the effect of the approximation.
- Minimise the zone of increasing storage by setting the storage gradient, F_s, to a higher value than the maximum storage factor, S_{max} (the distance between the point of inflection and the elevation of maximum storage is equal to $(S_{base} - S_{max})/F_s$, or $(1 - S_{max})/F_s$ if gradient and maximum factors are specified as multiples of S_{base}).
- Avoid setting the elevation of the change in storage to the same as that for hydraulic conductivity. Although conceptually these points may be at the same level, a small offset (~1 metre) may help to reduce instabilities.

2.2.3 Initial conditions: auto-conversion for steady state simulations

As mentioned in Section 2.2.1 (*Issues relating to convergence*) the initial conditions for a steady state VKD simulation need to be similar to the solution in order for the model to converge. For this reason a methodology has been developed where an initial solution is found using a simplified model, the VKD parameters are then introduced and a second simulation with VKD active is started using initial heads from the simplified model. This stepwise approach provides a means of building up the complexity of a model, whilst avoiding numerical instabilities.

Auto-conversion options

As part of the modifications to the MODFLOW code, two new steady state options are available that implement the stepwise approach shown in Figure 2.4 (after page 54). These options can be used to convert a standard MODFLOW-96 model to a model with VKD. The first option converts a model layer with specified transmissivities to a VKD layer, the second option converts a layer with specified hydraulic conductivity to VKD. The following steps are implemented when either of the two new options are activated:

1) Depending on the option chosen, model layers in which VKD is required can be assigned either transmissivity or (vertically averaged) hydraulic conductivity values, which are then used in the first simulation to calculate the initial heads. In addition to the transmissivity or hydraulic conductivity values, the user must specify parameters relating to the variation of hydraulic conductivity with depth. These are (see Figure 2.5 and Table 2.1): the upper thickness, the K gradient factor (f), the maximum K factor (Kmax) and the bottom elevation (and top elevation if the layer can become confined). The upper thickness, gradient and maximum factors are not used in the first simulation. When using the auto conversion option the hydraulic conductivity gradient and maximum hydraulic conductivity must be entered in terms of a multiple of the base hydraulic conductivity (absolute values in hydraulic conductivity units are not supported).

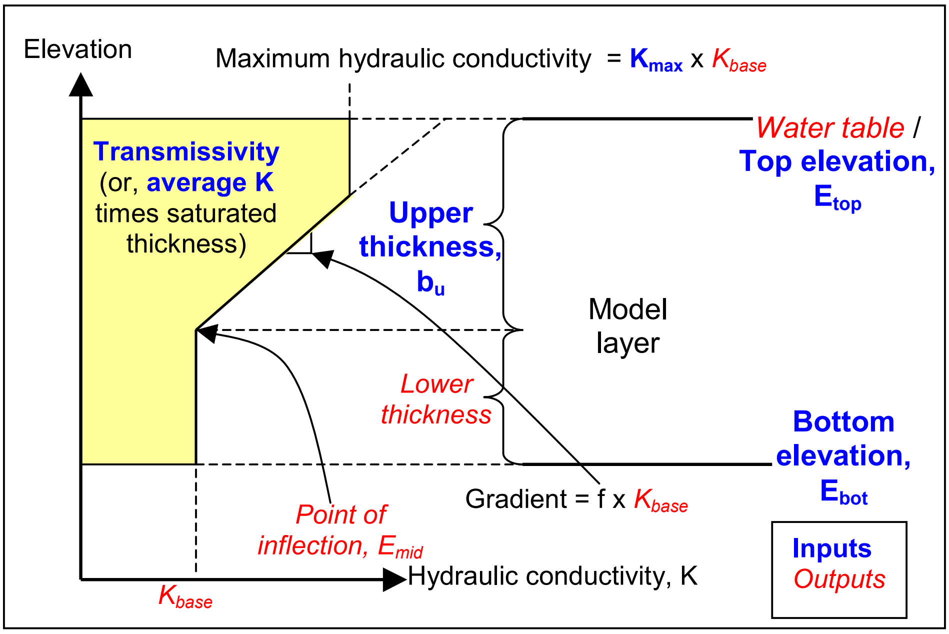


Figure 2.5 Inputs and outputs for automatic conversion option

2) Once the first simulation has produced a solution, the parameters required for the VKD simulation (Figure 2.1) are calculated. The elevation of the change in hydraulic conductivity (point of inflection) is calculated from the upper thickness and the calculated head (or top elevation if the node is confined). If the specified upper thickness is greater than the total saturated thickness then it is reduced and messages are written to the output file (the standard MODFLOW listing file for the first simulation). The base hydraulic conductivity (Kbase) is calculated from the following relationship:

$$K_{base} = \frac{T}{\left(h - E_{bot} - \frac{1}{2f}(K_{max}-1)^2 + b_u(K_{max}-1)\right)}$$

Equation (2.10)

If K_{max} is greater than b_u x f + 1, it is higher than the highest possible K attainable by letting K increase linearly up to the water table/ top elevation, as shown in Figure 2.6. In this case K_{base} is calculated as:

$$K_{base} = \frac{T}{\left(h - E_{bot} + \frac{b_u^2 f}{2}\right)}$$

Equation (2.11)

where K_{base} is the base hydraulic conductivity [LT^{-1}],
T is the transmissivity [L^2T^{-1}],
h is the groundwater head [L],
E_{bot} is the elevation of the base of the layer [L],
f is the hydraulic conductivity gradient factor [L^{-1}] (can not be [T^{-1}]),
K_{max} is the maximum hydraulic conductivity factor [-] (can not be [LT^{-1}]), and
b_u is the thickness of the upper part of the layer [L].

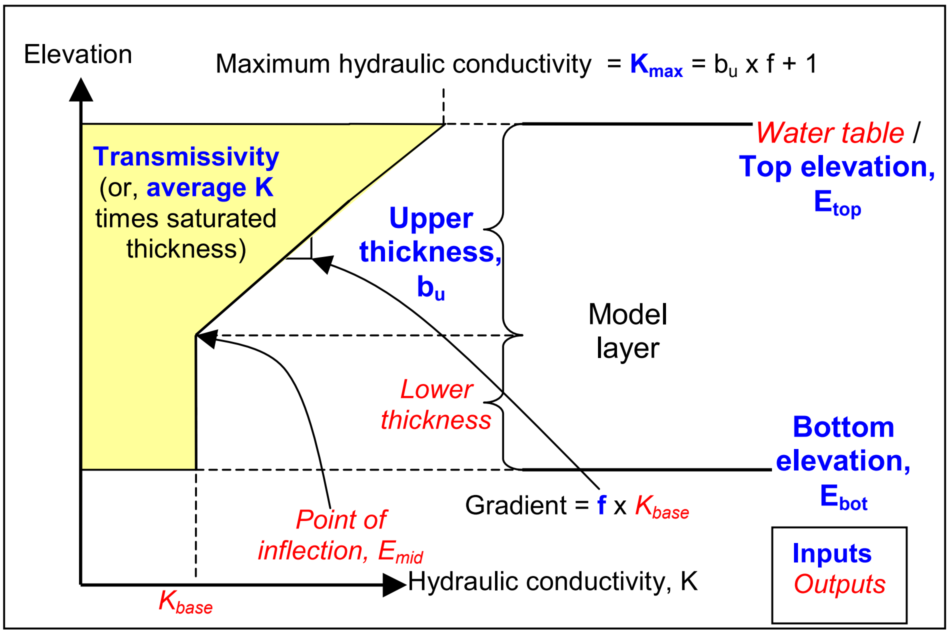


Figure 2.6 Hydraulic conductivity profile when K_{max} is greater than b_u x f + 1

The calculated values are then written to a new BCF file, the initial heads are written to a new BAS (basic) file, and a new NAME file is created with references to both these new files, any other input files from the original model, and new output files.

The formats for the arrays in the new BCF and BAS files remain as they were in the original files, with the exception of the newly calculated variables (initial heads, base hydraulic conductivity and point of inflection) which are automatically given formats with a high degree of precision (10E23.16). The hydraulic conductivity gradient and maximum factors are written in terms of multiples of the base hydraulic conductivity (IKBASE is zero).

The names of the new BAS, BCF and output files can be specified in the original NAME file (see Section 6.2). If output file names are omitted, the original filenames are used with the last character of the filename replaced with a "2" (eg, "Run1.ba2", "Run1.bc2", "Run1.hd2", etc). The new NAME file is always given the same name as the original NAME file, with the last character replaced with a "2" (eg, "Run1.na2").

3) The second simulation is then started automatically using the new input files. The results of the second simulation should be virtually identical to those from the first simulation. This can be verified by checking that the drawdowns from the second simulation are very close to zero (the drawdowns should be less than twice the head convergence criteria, and any differences in flow should be less than twice the flow convergence criteria).

Table 2.1 Inputs and outputs for the first simulation of an auto-conversion run

Symbol	Description	Variable name in MODFLOW	Dimensions
	Inputs for first simulation		
T (or K)	Transmissivity (or hydraulic conductivity)	Tran (stored in CC array) (or HY)	L^2T^{-1} (or LT^{-1})
E_{bot}	Bottom elevation	BOT	L
E_{top}	Top elevation (for LAYCON = 5 only)	TOP	L
f	Hydraulic conductivity gradient (or gradient *factor*)	VKGRAD	T^{-1} (or L^{-1})
K_{max}	Maximum hydraulic conductivity (or max K *factor*)	VKMAX	LT^{-1} or -
b_u	Upper thickness	VMID	L
H	Groundwater head (water table if unconfined)	HNEW	L
K_{base}	Base hydraulic conductivity	HY	LT^{-1}
E_{mid}	Point of inflection	VMID	L
-	Lower thickness	-	L

For a more detailed description of the inputs required for each of the options see the input instructions (Section 2.3).

2.2.4 Spatially variable anisotropy (X- & Y-direction transmissivities)

These code changes are based on those described by Ruskauff and Kladias (1996) and were the first to be implemented as an introduction to modifying the code and to allow comparison with the Birmingham University code (against which the modified MODFLOW code was tested). MODFLOW-96 allows column to row anisotropy to vary only between different layers. Some geologic settings may require a variable anisotropic formulation within a layer in order to be properly represented. For instance, fracturing may be more pronounced along the crest of an anticline than the flanks, dissolution of limestone more intense near discharge areas, and channel deposits abutting overbank deposits.

Modifications to the code were made to allow each grid block in the model to have its own anisotropy, and still retain backward compatibility with the original formulation. Further modifications were then made to allow X- and Y-direction transmissivities or hydraulic conductivities to be specified directly.

2.2.5 Internodal transmissivity/hydraulic conductivity

This change to the code was made primarily to allow a direct comparison with the Birmingham University VKD modelling code (which is node-centred rather than block-centred like MODFLOW). However, it does have the added advantage of giving greater control over a model's transmissivity distribution, especially when used in conjunction with the variable anisotropy option.

MODFLOW-96 calculates the conductance between a pair of model nodes based on an average of the transmissivities specified (or calculated) for those nodes. A number of different averaging techniques are available, including harmonic, arithmetic, or logarithmic means. However, all these techniques produce conductances derived from the properties assigned to both the cells. This modification allows inter-nodal transmissivities (or hydraulic conductivities) to be specified for a model layer. In this case each nodes transmissivity value is applied between that node and the next node to the right (for transmissivity along rows) or between the node and the next node to the front (for transmissivity along columns). For unconfined aquifers, the arithmetic mean of the groundwater heads (and bottom elevations and other properties) is used to calculate the conductances. This modification also allows MODFLOW to be compared with node-centred codes such as the Birmingham University code, (provided that parameter values (ie, recharge, storage, transmissivity, etc) are halved at edge nodes, and quartered at corner nodes). It also allows greater control of the parameter values used in the flow equations, especially for modelling narrow features associated with rivers or faults.

2.2.6 Explicit transmissivity calculation

Another change that was required to allow a direct comparison with the Birmingham University code was to include an option that keeps calculated transmissivities constant during a time step. If this option is selected then the heads from the previous time step are used to calculate the transmissivity, rather than the heads from the previous iteration. This formulation is slightly less accurate than the implicit formulation used as default, but has the advantage of reducing the number of iterations required for convergence. However, it should be noted that although this option does not make the formulation of the finite difference equations fully explicit, it is still possible that oscillations may be produced in the time-variant solution if there are large changes in head between time steps.

2.2.7 Transmissivity output option

This option has been included to enable the results of the transmissivity calculations to be checked. If the option is selected, the calculated X- and Y-direction inter-nodal transmissivities are written to either the output file (the text-formatted listing file), or a binary output file at the end of each time step. The values can then be checked using a text editor. It is not recommended that this option be used for time-variant models, as the output files produced can be very large.

2.3 Input instructions

2.3.1 Summary of new options

This new version of the MODFLOW modelling code allows a number of new options to be set (shown in bold) in the input instructions for the BCF package. A summary of the inputs required for a VKD simulation is provided in Table 2.2 (Section 2.3.4). The new code should still be compatible with old MODFLOW input files (versions up to MODFLOW-96). The new options included are:

▪ **Spatially variable anisotropy -**	set **ITRPY = 1** and use an array for TRPY for each layer
▪ **X & Y-direction transmissivities or hydraulic conductivities -**	set **ITRPY = 2** and include a 2nd array following the original transmissivity or hydraulic conductivity arrays
▪ **Explicit transmissivity calculation (more stable) -**	set **IHOLD = 1**
▪ **Write calculated transmissivity values to the listing file -**	set **ITRANS <> 0** (not recommended for transient simulations)
▪ **Internodal transmissivity / hydraulic conductivity -**	set LAYAVG = **40** (first digit of LAYCON = **4**X)
▪ **Variable hydraulic conductivity (VKD) and specific yield with depth (VSD) – unconfined -**	set LAYCON = **4** (last digit of LAYCON = X**4**) and include the **VMID, VKGRAD** & **VKMAX** arrays (also **VSMID, VSGRAD** & **VSMAX** arrays if time variant)
▪ **Variable hydraulic conductivity (VKD) and specific yield with depth (VSD) – confined / unconfined -**	set LAYCON = **5** (last digit of LAYCON = X**5**) and include the TOP, **VMID, VKGRAD** & **VKMAX** arrays (also **VSMID, VSGRAD** & **VSMAX** arrays if time variant)
▪ **Alternative steady state options for VKD simulations -**	Set ISS = **2** (Converts specified transmissivity layers to VKD layers) or set ISS = **3** (Converts specified hydraulic conductivity layers to VKD layers). Layers to be converted should be specified as LAYCON = X**4** or X**5** (other layer types will not be affected) and include the BOT, **Upper Thickness**, **VKGRAD** & **VKMAX** arrays

2.3.2 Input format

As this version of the code is based on MODFLOW-96, free-format input is permitted (note: if an old free-format input file does not work, four zeros may need to be added to the end of the first line of the BCF file for ITRPY, IHOLD, ITRANS and IKBASE). The following shows the input instructions for the new BCF package (including format codes for fixed format input, and modules used to read arrays).

Read in subroutine BCF5AL

1.	Data:	ISS	IBCFCB	HDRY	IWDFLG	WETFCT	IWETIT	IHDWET	**ITRPY**	**IHOLD**	**ITRANS**	**IKBASE**
	Format:	I10	I10	F10.0	I10	F10.0	I10	I10	**I10**	**I10**	**I10**	**I10**

2. Data: LAYCON(NLAY) (Maximum of 200 layers)
   Format: 40I2

(If there are 40 or fewer layers, use one record; otherwise, use two or more records, with 40 values in each record.)

Read in subroutine BCF5RP

IF ANISOTROPY IS SPECIFIED FOR EACH LAYER (**ITRPY = 0**)

3. Data: TRPY(NLAY)
   Module: U1DREL

IF ANISOTROPY IS SPECIFIED FOR EACH NODE (ITRPY is not 0 or 2)

3a. Data: TRPY(NCOL,NROW,NLAY)
Module : U2DREL
(Repeat for each layer.)

REQUIRED FOR ALL SIMULATIONS

4. Data: DELR(NCOL)
   Module: U1DREL

5. Data: DELC(NROW)
   Module: U1DREL

A subset of the following two-dimensional arrays is used to describe each layer. The arrays needed for each layer depend on the layer-type code (LAYCON), whether the simulation is transient (ISS=0) or steady state (ISS not 0), if the wetting capability is active (IWDFLG not 0), and whether nodal anisotropy values are used (ITRPY not 0). If an array is not needed, it must be omitted. In no situation will all arrays be required. The required arrays (items 6-19) for layer 1 are read first; then the arrays for layer 2, etc.

IF THE SIMULATION IS TRANSIENT (ISS = 0)

6. Data: Sf1(NCOL,NROW)
   Module: U2DREL

IF THE LAYER-TYPE CODE (LAYCON) IS 0 OR 2

7. Data: TranX(NCOL,NROW)
   Module: U2DREL

IF THE LAYER-TYPE CODE (LAYCON) IS 0 OR 2, AND TRANSMISSIVITY IS SPECIFIED IN BOTH X- AND Y-DIRECTIONS (ITRPY = 2)

7a. Data: TranY(NCOL,NROW)
   Module: U2DREL

IF THE LAYER-TYPE CODE (LAYCON) IS 1, 3, **4 OR 5**

8. Data: HY(NCOL,NROW)
   Module: U2DREL

IF THE LAYER-TYPE CODE (LAYCON) IS 1, 3, 4 OR 5 AND HYDRAULIC CONDUCTIVITY IS SPECIFIED IN BOTH X- AND Y-DIRECTIONS (ITRPY = 2)

8a. Data: Hy-y(NCOL,NROW)
   Module: U2DREL

IF THE LAYER-TYPE CODE (LAYCON) IS 1, 3, **4 OR 5**

9. Data: BOT(NCOL,NROW)
   Module: U2DREL

IF NOT THE BOTTOM LAYER

10. Data: Vcont(NCOL,NROW)
   Module: U2DREL

IF THE SIMULATION IS TRANSIENT (ISS=0) AND THE LAYER-TYPE CODE (LAYCON) IS 2, 3 OR **5**

11. Data: Sf2(NCOL,NROW)
   Module: U2DREL

IF THE LAYER-TYPE CODE (LAYCON) IS 2, 3 OR **5**

12. Data: TOP(NCOL,NROW)
    Module: U2DREL

IF THE LAYER-TYPE CODE (LAYCON) IS 1, 3, **4 OR 5**, AND THE WETTING CAPABILITY IS ACTIVE (IWDFLG IS NOT 0)

13. Data: WETDRY(NCOL,NROW)
    Module: U2DREL

IF THE LAYER-TYPE CODE (LAYCON) IS 4 OR 5

14. Data: VMID(NCOL,NROW)
Module: U2DREL

IF THE LAYER-TYPE CODE (LAYCON) IS 4 OR 5

15. Data: VKGRAD(NCOL,NROW)
Module: U2DREL

IF THE LAYER-TYPE CODE (LAYCON) IS 4 OR 5

16. Data: VKMAX(NCOL,NROW)
Module: U2DREL

IF THE SIMULATION IS TRANSIENT (ISS=0) AND THE LAYER-TYPE CODE (LAYCON) IS 4 OR 5

17. Data: VSMID(NCOL,NROW)
Module: U2DREL

IF THE SIMULATION IS TRANSIENT (ISS=0) AND THE LAYER-TYPE CODE (LAYCON) IS 4 OR 5

18. Data: VSGRAD(NCOL,NROW)
Module: U2DREL

IF THE SIMULATION IS TRANSIENT (ISS=0) AND THE LAYER-TYPE CODE (LAYCON) IS 4 OR 5

19. Data: VSMAX(NCOL,NROW)
Module: U2DREL

2.3.3 Explanation of parameters used in input instructions

ISS is the steady-state flag

If ISS is not 0, the simulation is steady state.

If ISS = 0, the simulation is transient.

If ISS = 2, two steady state simulations are run. The first simulation uses specified *transmissivities* (input into the HY array) for VKD layers (LAYCON is 4 or 5 – see below), the second uses the variable hydraulic conductivity with depth (VKD) parameters. In this case the original BCF package should have the thickness (not elevation) of the upper zone input into the VMID array. Secondary BCF, BAS and NAM files are created after the first simulation (filenames for these files can be specified in the name file – see Section 6.2). These are used to set up the second simulation.

If ISS = 3, two steady state simulations are run. The first simulation uses specified *hydraulic conductivities* (input into the HY array) for VKD layers (LAYCON is 4 or 5 – see below), the second uses the variable hydraulic conductivity with depth (VKD) parameters. In this case the original BCF package should have the thickness (not elevation) of the upper zone input into the VMID array. Secondary BCF, BAS and NAM files are created after the first simulation (filenames for these files can be specified in the name file – see Section 6.2). These are used to set up the second simulation.

IBCFCB is a flag and a unit number.

If IBCFCB > 0, it is the unit number on which cell-by-cell flow terms will be recorded whenever ICBCFL (see Output Control in McDonald and Harbaugh [1988, p. 4-15]) is set; the terms which are saved will include cell-by-cell storage terms, cell-by-cell constant head flows, and cell-by-cell flow between adjacent cells.

If IBCFCB = 0, cell-by-cell flow terms will not be printed or recorded

If IBCFCB < 0, flow for each constant-head cell will be printed in the listing file whenever ICBCFL is set; sell-by-cell storage terms and cell-by-cell flow between adjacent cells will not be recorded or printed

HDRY is the head that is assigned to cells that are converted to dry during a simulation. Although this value plays no role in the model calculations, it is useful as an indicator when looking at the resulting heads that are output from the model. HDRY is thus similar to HNOFLO in the Basic Package, which is the value assigned to cells that are no-flow cells at the start of a model simulation. It is wise to use a value of HDRY which is below the bottom of the lowest part of the aquifer. This avoids dry cells from becoming active if heads are to be used as starting conditions for a subsequent run

IWDFLG is a flag that determines if the wetting capability is active.

If IWDFLG = 0, the wetting capability is inactive.

If IWDFLG is not 0, the wetting capability is active

WETFCT is a factor that is included in the calculation of the head that is initially established at a cell when it is converted from dry to wet. (see IHDWET).

IWETIT is the iteration interval for attempting to wet cells. Wetting is attempted every IWETIT iterations. If using the PGG solver (Hill, 1990), this apples to outer iterations, not inner iterations. If IWETIT is 0, it is changed to 1.

IHDWET is a flag that determines which equation is used to calculate the initial head at cells that become wet:

If IHDWET = 0, equation (3a) is used:

H = BOT + WETFCT (hn – BOT),

(hn = head in neighbouring cell causing cell to wet)

If IHDWET is not 0, equation (3b) is used:

H = BOT + WETFCT (THRESH).

ITRPY is a flag that determines whether the anisotropy factor (TRPY) is specified for each layer or for each node

If ITRPY = 0, anisotropy (TRPY) is a one-dimensional array with one value per layer

If ITRPY is not 0 and not 2, anisotropy (TRPY) is a three-dimensional array with a value for each cell

If ITRPY = 2, anisotropy (TRPY) is a three-dimensional array with a value for Y-direction transmissivities or hydraulic conductivities for each cell (X-direction values remain in the TranX or HY array).

IHOLD is a flag that determines whether the calculation of transmissivity (for LAYCON values of 1, 3, 4 or 5) uses the heads calculated in the previous timestep (explicit formulation) or those calculated in the previous iteration (implicit formulation).

If IHOLD = 0, the transmissivity is updated every iteration using the groundwater heads from the current timestep (implicit – slower but more accurate – normal MODFLOW formulation).

If IHOLD = 1, the transmissivity is updated every timestep using the groundwater heads from the previous timestep (explicit – quicker and more stable).

ITRANS is a flag that allows the calculated internodal transmissivities to be output to the listing file every timestep

If ITRANS = 0, internodal transmissivity will not be output to the listing file

If ITRANS > 0, internodal transmissivity will be output to the listing file every timestep. Not recommended for transient simulations.

If ITRANS < 0, it is the unit number on which internodal transmissivity will be recorded whenever IHDDFL (see Output Control in McDonald and Harbaugh [1988, p. 4-15]) is set. The values are written to a binary file using the same format as the head and drawdown output files. The file must be specified in the name file (Section 6.2)

IKBASE is a flag that denotes whether VKGRAD and VKMAX will be defined as a multiple of HY per unit length or in units of hydraulic conductivity per unit length (and whether VSGRAD and VSMAX will be defined as a multiple of Sf per unit length or in units of specific yield per unit length)

If IKBASE = 0, VKGRAD and VKMAX will be entered as multiples of HY per unit length. VSGRAD and VSMAX will be entered as multiples of Sf1 or Sf2 per unit length (depending which represents specific yield – denoted by the value of LAYCON)

If IKBASE is not 0, VKGRAD and VKMAX will be entered in units of hydraulic conductivity per unit length. VSGRAD and VSMAX will be entered in units of specific yield per unit length

LAYCON is the layer-type code consisting of two parts. The first digit (multiplied by 10) gives the averaging method to be used for calculation of the branch conductances (LAYAVG). The second digit describes the layer type (LAYCON). Each element holds the code for the respective layer. Read both values for each layer. There is a limit of 200 layers. If there are 40 or fewer layers, use one record; otherwise, use two or more records. Leave unused elements in a record blank

LAYAVG:

00 – Harmonic averaging of nodal transmissivity values

10 – Arithmetic averaging of nodal transmissivity values

20 – Logarithmic averaging of nodal transmissivity values

30 – Logarithmic averaging of hydraulic conductivity, arithmetic averaging of saturated thickness

40 – Transmissivity/hydraulic conductivity values are specified between nodes, saturated thickness (if applicable) is arithmetically averaged.

LAYCON:

0 – confined – Transmissivity and storage coefficient of the layer are constant for the entire simulation

1 – unconfined – Transmissivity of the layer varies. It is calculated from the saturated thickness and hydraulic conductivity. The storage coefficient is constant. This layer type is valid only for layer 1.

2 – confined/unconfined – Transmissivity of the layer is constant. The storage coefficient may alternate between confined and unconfined values. Vertical flow from above is limited if the layer desaturates.

3 – confined/unconfined – Transmissivity of the layer varies. It is calculated from the saturated thickness and hydraulic conductivity. The storage coefficient may alternate between confined and unconfined values. Vertical flow from above is limited if the aquifer desaturates.

4 – unconfined variable hydraulic conductivity and specific yield with depth (VKD) – Transmissivity and specific yield of the layer varies. Transmissivity is calculated from the saturated thickness, the hydraulic conductivity, the hydraulic conductivity gradient factor, and the maximum hydraulic conductivity factor. The specific yield is calculated from the groundwater head, the primary storage coefficient, the storage gradient factor, and the maximum storage factor.

5 – confined/unconfined variable hydraulic conductivity and specific yield with depth (VKD) – Transmissivity and specific yield of the layer varies. Transmissivity is calculated from the saturated thickness, the hydraulic conductivity, the hydraulic conductivity gradient factor, and the maximum hydraulic conductivity factor. The specific yield is calculated from the groundwater head, the secondary storage coefficient, the storage gradient factor, and the maximum storage factor. The storage coefficient may alternate between confined and unconfined values. Vertical flow from above is limited if the layer desaturates.

TRPY is a one or three-dimensional array (depending on the value of **ITRPY** – see above) containing an anisotropy factor for each layer. It is the ratio of transmissivity or hydraulic conductivity (whichever is being used) along a column to transmissivity or hydraulic conductivity along a row. Set to 1.0 for isotropic conditions. This can be either a single array with one value per layer, or a separate array for each layer (if **ITRPY** is not **0**). If a single array is being used (**ITRPY** = **0**) only one array control record need be included for the entire array. If the value is the same for all layers, the entire array can be specified by setting LOCAT to 0 and setting CNSTNT to the value that applies to all layers. Read only if **ITRPY** is not **2**.

DELR is the cell width along rows. Read one value for each of the NCOL columns. This is a single array with one value for each column.

DELC is the cell width along columns. Read one value for each of the NROW rows. This is a single array with one value for each row.

Sf1 is a primary storage coefficient. Read only for a transient simulation (steady-state flag, ISS, is 0). For LAYCON equal to 1 or **4**, Sf1 will always be specific yield, while for LAYCON equal to 2, 3 or **5** Sf1 will always be confined storage coefficient. For LAYCON equal to 0, Sf1 would normally be confined storage coefficient; however, a LAYCON value of 0 can also be used to simulate watertable conditions where drawdowns are expected to remain everywhere a small fraction of the saturated thickness, and where there is no layer above, or flow from above is negligible. In this case, specific yield values would be entered for Sf1. **If LAYCON is 4, the primary storage coefficient relates to the specific yield available from the lower (constant storage) zone**.

TranX is the transmissivity along rows. TranX is multiplied by TRPY to obtain transmissivity along columns (if **ITRPY** is not **2**). Read only for layers where LAYCON is 0 or 2.

TranY is the transmissivity along columns. Read only for layers where LAYCON is 0 or 2 and ITRPY = 2.

HY is the hydraulic conductivity along rows. HY is multiplied by TRPY to obtain hydraulic conductivity along columns (if **ITRPY** is not **2**). Read only for layers where LAYCON is 1, 3, **4 or 5**. **(If ISS = 2, and LAYCON = 4 or 5, then Transmissivity is read into this array. If ISS = 3, and LAYCON = 4 or 5, then average hydraulic conductivity is read into this array)**.

Hy-y is the hydraulic conductivity along columns. Read only for layers where LAYCON is 1, 3, 4 or 5 and ITRPY = 2. (If ISS = 2, and LAYCON = 4 or 5, then Transmissivity is read into this array If ISS = 3, and LAYCON = 4 or 5, then average hydraulic conductivity is read into this array).

BOT is the elevation of the aquifer bottom. Read only for layers where LAYCON is 1, 3, **4 or 5**.

Vcont is the vertical hydraulic conductivity divided by the thickness from a layer to the layer below. The value for a cell is the hydraulic conductivity divided by thickness for the material between the node in that cell and the node in the cell below. Because there is not a layer beneath the bottom layer, Vcont cannot be specified for the bottom layer.

sf2 is the secondary storage coefficient. Read only for layers where LAYCON is 2, 3 or **5**, and only if the simulation is transient (steady-state flag, ISS, is 0). The secondary storage coefficient is always specific yield. **(If LAYCON = 5, the secondary storage coefficient relates to the specific yield available from the lower (constant storage) zone)**.

TOP is the elevation of the aquifer top. Read only for layers where LAYCON is 2, 3 or **5.**

WETDRY is a combination of the wetting threshold and a flag to indicate which neighbouring cells can cause a cell to become wet. If WETDRY < 0, only the cell below a dry cell can cause the cell to become wet. If WETDRY > 0, the cell below a dry cell and four horizontally adjacent cells can cause a cell to become wet. IF WETDRY is 0, the cell cannot be wetted. The absolute value of WETDRY is the wetting threshold, when the sum of BOT and the absolute value of WETDRY at a dry cell is equalled or exceeded by the head at an adjacent cell, the cell is wetted. Read only if LAYCON is 1, 3, **4 or 5** and IWDFLG is not 0.

VMID is the elevation within the layer where hydraulic conductivity changes from being constant to varying with depth (point of inflection). Read only for layers where LAYCON is 4 or 5. (If ISS = 2 or 3 then the thickness of the upper (variable hydraulic conductivity) zone is read into this array).

VKGRAD is the hydraulic conductivity gradient (or gradient factor – depending on the value of IKBASE) applied to the upper (variable hydraulic conductivity) zone of an aquifer. Read only for layers where LAYCON is 4 or 5. If IKBASE is 0 it is multiplied by the hydraulic conductivity to give the hydraulic conductivity gradient. A positive value means that hydraulic conductivity increases linearly towards the top of the aquifer. Negative values can *not* be used to represent a linear decrease in hydraulic conductivity towards the top of the aquifer; if a zero or negative value is entered then both VKGRAD and VKMAX (below) are set to one. The hydraulic conductivity gradient only applies when the water table is in the upper (variable hydraulic conductivity) zone, ie, the head in the layer is above the VMID elevation.

VKMAX is the maximum hydraulic conductivity (or factor – depending on the value of IKBASE) applied to the upper (variable hydraulic conductivity) zone. Read only for layers where LAYCON is 4 or 5. If IKBASE is 0 it is multiplied by the hydraulic conductivity to give the maximum hydraulic conductivity.

VSMID is the elevation within the layer where storage changes from being constant to varying with depth (point of inflection). Read only for layers where LAYCON is 4 or 5 and if ISS is zero.

VSGRAD is the specific yield gradient (or gradient factor – depending on the value of IKBASE) applied to the upper (variable storage) zone of an aquifer. Read only if ISS is zero, for layers where LAYCON is 4 or 5. If IKBASE is 0 it is multiplied by the specific yield to give the specific yield gradient. A positive value means that specific yield increases linearly towards the top of the aquifer. Negative values can *not* be used to represent a linear decrease in specific yield towards the top of the aquifer; the storage calculations will be corrupted, leading to unrealistic results. The specific yield gradient only applies when the water table is in the upper (variable hydraulic conductivity) zone, ie, the head in the aquifer is above the VSMID elevation.

VSMAX is the maximum storage (or factor – depending on the value of IKBASE) applied to the upper (variable storage) zone. Read only if ISS is zero, for layers where LAYCON is 4 or 5. If IKBASE is 0 it is multiplied by the specific yield to give the maximum specific yield.

2.3.4 Summary of the inputs required for a VKD simulation

Table 2.2 Summary input instructions for VKD simulations

Input block	Steady State VKD	Alternative Steady State VKD (involves 2 simulations)	Time Variant VKD / VSD
1.	Set steady state flag (ISS) = 1	Set ISS = **2 (convert from specified transmissivity layers) or 3 (convert from specified hydraulic conductivity layers)**	Set ISS = 0
2.	Layer type (LAYCON) = X**4** or X**5**	LAYCON = X**4** or X**5**	LAYCON = X**4** or X**5**
3. or 3a.	Anisotropy ratio (TRPY) (if **ITRPY** is not **2**)	TRPY (if **ITRPY** is not **2**)	TRPY (if **ITRPY** is not **2**)
4.	Cell width (DELR)	DELR	DELR
5.	Cell height (DELC)	DELC	DELC
6.	-	-	Sf1 (Specific yield of lower zone if LAYCON = **4**. Confined storage coeff. if LAYCON = **5**)
7.	-	-	-
7a.	**-**	**-**	**-**
8.	Hydraulic conductivity (HY) (of lower zone)	Steady State Transmissivity if ISS is 2, or average hydraulic conductivity if ISS is 3	HY (of lower zone)
8a.	**Y-direction hydraulic conductivity (HY-y) of lower zone (if ITRPY = 2)**	**(if ITRPY = 2) Y-direction transmissivity (ISS=2), or hydraulic conductivity (ISS=3)**	**HY-y (if ITRPY = 2)**
9.	Bottom elevation of layer (BOT)	BOT	BOT
10.	Vertical hydraulic conductivity over distance between layers (Vcont) (if not bottom layer)	Vcont (if not bottom layer)	Vcont (if not bottom layer)
11.	-	-	Sf2 (Sy of lower zone – if LAYCON = **5**)
12.	Top elevation of layer (TOP) (if LAYCON = **5**)	TOP (if LAYCON = **5**)	TOP (if LAYCON = **5**)

Table 2.2 Summary input instructions for VKD simulations (continued)

Input block	Steady State VKD	Alternative Steady State VKD (involves 2 simulations)	Time Variant VKD / VSD
13.	Wetting threshold (WETDRY) (if IWDFLG is not 0)	WETDRY (if IWDFLG is not 0)	WETDRY (if IWDFLG is not 0)
14.	**Elevation of change in hydraulic conductivity (VMID)**	**VMID**	**VMID**
15.	**Hydraulic conductivity gradient factor (VKGRAD)**	**VKGRAD**	**VKGRAD**
16.	**Maximum hydraulic conductivity factor (VKMAX)**	**VKMAX**	**VKMAX**
17.	**-**	**-**	**Elevation of change in storage (VSMID)**
18.	**-**	**-**	**Storage gradient factor (VSGRAD)**
19.	**-**	**-**	**Maximum storage factor (VSMAX)**

2.3.5 Example BCF input file

```
  2      50   -888.0       0    0.000       0       0       0       0       1       0
 15
  0    1.000(10E12.4)                0    ANISOTROPY RATIO
  0  500.000(10E12.4)                0    DELTA-X or C
  0  500.000(10E12.4)                0    DELTA-Y or R
 11    1.000(25F7.1)                 0    TRANSMISSIVITY ALONG ROWS Layer   1
2050.0 2050.0 2050.0 2050.0 1950.0 1850.0 1750.0 1650.0 1550.0 1450.0 1350.0 1250.0 1150.0 1050.0 950.0 850.0 750.0 650.0 550.0 450.0 350.0 250.0 150.0 50.0
2050.0 2050.0 2050.0 2050.0 1950.0 1350.0 1250.0 1150.0 1050.0 950.0 850.0 750.0 650.0 0.0
1950.0 1350.0 1250.0 1150.0 1050.0 950.0 850.0 750.0 650.0 0.0
1950.0 1350.0 1250.0 1150.0 1050.0 950.0 850.0 750.0 650.0 0.0
1950.0 1350.0 1250.0 1150.0 1050.0 950.0 850.0 750.0 650.0 0.0
1950.0 1350.0 1250.0 1150.0 1050.0 950.0 850.0 750.0 650.0 0.0
1950.0 1350.0 1250.0 1150.0 1050.0 950.0 850.0 750.0 650.0 0.0
1950.0 1350.0 1250.0 1150.0 1050.0 950.0 850.0 750.0 650.0 0.0
1950.0 1350.0 1250.0 1150.0 1050.0 950.0 850.0 750.0 650.0 0.0
1950.0 1350.0 1250.0 1150.0 1050.0 950.0 850.0 750.0 650.0 0.0
1950.0 1350.0 1250.0 1150.0 1050.0 950.0 850.0 750.0 650.0 0.0
1950.0 1850.0 1750.0 1650.0 1550.0 1450.0 1350.0 .0 750.0 650.0 550.0 450.0 350.0 250.0 150.0 50.0
2050.0 2050.0 2050.0 2050.0 1950.0 1850.0 1750.0 1650.0 1550.0 1450.0 1350.0 .0 250.0 150.0 50.0
2050.0 2050.0 2050.0 2050.0 1950.0 1850.0 1750.0 1650.0 1550.0 1450.0 1350.0 .0 250.0 150.0 50.0
2050.0 2050.0 2050.0 2050.0 1950.0 1850.0 1750.0 1650.0 1550.0 1450.0 1350.0 .0 250.0 150.0 50.0
2050.0 2050.0 2050.0 2050.0 1950.0 .0 250.0 150.0 50.0
2050.0 2050.0 2050.0 2050.0 1950.0 .0 250.0 150.0 50.0
2050.0 2050.0 2050.0 2050.0 1950.0 .0 250.0 150.0 50.0
2050.0 2050.0 2050.0 2050.0 1950.0 .0 250.0 150.0 50.0
2050.0 2050.0 2050.0 2050.0 1950.0 1850.0 1750.0 1650.0 1550.0 1450.0 1350.0 .0 250.0 150.0 50.0
2050.0 2050.0 2050.0 2050.0 1950.0 1850.0 1750.0 1650.0 1550.0 1450.0 1350.0 .0 250.0 150.0 50.0
2050.0 2050.0 2050.0 2050.0 1950.0 1850.0 1750.0 1650.0 1550.0 1450.0 1350.0 1250.0 1150.0 1050.0 950.0 850.0 750.0 650.0 550.0 450.0 350.0 250.0 150.0 50.0
 11    1.000(25f7.2)                 0    BOTTOM Layer   1
 36.84  36.84  36.80  36.69  40.97  45.24  49.52  53.79  58.06  62.32  66.59  70.84  75.09  79.34  83.58  87.81  92.03  96.24 100.42 104.58 108.70 112.76 116.69 120.22
 36.84  36.84  36.80  36.69  40.97  45.24  49.52  53.79  58.06  62.32  66.59  70.84  75.09  79.34  83.58  87.81  89.75  91.50  93 0.25 102.00
 36.84  36.84  36.80  36.69  40.97  45.24  49.52  53.79  58.06  62.32  66.59  70.84  75.09  79.34  83.58  87.81  91.75  93.50  95 2.25 104.00
 36.84  36.84  36.80  36.69  40.97  45.24  49.52  53.79  58.06  62.32  66.59  70 81  92.03  95.50  97 4.25 106.00
 36.84  36.84  36.80  36.69  40.97  45.24  49.52  53.79  58.06  62.32  66.59  70 81  92.03  96.24  99 6.25 108.00
 36.84  36.84  36.80  36.69  40.97  45.24  49.52  53.79  58.06  62.32  66.59  70 81  92.03  96.24 100 8.25 110.00
 36.84  36.84  36.80  36.69  40.97  45.24  49.52  53.79  58.06  62.32  66.59  70 81  92.03  96.24 100.42 104.58 106.75 108.50 110.25 112.00
 36.84  36.84  36.80  36.69  40.97  45.24  49.52  53.79  58.06  62.32  66.59  70.84  75.09  79.34  83.58  87.81  92.03  96.24 100.42 104.58 108.70 110.50 112.25 114.00
 36.84  36.84  36.80  36.69  40.97  45.24  49.52  53.79  58.06  62.32  66.59  70.84  75.09  79.34  83.58  87.81  92.03  96.24 100.42 104.58 108.70 112.50 114.25 116.00
```

Item: 1
Variables: IBCFCB, HDRY, IWDFLG, WETFCT, IWETIT & IHDWET

Item: 1
Variable: **IKBASE**
Value: **0**
Option: VKGRAD and VKMAX are specified as multiples of K_{base}.

Item: 1
Variable: ISS
Value: **2**
Option: **Auto-conversion simulation using specified transmissivities.**

Item: 2
Variable: LAYCON
Value: 1**5**
Option: **Layer transmissivities will be calculated using confined VKD relationships** (& arithmetic averaging).

Item: 1
Variable: **ITRANS**
Value: **1**
Option: **Calculated inter-nodal transmissivities will be written to the listing file.**

Item: 1
Variable: **ITRPY**
Value: **0**
Option: **A single value for anisotropy will be read for each layer (item 3).**

Item: 1
Variable: **IHOLD**
Value: **0**
Option: **Transmissivities will be calculated using heads from the current time step.**

Items: 3, 4 & 5
Variables: TRPY, DELR & DELC
Values: 1, 500 m & 500 m

Item: 7
Variable: Tran
Value: 50 to 2050 m^2/d

Item: 9
Variable: BOT
Value: 36.8 to 120.22 m

```
  36.84  36.84  36.80  36.69  40.97  45.24  49.52  53.79  58.06  62.32  66.59  70.84  75.09  79.34  83.58  87.81  92.03  96.24 100.42 104.58 108.70 112.76 116.25 118.00
  36.84  36.84  36.80  36.69  40.97  45.24  49.52  53.79  58.06  62.32  66.59  70.84  75.09  79.34  83.58  87.81  92.03  96.24 100.42 104.58 108.70 112.76 116.69 120.00
  36.84  36.84  36.80  36.69  40.97  45.24  49.52  53.79  58.06  62.32  66.59  70.84  75.09  79.34  83.58  87.81  92.03  96.24 100.42 104.58 108.70 112.76 116.69 120.22
  36.84  36.84  36.80  36.69  40.97  45.24  49.52  53.79  58.06  62.32  66.59  70.84  75.09  79.34  83.58  87.81  92.03  96.24 100.42 104.58 108.70 112.76 116.69 120.22
  36.84  36.84  36.80  36.69  40.97  45.24  49.52  53.79  58.06  62.32  66.59  70.84  75.09  79.34  83.58  87.81  92.03  96.24 100.42 104.58 108.70 112.76 116.69 120.22
  36.84  36.84  36.80  36.69  40.97  45.24  49.52  53.79  58.06  62.32  66.59  70.84  75.09  79.34  83.58  87.81  92.03  96.24 100.42 104.58 108.70 112.76 116.69 120.22
  36.84  36.84  36.80  36.69  40.97  45.24  49.52  53.79  58.06  62.32  66.59  70.84  75.09  79.34  83.58  87.81  92.03  96.24 100.42 104.58 108.70 112.76 116.69 120.22
  36.84  36.84  36.80  36.69  40.97  45.24  49.52  53.79  58.06  62.32  66.59  70.84  75.09  79.34  83.58  87.81  92.03  96.24 100.42 104.58 108.70 112.76 116.69 120.22
  36.84  36.84  36.80  36.69  40.97  45.24  49.52  53.79  58.06  62.32  66.59  70.84  75.09  79.34  83.58  87.81  92.03  96.24 100.42 104.58 108.70 112.76 116.69 120.22
  36.84  36.84  36.80  36.69  40.97  45.24  49.52  53.79  58.06  62.32  66.59  70.84  75.09  79.34  83.58  87.81  92.03  96.24 100.42 104.58 108.70 112.76 116.69 120.22
  36.84  36.84  36.80  36.69  40.97  45.24  49.52  53.79  58.06  62.32  66.59  70.84  75.09  79.34  83.58  87.81  92.03  96.24 100.42 104.58 108.70 112.76 116.69 120.22
  36.84  36.84  36.80  36.69  40.97  45.24  49.52  53.79  58.06  62.32  66.59  70.84  75.09  79.34  83.58  87.81  92.03  96.24 100.42 104.58 108.70 112.76 116.69 120.22
  36.84  36.84  36.80  36.69  40.97  45.24  49.52  53.79  58.06  62.32  66.59  70.84  75.09  79.34  83.58  87.81  92.03  96.24 100.42 104.58 108.70 112.76 116.69 120.22
       11     1.000(25f6.1)                   0     TOP Layer   1
 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0
 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 120.0 119.0 118.0 117.0 116.0 115.0 114.0 113.0 112.0
 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 121.0 120.0 119.0 118.0 117.0 116.0 115.0 114.0
 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 122.0 121.0 120.0 119.0 118.0 117.0 116.0
 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 123.0 122.0 121.0 120.0 119.0 118.0
 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 124.0 123.0 122.0 121.0 120.0
 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 126.0 125.0 124.0 123.0 122.0
 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 127.0 126.0 125.0 124.0
 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 128.0 127.0 126.0
 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999 [illegible] [illegible] [illegible] [illegible] 999.0 999.0 999.0 999.0 129.0 128.0
 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999 [illegible] [illegible] [illegible] [illegible] 999.0 999.0 999.0 999.0 999.0 130.0
 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999 [illegible] [illegible] [illegible] [illegible] 999.0 999.0 999.0 999.0 999.0 999.0
 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999 [illegible] [illegible] [illegible] [illegible] 999.0 999.0 999.0 999.0 999.0 999.0
 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0
 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0
 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0
 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0
 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0
 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0
 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0
 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0
 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0 999.0
```

Item: 12
Variable: TOP
Value: 112 to 999 m

```
      11    1.000(25f6.2)                    0     THICKNESS OF UPPER ZONE Layer   1
10.00 10.00 10.00 10.00 10.00 10.00 10.00 10.00 10.00 10.00 10.00 10.00 10.00 10.00 10.00 10.00 10.00 10.00 10.00 10.00 10.00 10.00 10.00 10.00
10.00 10.00 10.00 10.00  9.75  9.50  9.25  9.00  8.75  8.50  8.25  8.00  7.75  7.50  7.25  7.00  6.75  6.50  6.25  6.00  5.75  5.50  5.25  5.00
10.00 10.00 10.00 10.00  9.75  9.50  9.25  9.00  8.75  8.50  8.25  8.00  7.75  7.50  7.25  7.00  6.75  6.50  6.25  6.00  5.75  5.50  5.25  5.00
10.00 10.00 10.00 10.00  9.75  9.50  9.25  9.00  8.75  8.50  8.25  8.00  7.75  7.50  7.25  7.00  6.75  6.50  6.25  6.00  5.75  5.50  5.25  5.00
10.00 10.00 10.00 10.00  9.75  9.50  9.25  9.00  8.75  8.50  8.25  8.00  7.75  7.50  7.25  7.00  6.75  6.50  6.25  6.00  5.75  5.50  5.25  5.00
10.00 10.00 10.00 10.00  9.75  9.50  9.25  9.00  8.75  8.50  8.25  8.00  7.75  7.50  7.25  7.00  6.75  6.50  6.25  6.00  5.75  5.50  5.25  5.00
10.00 10.00 10.00 10.00  9.75  9.50  9.25  9.00  8.75  8.50  8.25  8.00  7.75  7.50  7.25  7.00  6.75  6.50  6.25  6.00  5.75  5.50  5.25  5.00
10.00 10.00 10.00 10.00  9.75  9.50  9.25  9.00  8.75  8.50  8.25  8.00  7.75  7.50  7.25  7.00  6.75  6.50  6.25  6.00  5.75  5.50  5.25  5.00
10.00 10.00 10.00 10.00  9.75  9.50  9.25  9.00  8.75  8.50  8.25  8.00  7.75  7.50  7.25  7.00  6.75  6.50  6.25  6.00  5.75  5.50  5.25  5.00
10.00 10.00 10.00 10.00  9.75  9.50  9.25  9.00  8.75  8.50  8.25  8.00  7.75  7.50  7.25  7.00  6.75  6.50  6.25  6.00  5.75  5.50  5.25  5.00
10.00 10.00 10.00 10.00  9.75  9.50  9.25  9.00  8.75  8.50  8.25  8.00  7.75  7.50  7.25  7.00  6.75  6.50  6.25  6.00  5.75  5.50  5.25  5.00
10.00 10.00 10.00 10.00  9.75  9.50  9.25  9.00  8.75  8.50  8.25  8.00  7.75  7.50  7.25  7.00  6.75  6.50  6.25  6.00  5.75  5.50  5.25  5.00
10.00 10.00 10.00 10.00  9.75  9.50  9.25  9.00  8.75  8.50  8.25  8.00  7.75  7.50  7.25  7.00  6.75  6.50  6.25  6.00  5.75  5.50  5.25  5.00
10.00 10.00 10.00 10.00  9.75  9.50  9.25  9.00  8.75  8.50  8.25  8.00  7.75  7.50  7.25  7.00  6.75  6.50  6.25  6.00  5.75  5.50  5.25  5.00
10.00 10.00 10.00 10.00  9.75  9.50  9.25  9.00  8.75  8.50  8.25  8.00  7.75  7.50  7.25  7.00  6.75  6.50  6.25  6.00  5.75  5.50  5.25  5.00
10.00 10.00 10.00 10.00  9.75  9.50  9.25  9.00  8.75  8.50  8.25  8.00  7.75  7.50  7.25  7.00  6.75  6.50  6.25  6.00  5.75  5.50  5.25  5.00
10.00 10.00 10.00 10.00  9.75  9.50  9.25  9.00  8.75  8.50  8.25  8.00  7.75  7.50  7.25  7.00  6.75  6.50  6.25  6.00  5.75  5.50  5.25  5.00
10.00 10.00 10.00 10.00  9.75  9.50  9.25  9.00  8.75  8.50  8.25  8.00  7.75  7.50  7.25  7.00  6.75  6.50  6.25  6.00  5.75  5.50  5.25  5.00
10.00 10.00 10.00 10.00  9.75  9.50  9.25  9.00  8.75  8.50  8.25  8.00  7.75  7.50  7.25  7.00  6.75  6.50  6.25  6.00  5.75  5.50  5.25  5.00
10.00 10.00 10.00 10.00  9.75  9.50  9.25  9.00  8.75  8.50  8.25  8.00  7.75  7.50  7.25  7.00  6.75  6.50  6.25  6.00  5.75  5.50  5.25  5.00
10.00 10.00 10.00 10.00  9.75  9.50  9.25  9.00  8.75  8.50  8.25  8.00  7.75  7.50  7.25  7.00  6.75  6.50  6.25  6.00  5.75  5.50  5.25  5.00
10.00 10.00 10.00 10.00  9.75  9.50  9.25  9.00  8.75  8.50  8.25  8.00  7.75  7.50  7.25  7.00  6.75  6.50  6.25  6.00  5.75  5.50  5.25  5.00
       0    0.600(10E12.4)                   0     HYDRAULIC CONDUCTIVITY GRADIENT FACTOR Layer 1
       0  100.000(10E12.4)                   0     MAXIMUM HYDRAULIC CONDUCTIVITY FACTOR Layer 1
```

Item: 16
Variable: VKMAX
Value: 100
Option: The maximum value of hydraulic conductivity in the upper part of the VKD profile will be limited to 100 times the base hydraulic conductivity.

Item: 15
Variable: VKGRAD
Value: 0.6
Option: The slope of the hydraulic conductivity gradient in the upper part of the VKD profile will be 0.6 times the base hydraulic conductivity value per meter.

Item: 14
Variable: VMID
Value: from 5 to 10 m
Option: The thickness of the upper part of the VKD profile will vary between 5 and ten meters.

2.4 Module documentation for BCF package

The modified Block-Centred Flow Package (BCF-VKD1) has two new primary modules and three new submodules (highlighted in **bold** below). Modifications have also been made to all of the original modules. The relationships of the modules to the main program and to each other is shown in Figure 2.7.

Primary Modules

BCF5AL	Allocates space for data arrays.
BCF5RP	Reads all data needed by the package, invokes SBCF1N to reconcile input transmissive values with the IBOUND array, and calculates storage capacities and constant conductances.
BCF5AD	Set HOLD to BOT whenever a wettable cell is dry.
BCF5FM	Calculates all coefficients of the system of equations that are not constant and invokes SBCF5H to calculate horizontal branch conductances in partially saturated layers.
BCF5VK	**Writes new BCF, BAS and NAME files for 2nd initial conditions for VKD run. Initial heads from previous (specified T or K) run, top and bottom elevations calculated from heads and upper thicknesses, ISS set to 1, return to start of simulation.**
BCF5OT	**Print and record internodal transmissivities.**

Submodules

SBCF5N	Reconciles input transmissive values with the IBOUND array and calculates storage capacities and constant conductances. Invokes SBCF5C, A, L or I to calculate horizontal branch conductances for layers where transmissivity is constant.
SBCF5H	Calculates transmissivity for cells in layers where it depends on heads and invokes SBCF5C, A, L, U or I to calculate horizontal branch conductances.
SBCF5C	Calculates branch conductances using harmonic mean of cell transmissivities (activated by LAYAVG = 00).
SBCF5A	Calculates branch conductances using arithmetic mean of cell transmissivities (activated by LAYAVG = 10).
SBCF5L	Calculates branch conductances using logarithmic mean of cell transmissivities (activated by LAYAVG = 20).
SBCF5U	Calculates branch conductances using arithmetic mean saturated thickness and logarithmic mean hydraulic conductivity (activated by LAYAVG = 30).
SBCF5I	**Calculates branch conductances using internodal transmissivity (activated by LAYAVG = 40).**
SBCF5B	Calculates flow between adjacent cells in a subregion of the grid.
SBCF5S	Calculates storage budget flow term for BCF.
SBCF5F	Calculates flow from constant-head cells.
B12DRI	**Routine to read 1 & 2-D real data input matrices and keep a record of the format.**
SBCF5W	**Routine to write arrays to new BCF and BAS packages. Sets all values at no-flow and dry cells to zero, checks remaining values to see if they are all equal and prints the array record to the new files.**

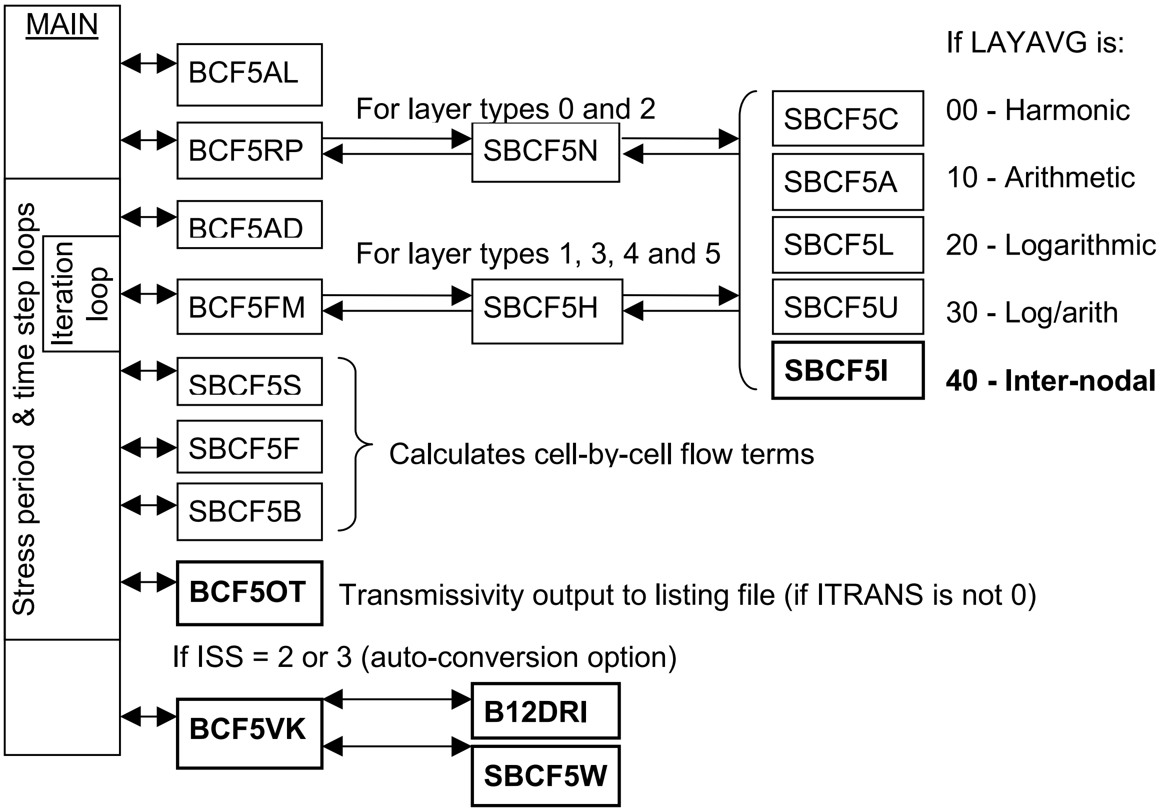
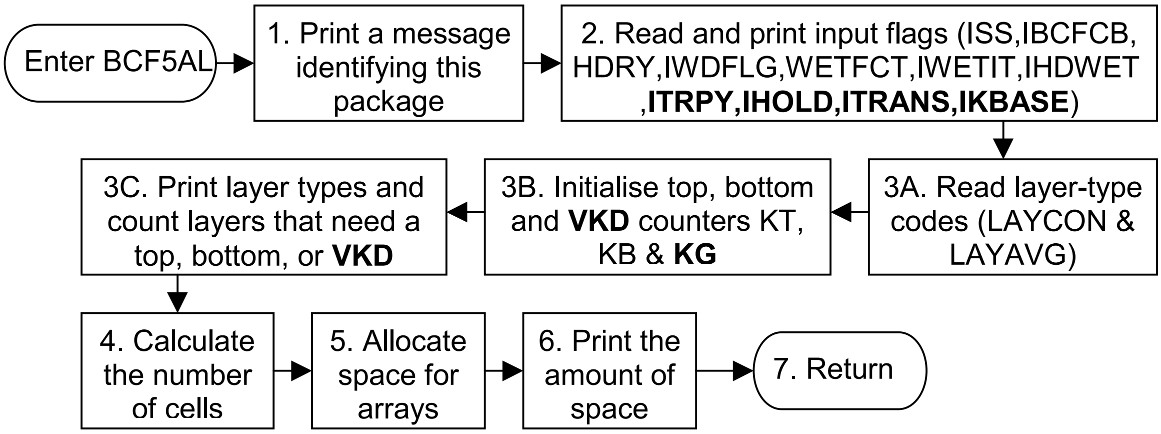


Figure 2.7 Relationship between the modules in the Block Centred Flow package

2.4.1 Changes to module BCF5AL

- Different title printed to listing file
- Read new input flags (ITRPY, IHOLD, ITRANS and IKBASE)
- Allow new layer types and interblock T code
- Allocate storage space for new arrays (expanded TRPY array, VKGRAD, VMID, VKMAX, VSGRAD, VSMID and VSMAX)

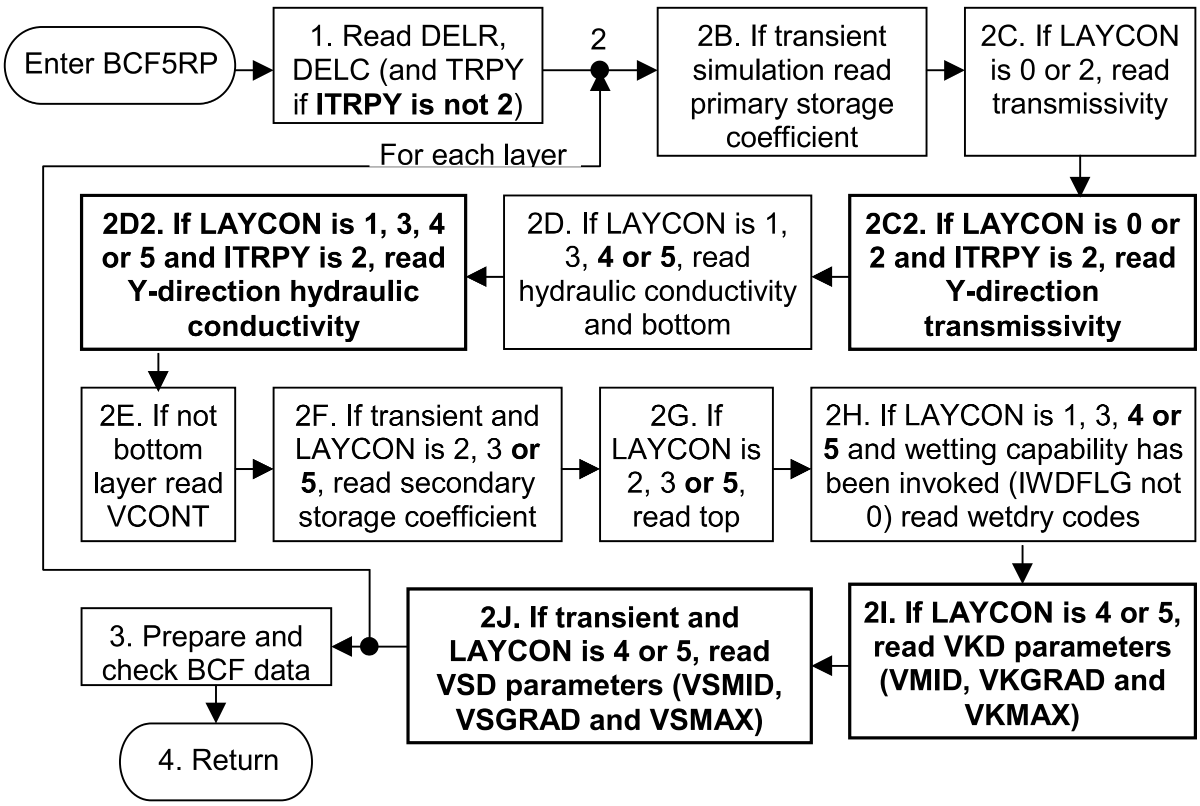
Flow Chart for module BCF5AL



2.4.2 Changes to module BCF5RP

- Add new array titles
- Allow new layer types
- Allow reading of arrays for spatially variable anisotropy
- Read new arrays (as for BCF5AL)

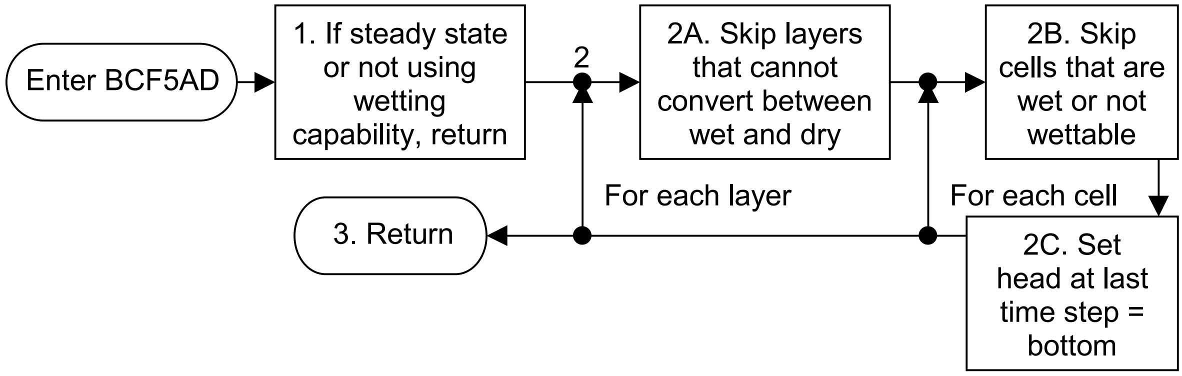
Flow chart for module BCF5RP



2.4.3 Changes to module BCF5AD

- Allow new layer types

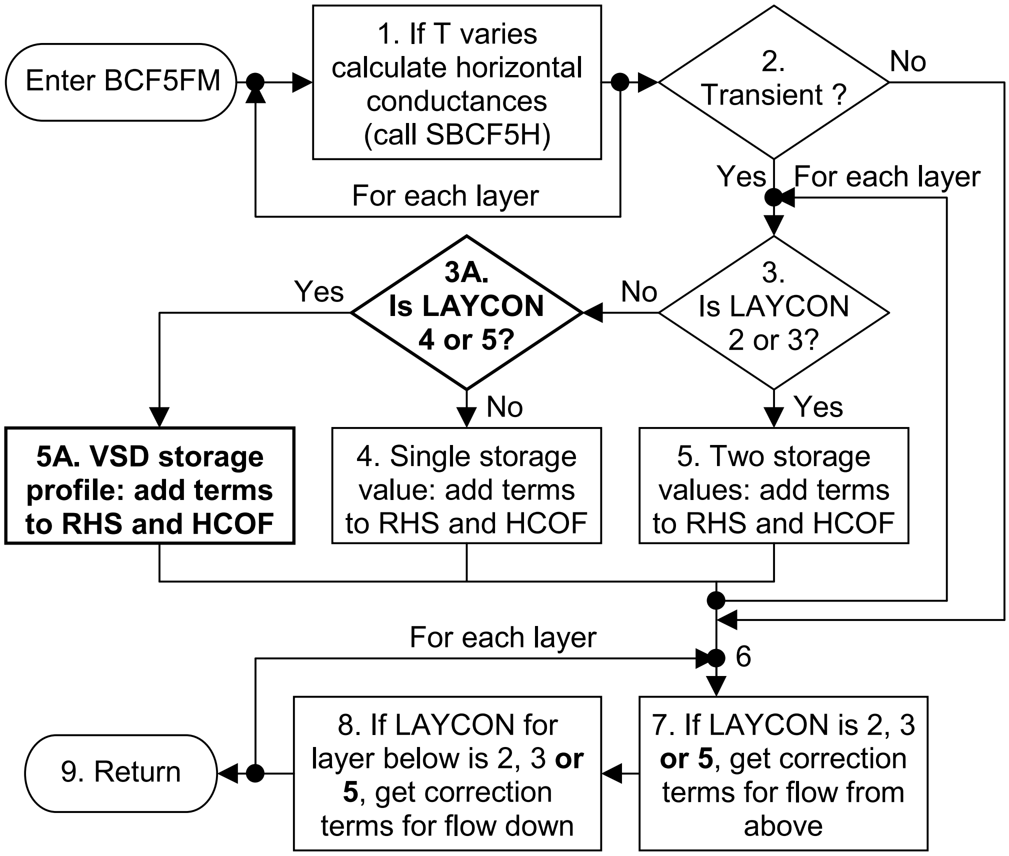
Flow chart for module BCF5AD



2.4.4 Changes to module BCF5FM

- Allow new layer types
- Include variable storage with depth calculations

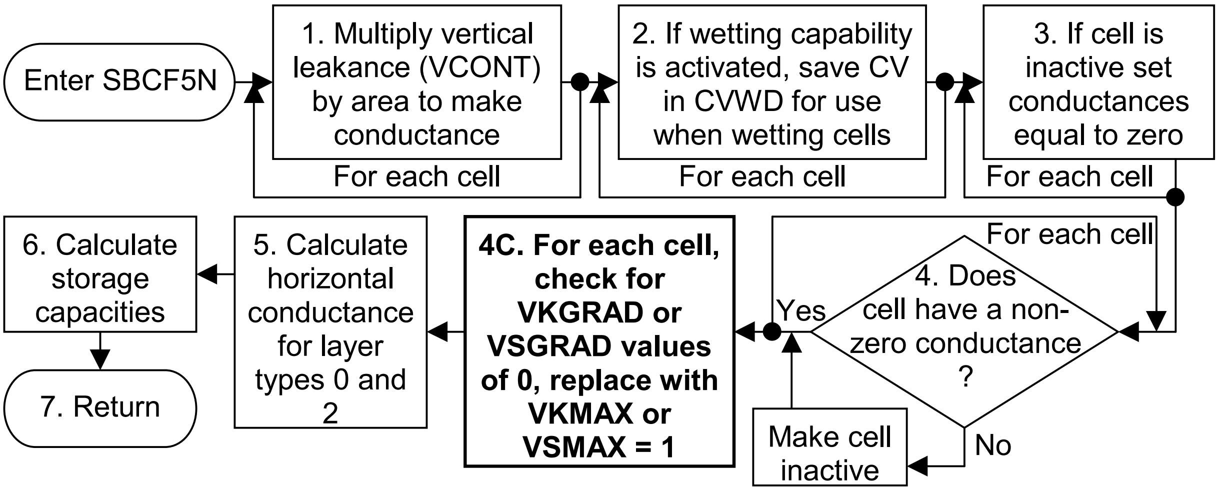
Flow chart for module BCF5FM



2.4.5 Changes to module SBCF5N

- Allow new layer types and interblock T code.
- Check all connecting branch conductances when using internodal transmissivity.
- Check for hydraulic conductivity gradient factor values of zero and replace with maximum hydraulic conductivity factor of one.
- Check for storage gradient factor values of zero and replace with maximum storage factor of one.

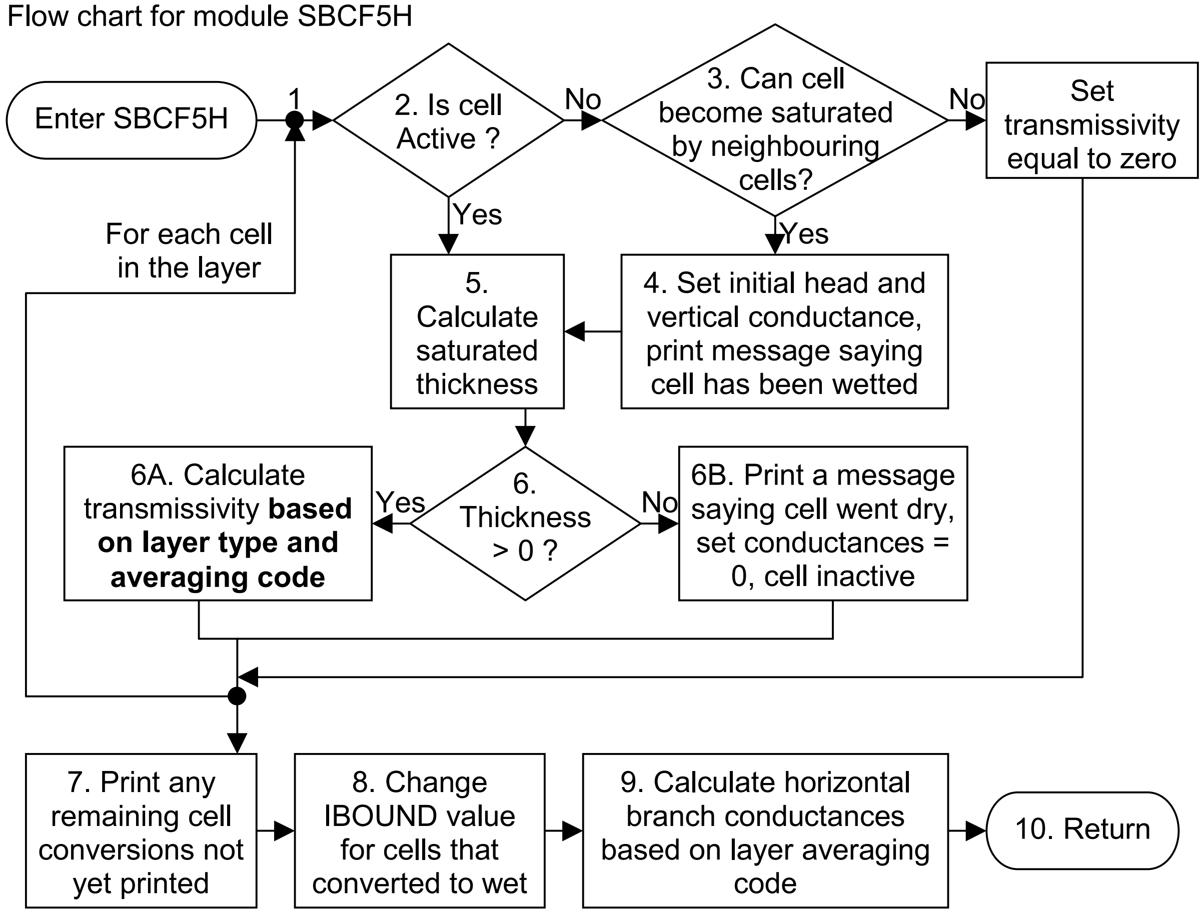
Flow chart for module SBCF5N



2.4.6 Changes to module SBCF5H

- Allow new layer types and interblock T code
- Allow explicit transmissivity option
- Include variable hydraulic conductivity with depth (VKD) calculations
- Include calculations for internodal transmissivity option (interblock T code)
- Allow spatially variable anisotropy

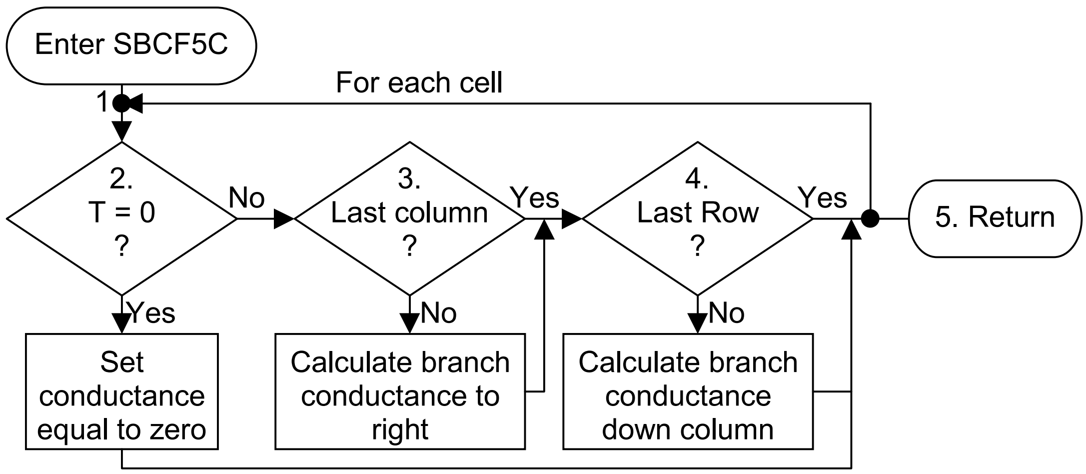
Flow chart for module SBCF5H



2.4.7 Changes to modules SBCF5C, SBCF5A, SBCF5L and SBCF5U

- Allow spatially variable anisotropy

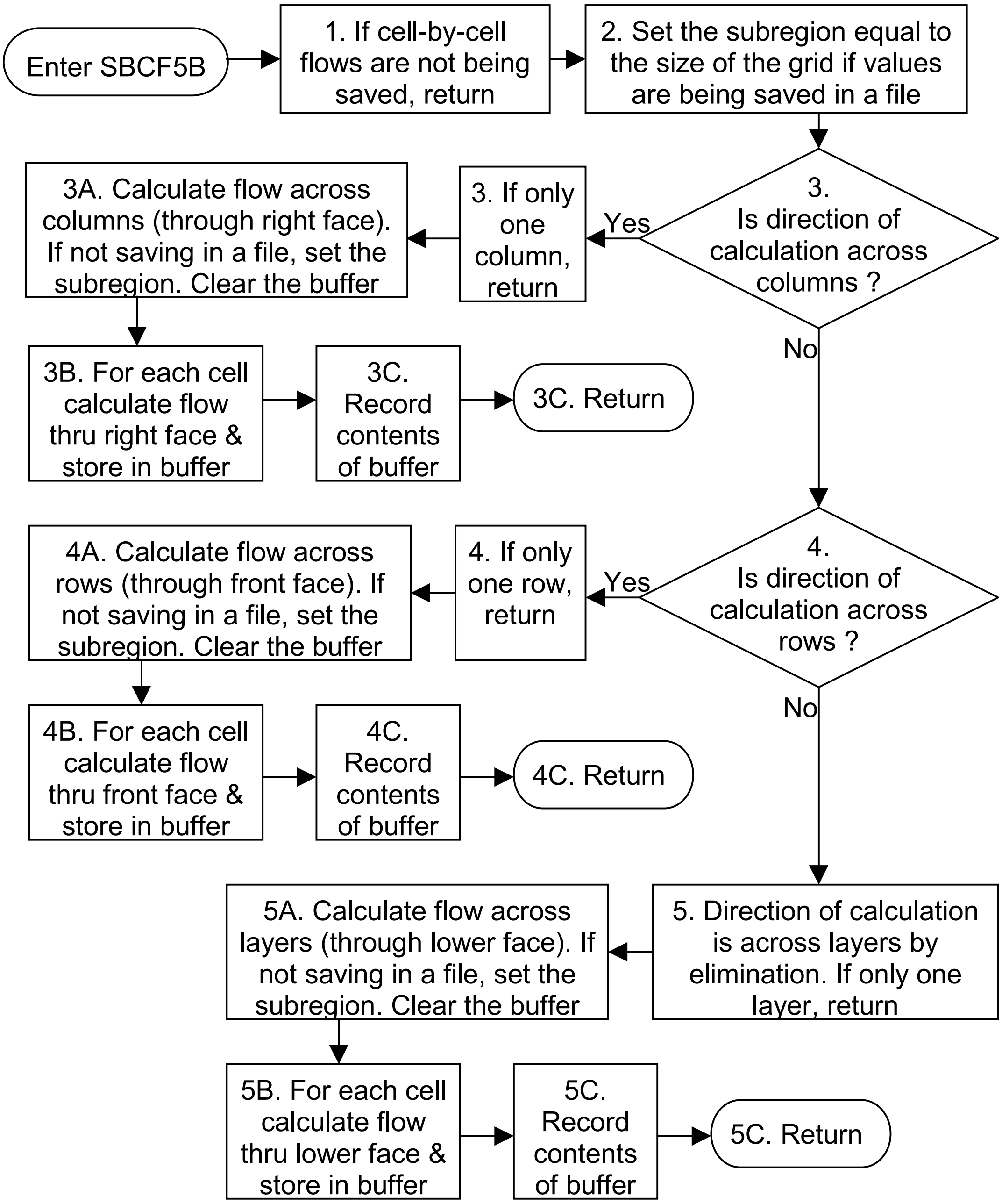
Flow chart for modules SBCF5C, SBCF5A, SBCF5L, SBCF5U and SBCF5I



2.4.8 Changes to module SBCF5B

- Allow new layer types

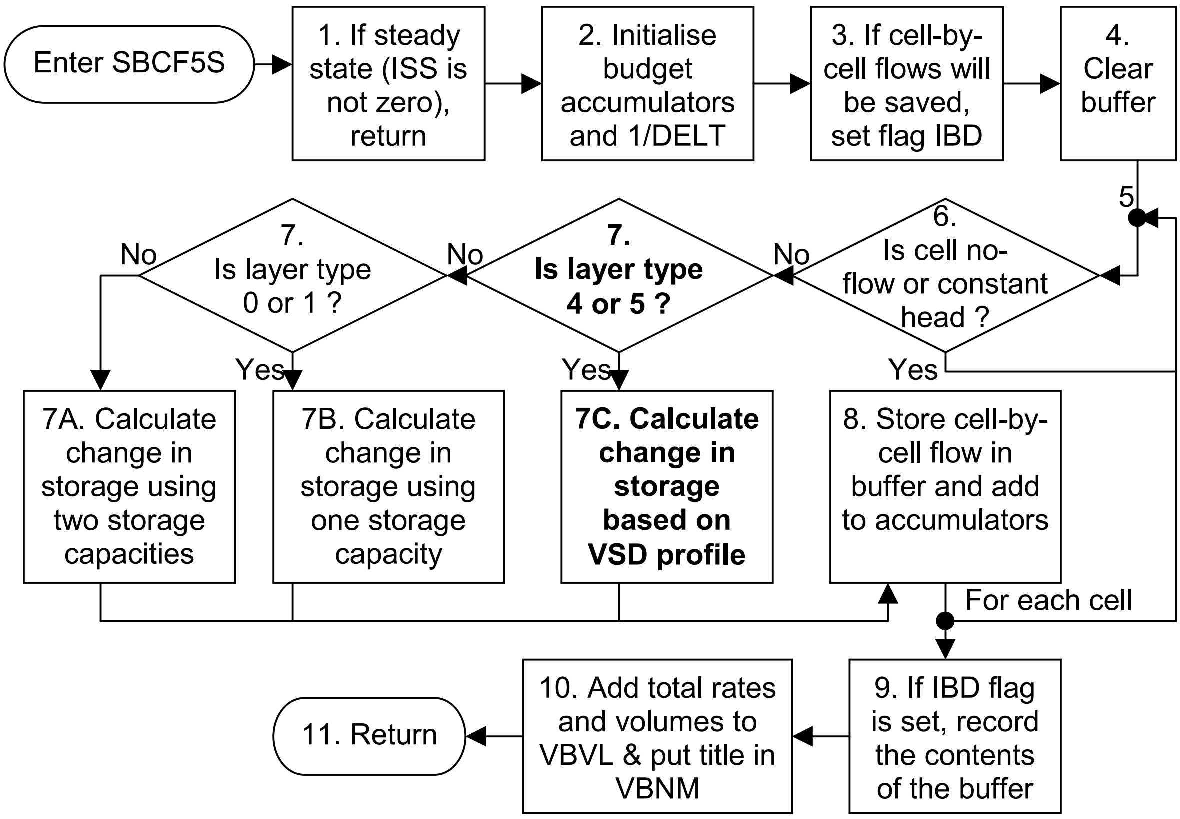
Flow chart for module SBCF5B



2.4.9 Changes to module SBCF5S

- Allow new layer types
- Include variable storage with depth calculations

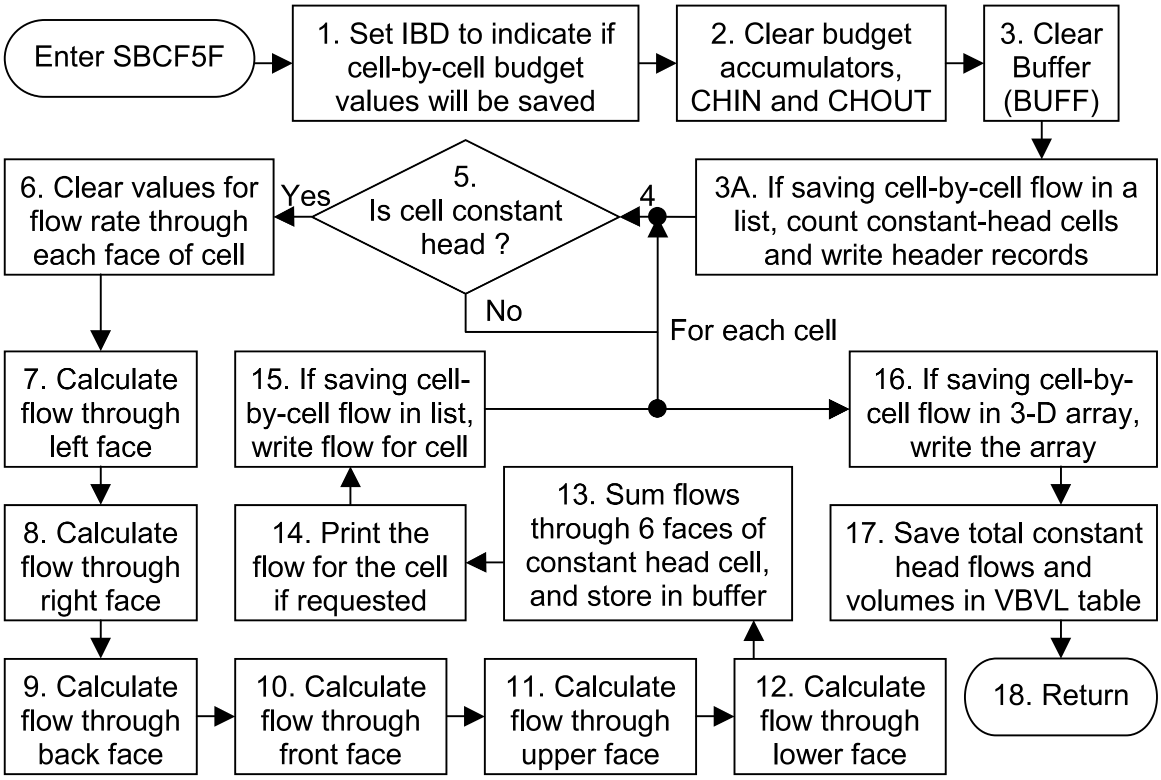
Flow chart for module SBCF5S



2.4.10 Changes to module SBCF5F

- Allow new layer types

Flow chart for module SBCF5F



2.4.11 Narrative for module BCF5VK

This module opens the name file, BAS and BCF files, reads the contents of the files, calculates the VKD parameters from the results of the first simulation (constant transmissivity or hydraulic conductivity simulation) and writes the new name, BAS and BCF files for the second simulation. Module BCF5VK is called by the main program after completion of the first simulation if the steady state flag (ISS) is 2 or 3. The module calls utility modules URWORD, U2DINT and ULAPRW, and the new BCF modules B12DRI and SBCF5W.

Module BCF5VK performs its functions in the following order:

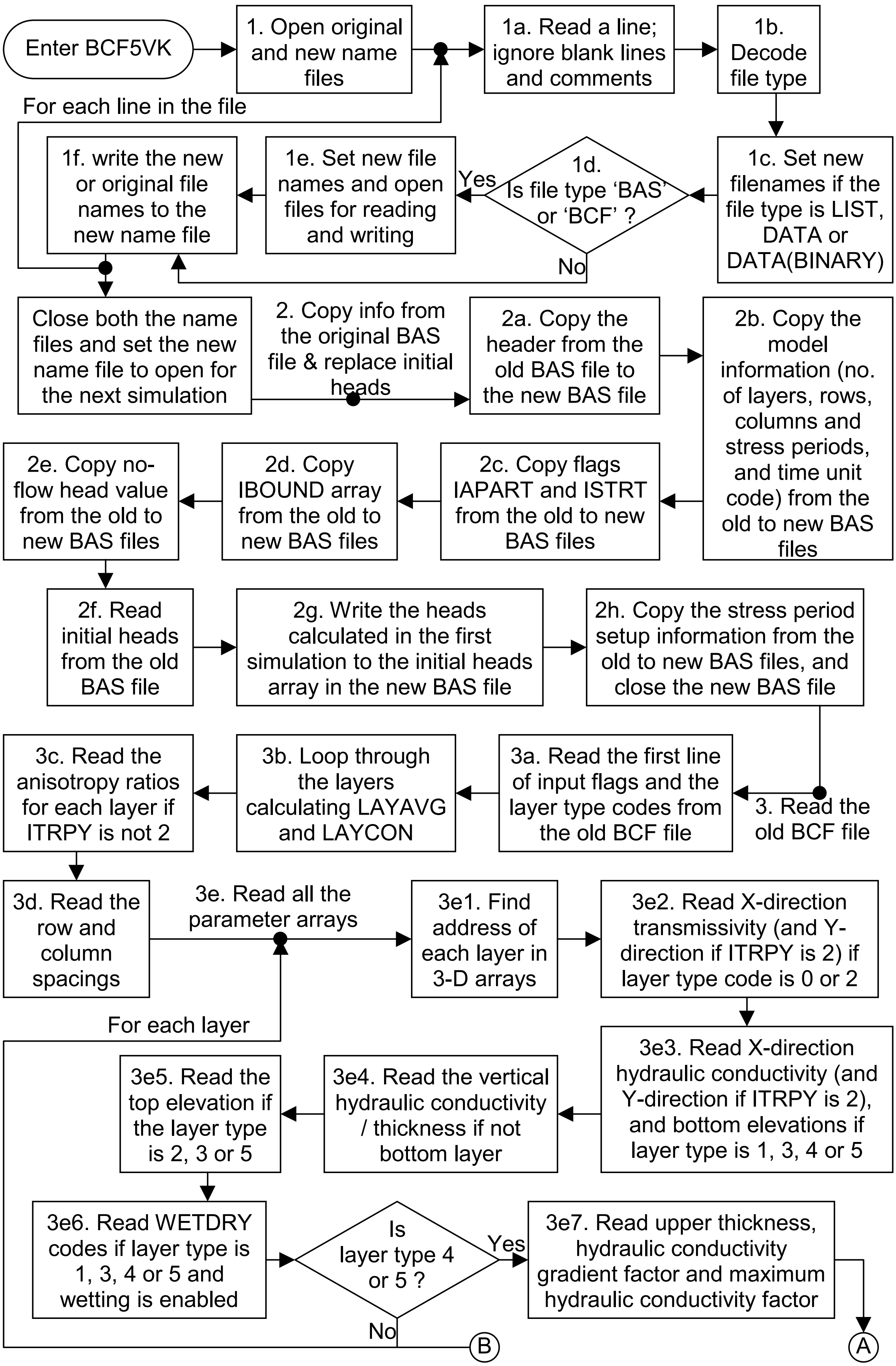
1. Open original and new name files (code based on BAS module SBAS5O)
   a. Read a line; ignore blank lines and comments (denoted by '#' character)
   b. decode the file type
   c. if the file type is 'LIST', 'DATA' or 'DATA(BINARY)' set new filenames
   d. if the file type is 'BAS' or 'BCF' go to step 1e, otherwise go to step 1f
   e. set new filenames for new BAS and BCF input files, and open the original and new files ready for reading and writing
   f. write the new (or original) filenames to the new name file, return to step 1a if there is another line, otherwise close both the name files and set the new name file to open for the next simulation
2. Read the original BAS file and write a new BAS file with initial heads from the first simulation (code based on BAS modules BAS5DF, BAS5AL & BAS5RP):
   a. Copy the text header from the old BAS file to the new BAS file
   b. Copy the model information (number of layers, rows, columns and stress periods, and the time unit code) from the old BAS file to the new BAS file
   c. Copy flags IAPART and ISTRT from the old BAS file to the new BAS file
   d. Copy IBOUND array from the old BAS file to the new BAS file
   e. Copy no-flow head value from the old BAS file to the new BAS file
   f. Read initial heads from the old BAS file
   g. Write the heads calculated in the first simulation to the initial heads array in the new BAS file
   h. Copy the stress period setup information from the old BAS file to the new BAS file and close the new BAS file
3. Read the old BCF file (code based on BCF modules BCF5AL and BCF5RP)
   a. Read ISS (steady-state flag), IBCFCB (flag for printing or unit number for recording cell-by-cell flow terms), HDRY (head at cells that convert to dry), wetting parameters, anisotropy option, explicit transmissivity calculation option, transmissivity output option, and layer type codes
   b. Loop through layers calculating LAYAVG and LAYCON
   c. Read the anisotropy ratios for each layer if ITRPY is not 2
   d. Read the row and column spacings
   e. Read all the parameters for each layer
      1. Find address of each layer in 3-D arrays
      2. Read X-direction transmissivity into array CC if layer type code is 0 or 2 (if ITRPY is 2, copy CC into array CR and read Y-direction transmissivity into array CC)
      3. Read X-direction hydraulic conductivity into array HY (and Y-direction hydraulic conductivity into array TRPY if ITRPY is 2) and bottom elevations into array BOT if layer type code is 1, 3, 4 or 5
      4. Read the vertical hydraulic conductivity / thickness into array CV if not bottom layer
      5. Read top elevation into array TOP if layer type code is 2, 3 or 5
      6. Read the WETDRY codes if layer type is 1, 3, 4 or 5 and wetting capability has been invoked (IWDFLG is not zero)

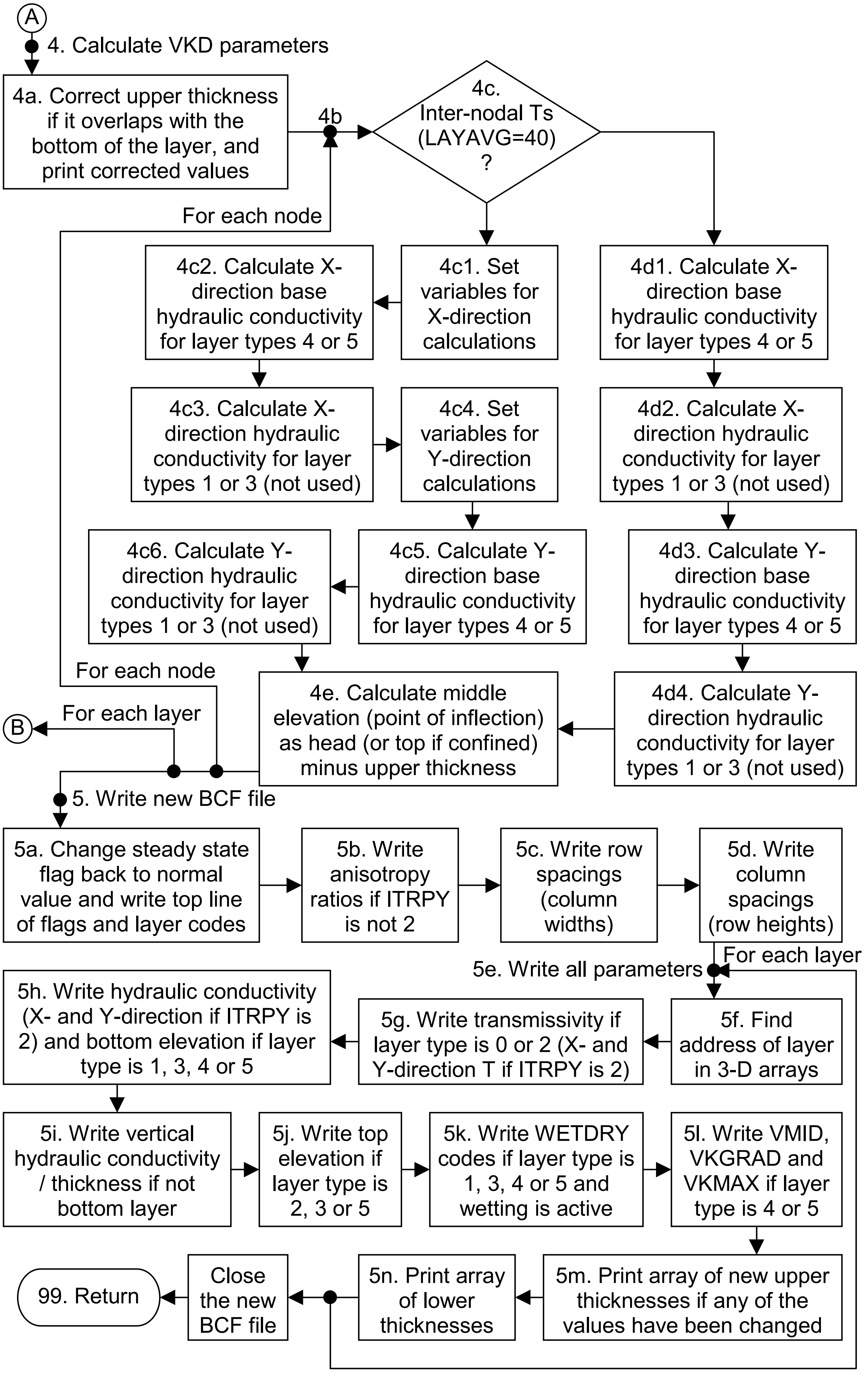
7. Read upper thickness into array VMID, hydraulic conductivity gradient factor into array VKGRAD, and maximum hydraulic conductivity factor into array VKMAX if layer type is 4 or 5

4. Calculate VKD parameters (base hydraulic conductivities, anisotropy ratios, and middle elevations (point of inflection)) if layer type is 4 or 5
   a. Correct upper thickness if it overlaps with the bottom of the layer (ie, if head minus upper thickness is less than the bottom elevation) and write corrected values to the output file for the first simulation
   b. Loop through the cells in the layer calculating the VKD parameters:
   c. Internodal transmissivity option:
      1. Set variables for X-direction calculations
      2. Calculate X-direction base hydraulic conductivities for layer types 4 or 5
      3. Calculate X-direction hydraulic conductivities for layer types 1 or 3 (not used)
      4. Set variables for Y-direction calculations
      5. Calculate Y-direction base hydraulic conductivities for layer types 4 or 5
      6. Calculate Y-direction hydraulic conductivities for layer types 1 or 3 (not used)
   d. Block centred transmissivity option:
      1. Calculate X-direction base hydraulic conductivities for layer types 4 or 5
      2. Calculate X-direction hydraulic conductivities for layer types 1 or 3 (not used)
      3. Calculate Y-direction base hydraulic conductivities for layer types 4 or 5
      4. Calculate Y-direction hydraulic conductivities for layer types 1 or 3 (not used)
   e. Calculate middle elevation (point of inflection, VMID) as head (or top if confined) minus upper thickness
5. Write new BCF file
   a. Change steady state flag (ISS) back to normal steady state value (1) and write top lines of flags and layer codes
   b. Write anisotropy ratios if ITRPY is not 2
   c. Write row spacings (column widths)
   d. Write column spacings (row heights)
   e. Write all parameters for each layer (f to n):
   f. Find address of each layer in 3-D arrays
   g. Write transmissivity if layer type is 0 or 2 (X- and Y-direction transmissivity if ITRPY is 2)
   h. Write hydraulic conductivity (X- and Y-direction if ITRPY is 2) and bottom elevation if layer type is 1, 3, 4 or 5
   i. Write vertical hydraulic conductivity / thickness if not bottom layer
   j. Write top elevation if layer type is 2, 3 or 5
   k. Write WETDRY codes if layer type is 1, 3, 4 or 5 and wetting capability has been invoked (IWDFLG is not 0)
   l. Write middle elevation (point of inflection, VMID), hydraulic conductivity gradient factor (VKGRAD) and maximum hydraulic conductivity factor (VKMAX) if layer type is 4 or 5.
   m. Write array of new upper thicknesses to the first output file if any of the values have been changed
   n. Write array of lower thicknesses to the first output file and close the new BCF file

      99. RETURN

Flow chart for module BCF5VK




A
4. Calculate VKD parameters
4a. Correct upper thickness if it overlaps with the bottom of the layer, and print corrected values
4b
4c. Inter-nodal Ts (LAYAVG=40) ?
For each node
4c1. Set variables for X-direction calculations
4c2. Calculate X-direction base hydraulic conductivity for layer types 4 or 5
4c3. Calculate X-direction hydraulic conductivity for layer types 1 or 3 (not used)
4c4. Set variables for Y-direction calculations
4c5. Calculate Y-direction base hydraulic conductivity for layer types 4 or 5
4c6. Calculate Y-direction hydraulic conductivity for layer types 1 or 3 (not used)
4d1. Calculate X-direction base hydraulic conductivity for layer types 4 or 5
4d2. Calculate X-direction hydraulic conductivity for layer types 1 or 3 (not used)
4d3. Calculate Y-direction base hydraulic conductivity for layer types 4 or 5
4d4. Calculate Y-direction hydraulic conductivity for layer types 1 or 3 (not used)
For each node
For each layer
B
4e. Calculate middle elevation (point of inflection) as head (or top if confined) minus upper thickness
5. Write new BCF file
5a. Change steady state flag back to normal value and write top line of flags and layer codes
5b. Write anisotropy ratios if ITRPY is not 2
5c. Write row spacings (column widths)
5d. Write column spacings (row heights)
For each layer
5e. Write all parameters
5f. Find address of layer in 3-D arrays
5g. Write transmissivity if layer type is 0 or 2 (X- and Y-direction T if ITRPY is 2)
5h. Write hydraulic conductivity (X- and Y-direction if ITRPY is 2) and bottom elevation if layer type is 1, 3, 4 or 5
5i. Write vertical hydraulic conductivity / thickness if not bottom layer
5j. Write top elevation if layer type is 2, 3 or 5
5k. Write WETDRY codes if layer type is 1, 3, 4 or 5 and wetting is active
5l. Write VMID, VKGRAD and VKMAX if layer type is 4 or 5
5m. Print array of new upper thicknesses if any of the values have been changed
5n. Print array of lower thicknesses
Close the new BCF file
99. Return

List of variables for module BCF5VK

Variable	Range	Definition
AHDRY	Module	Character representation of HDRY variable (character*10)
ANAME	Module	DIMENSION(20) Labels for printout of input array (character*24 array)
ATMP	Module	Character representation of TMP variable (head at no-flow cells) (character*10)
AWETFCT	Module	Character representation of WETFCT variable (character*10)
BBOT	Module	Temporary variable to store bottom elevation for a node (real)
BOT	Package	DIMENSION (NCOL,NROW,NBOT), Elevation of bottom of each layer. (NBOT is the number of layers for which LAYCON is 1, 3, 4 or 5.) (real array)
BUFF	Global	DIMENSION (NCOL,NROW,NLAY), Buffer array with one element for each cell in the model. (real array)
CC	Package	DIMENSION (NCOL,NROW,NLAY), Conductance in the column direction. CC(J,I,K) contains conductance between nodes (J,I,K) and (J,I+1,K). (real array)
CR	Package	DIMENSION (NCOL,NROW,NLAY), Conductance in the row direction. CR(J,I,K) contains conductance between nodes (J,I,K) and (J+1,I,K). (real array)
CUNIT	Global	Character identifier for the packages listed in the name file (character*4)
CV	Package	DIMENSION (NCOL,NROW,NLAY-1), Conductance in the vertical direction. CV(J,I,K) contains conductance between nodes (J,I,K) and (J,I,K+1). (real array)
DELC	Package	DIMENSION (NROW), Cell dimension in the column direction. DELC(I) contains width of row I. (real array)
DELR	Package	DIMENSION (NCOL), Cell dimension in the row direction. DELC(J) contains width of column J. (real array)
FMTIN	Module	DIMENSION (15,200), Input format for each array in each layer (character*20 array)
FNAM2	Module	Filename of the new name file (character*80)
FNAME	Global	Filename of the original name file (character*80)
HDRY	Module	Head value assigned to nodes that have become unsaturated and hence inactive (real)
HEADNG	Module	Dimension (2), Heading printed on output to identify the problem. (character*80 array)
HHNEW	Module	Temporary variable to store groundwater head for a node (double precision)

Variable	Range	Definition
HNEW	Global	DIMENSION (NCOL,NROW,NLAY), Most recent estimate of groundwater head in each cell (double precision array)
HOLD	Global	DIMENSION (NCOL,NROW,NLAY), groundwater head in each cell from last time step (real array)
HY	Package	DIMENSION (NCOL,NROW,NBOT), Hydraulic conductivity of a cell. (NBOT is the number of layers for which LAYCON is 1, 3, 4 or 5.) (real array)
I	Module	Index (integer*4)
IAPART	Module	Flag set by user. (integer*4) = 0, arrays RHS and BUFFER will share space in the X array. ≠ 0, arrays RHS and BUFFER will not share space in the X array.
IBASOUT	Global	Unit number that the new BASic file will be written to (set to unit 94 in the main program) (integer*4)
IBCFCB	Module	Flag and a unit number. (integer*4) > 0, unit number on which cell-by-cell flow terms will be recorded whenever ICBCFL is set. = 0, cell-by-cell flow terms will not be printed or recorded < 0, flow from each constant head cell will be printed whenever ICBCFL is set
IBCFOUT	Global	Unit number that the new BCF file will be written to (set to unit 96 in the main program) (integer*4)
IBOUND	Global	DIMENSION (NCOL,NROW,NLAY), Status of each cell. (integer*4 array) < 0, constant-head cell = 0, inactive (no-flow) cell > 0, variable-head cell
IERR	Module	Flag indicating whether an upper thickness had to be reduced (integer*4)
IFREFM	Global	Flag indicating whether free format input files are being used (integer*4)
IHDWET	Module	Flag that determines which equation is used to calculate the initial head at cells that become wet (integer*4)
IHOLD	Module	Flag (integer*4) = 0, transmissivity is calculated using the current head ≠ 0, transmissivity is calculated using the head from the last time step
III	Module	Index (integer*4)
INAM1	Module	Location on a line of the name file of the start of a filename (integer*4)

Variable	Range	Definition
INAM2	Module	Location on a line of the name file of the end of a filename (integer*4)
INBAS	Global	Primary unit number from which input to the BAS package will be read (integer*4)
INBCF	Package	Unit number from which input to the BCF package will be read (integer*4)
INOUT	Global	Unit number that the new name file will be written to (set to unit 95 in the main program) (integer*4)
INUNIT	Global	Unit number on which the original name file will be opened (set to unit 99 in the main program) (integer*4)
IOUT	Global	Primary unit number for all printed output (integer*4)
IP	Module	Index (integer*4)
IRECL	Module	Record length of binary files opened as direct access (integer*4)
ISS	Module	Flag (integer*4) = 0, simulation is transient (time-variant) ≠ 0, simulation is steady state
ISTART	Module	Location on a line of the name file of the start of a word or number (integer*4)
ISTOP	Module	Location on a line of the name file of the end of a word or number (integer*4)
ISTRT	Module	Flag (integer*4) = 0, starting heads will be saved so that drawdown can be calculated ≠ 0, starting heads will not be saved
ITRANS	Module	Flag (integer*4) = 0, inter-nodal transmissivity values will not be written to the output file ≠ 0, inter-nodal transmissivity values will be written to the output file every time step
ITRPY	Module	Flag (integer*4) = 0, a single value of the anisotropy ratio will be used for each layer = 2, X- and Y-direction transmissivities or hydraulic conductivities will be specified for each cell in each layer ≠ 0 or 2, a value of the anisotropy ratio will be used for each cell in each layer
ITYP1	Module	Location on a line of the name file of the start of a file type code (integer*4)
ITYP2	Module	Location on a line of the name file of the end of a file type code (integer*4)
IU	Module	Temporary store for a unit number (integer*4)
IU1	Module	Temporary store for a unit number (integer*4)
IU2	Module	Temporary store for a unit number (integer*4)

Variable	Range	Definition
IWDFLG	Module	Flag (integer*4) = 0, rewetting capability is not active ≠ 0, rewetting capability is active
IWETIT	Module	Iteration interval for attempting to wet cells (integer*4)
IXSEC	Global	Flag (integer*4) = 0, Model is not a cross section ≠ 0, Model is a cross section
J	Module	Index (integer*4)
JP	Module	Index (integer*4)
K	Module	Index for layers (integer*4)
KB	Module	Counter for the number of layers for which the bottom elevation is needed (LAYCON = 1, 3, 4 or 5) (integer*4)
KG	Module	Counter for the number of layers for which the gradient factor is needed (LAYCON = 4 or 5) (integer*4)
KK	Module	Temporary variable set equal to K (integer*4)
KT	Module	Counter for the number of layers for which the top elevation is needed (LAYCON = 2, 3, 4 or 5) (integer*4)
LAYAVG	Package	DIMENSION (200) Layer averaging code (integer*4 array): = 00, Harmonic averaging = 10, Arithmetic averaging = 20, Logarithmic averaging = 30, Log/arithmetic averaging = 40, Inter-nodal values used directly
LAYCON	Package	DIMENSION (200) Layer type code (integer*4 array): = 0, layer strictly confined = 1, Layer strictly unconfined = 2, Layer confined/unconfined (transmissivity is constant) = 3, Layer confined/unconfined (transmissivity depends on head)
LINE	Module	Line read from name file (character*80)
LINE1	Module	Temporary character string (character*80)
LINE2	Module	Temporary character string (character*80)
LLOC	Module	Counter specifying current position in a character string (integer*4)
N	Module	Temporary integer value (integer*4)
NCOL	Global	Number of columns in the grid (integer*4)
NLAY	Global	Number of layers in the grid (integer*4)
NPER	Global	Number of stress periods in the simulation (integer*4)
NROW	Global	Number of rows in the grid (integer*4)
NSTP	Module	Number of time steps in the current stress period (integer*4)

Variable	**Range**	**Definition**
PERLEN	Module	Length of the stress period (real)
PERLENNSTPTSMULT	Module	Character representation of the variables PERLEN, NSTP and TSMULT (character*30)
R	Module	Temporary real value (real)
SC1	Package	DIMENSION (NCOL,NROW,NLAY), Primary storage capacity for each cell (S*DELC*DELR) (real array)
SC2	Package	DIMENSION (NCOL,NROW,NTOP), Secondary storage capacity for each cell (real array) (NTOP is the number of layers for which LAYCON = 2, 3 or 5)
T	Module	Temporary variable for storing a transmissivity value for a cell (real)
TMP	Module	Temporary variable for storing the value of head for no-flow cells (real)
TOP	Package	DIMENSION (NCOL,NROW,NTOP), Elevation of the top of the layers (real array) (NTOP is the number of layers for which LAYCON = 2, 3 or 5)
TRPY	Package	DIMENSION (NCOL,NROW,NLAY), Anisotropy ratio: ratio of transmissivity in the column direction to transmissivity in the row direction (real array)
TSMULT	Module	Multiplier to get from one time step length to the next (real)
TTOP	Module	Temporary variable to store top elevation for a node (real)
VKGRAD	Package	DIMENSION (NCOL,NROW,NGRAD), Hydraulic conductivity gradient factor for each cell (NGRAD is the number of layers for which LAYCON = 4 or 5) (real array)
VKMAX	Package	DIMENSION (NCOL,NROW,NGRAD), Maximum hydraulic conductivity factor for each cell (NGRAD is the number of layers for which LAYCON = 4 or 5) (real array)
VMID	Package	DIMENSION (NCOL,NROW,NGRAD), Elevation of the point of inflection for each cell (NGRAD is the number of layers for which LAYCON = 4 or 5) (real array)
VVKGRAD	Module	Temporary variable to store the hydraulic conductivity gradient factor for a cell (real)
VVKMAX	Module	Temporary variable to store the maximum hydraulic conductivity factor for a cell (real)
VVMID	Module	Temporary variable to store the elevation of the point of inflection for a cell (real)

Variable	Range	Definition
WETDRY	Package	DIMENSION (NCOL,NROW,NLAY), Wetting threshold and flag (real array) < 0, only the cell below a dry cell can cause the cell to become wet > 0, the cell below a dry cell and four horizontally adjacent cells can cause a cell to become wet = 0, the cell cannot be wetted
WETFCT	Module	Factor that is included in the calculation of the head that is initially established at a cell when it is converted from dry to wet (real)

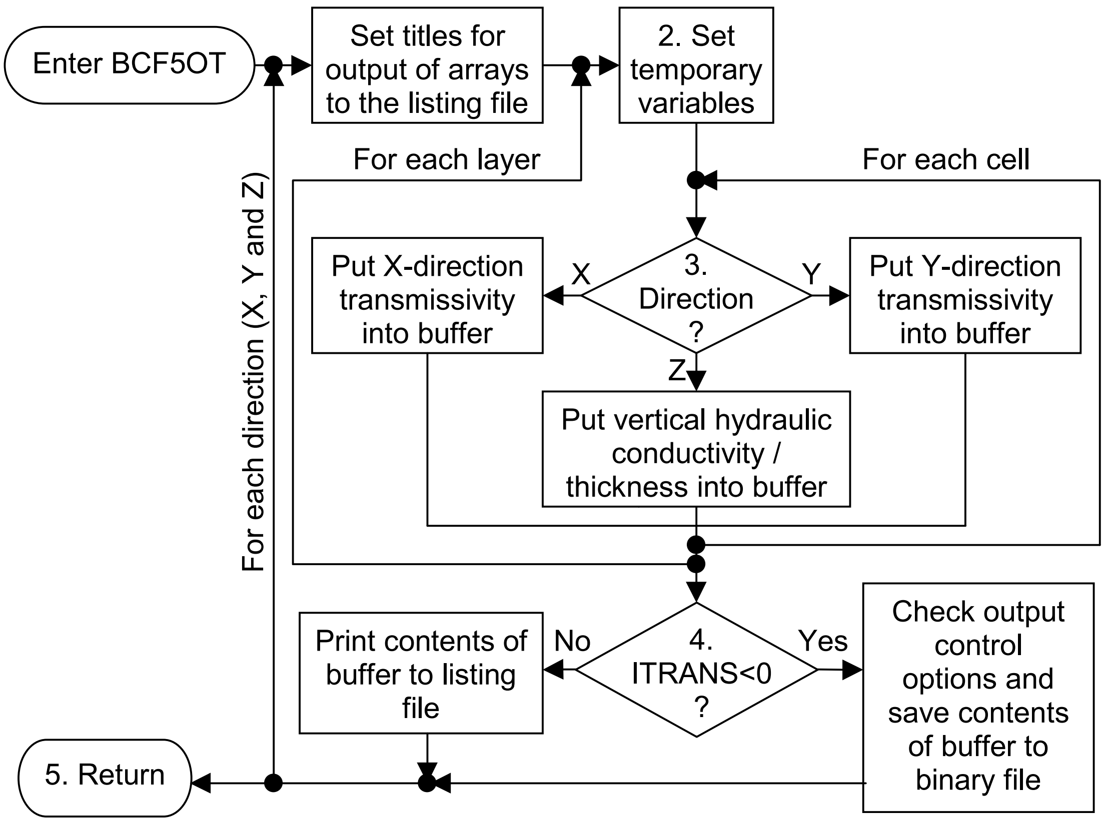
2.4.12 Narrative for module BCF5OT

This module prints arrays of X- and Y-direction interblock transmissivities, and vertical hydraulic conductivity/thickness to the listing file or a binary output file. Module BCF5OT is called by the main program after each time step if the flag ITRANS is not zero. If ITRANS is positive, the module writes transmissvity values to the listing file using utility modules ULAPRS or ULAPRW depending on the value of IHEDFM specified in the output control file (McDonald & Harbaugh, 1988, pp. 4-14 to 4-15). If ITRANS is negative, the module saves transmissivty values to a binary output file denoted by the negative value of ITRANS, using the utility modules ULASAV or ULASV2.

Module BCF5OT performs its functions in the following order:

0. Loop through X, Y and Z directions and set titles to write to listing file for each array
1. Loop through each layer
2. Set temporary variables
3. Loop through all cells putting X- or Y-direction transmissivity, or vertical hydraulic conductivity / thickness into the buffer array
4. Check sign of ITRANS and either print the contents of the buffer array to the listing file, or save them to a binary file (subject to output control specified for heads) and loop back for next layer
5. Return

Flow chart for module BCF5OT



List of variables for module BCF5OT

Variable	Range	Definition
BUFF	Global	DIMENSION (NCOL,NROW,NLAY), Buffer array with one element for each cell in the model. (real array)
CC	Package	DIMENSION (NCOL,NROW,NLAY), Conductance in the column direction. CC(J,I,K) contains conductance between nodes (J,I,K) and (J,I+1,K). (real array)
CHEDFM	Global	Character expression denoting format code for printed head output
CR	Package	DIMENSION (NCOL,NROW,NLAY), Conductance in the row direction. CR(J,I,K) contains conductance between nodes (J,I,K) and (J+1,I,K). (real array)
CV	Package	DIMENSION (NCOL,NROW,NLAY-1), Conductance in the vertical direction. CV(J,I,K) contains conductance between nodes (J,I,K) and (J,I,K+1). (real array)
DELC	Package	DIMENSION (NROW), Cell dimension in the column direction. DELC(I) contains width of row I. (real array)
DELR	Package	DIMENSION (NCOL), Cell dimension in the row direction. DELC(J) contains width of column J. (real array)
I	Module	Index (integer*4)
IBOUND	Global	DIMENSION (NCOL,NROW,NLAY), Status of each cell. (integer*4 array) < 0, constant-head cell = 0, inactive (no-flow) cell > 0, variable-head cell
IHEDFM	Global	Code for format in which head should be printed (same format used for transmissivity values) (integer*4)
IOFLG	Output Control module	DIMENSION (NLAY,4), Flags to control printing and recording of head and drawdown for each layer (NLAY,1) ≠ 0, heads will be printed (NLAY,2) ≠ 0, drawdown will be printed (NLAY,3) ≠ 0, heads and transmissivities will be recorded (NLAY,4) ≠ 0, drawdown will be recorded
IOUT	Global	Primary unit number for all printed output (integer*4)
ITRAN	Module	Direction index (integer*4)

Variable	Range	Definition
ITRANS	Package	Flag (integer*4) = 0, inter-nodal transmissivity values will not be written to the output file > 0, inter-nodal transmissivity values will be written to the output file every time step < 0, inter-nodal transmissivity values will be saved to a binary output file for every time step specified in the Output Control file (uses same flag as for head save)
ITRPY	Package	Flag (integer*4) = 0, a single value of the anisotropy ratio will be used for each layer = 2, X- and Y-direction transmissivities or hydraulic conductivities will be specified for each cell in each layer ≠ 0 or 2, a value of the anisotropy ratio will be used for each cell in each layer
IXSEC	Package	Flag (integer*4) = 0, Model is not a cross section ≠ 0, Model is a cross section
J	Module	Index (integer*4)
K	Module	Index for layers (integer*4)
KK	Module	Index (integer*4)
KL	Module	Index (integer*4)
KPER	Global	Stress period counter (integer*4)
KSTP	Global	Time step counter (integer*4). Reset at the start of each stress period
LBHDSV	Global	Flag denoting whether head save (and transmissivity) files will be labelled
NCOL	Global	Number of columns in the grid (integer*4)
NLAY	Global	Number of layers in the grid (integer*4)
NROW	Global	Number of rows in the grid (integer*4)
PERTIM	Global	Elapsed time during the current stress period
TEXT	Module	Title printed to the listing file to identify the array (character*16)
TOTIM	Global	Elapsed time in the simulation
TRPY	Package	DIMENSION (NCOL,NROW,NLAY), Anisotropy ratio: ratio of transmissivity in the column direction to transmissivity in the row direction (real array)
YX	Module	Temporary store for the anisotropy ratio (real)

2.4.13 Narrative for module SBCF5I

The module SBCF5I calculates horizontal branch conductances for a layer from inter-nodal transmissivity and cell dimensions. It is called by submodules SBCF5N and SBCF5H. The module is based on module SBCF5A and performs the same functions as those described for the modules in Section 2.4.7.

1) Process cells one at a time calculating branch conductances from that cell to the one on the right and the one in front.

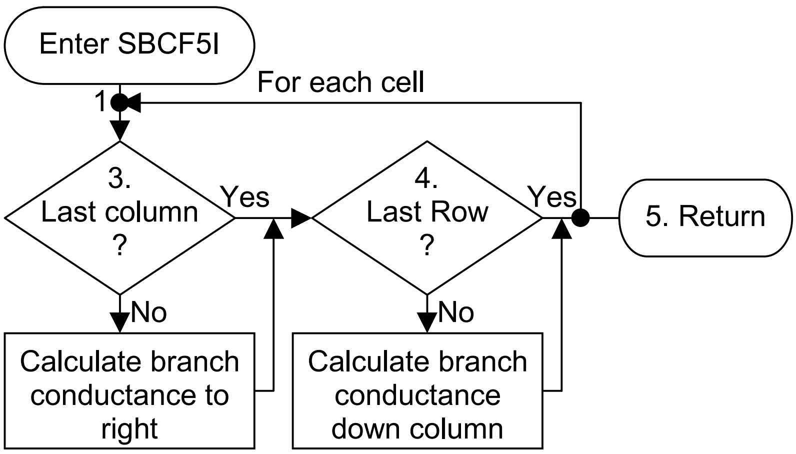
2) This step has been removed as there is no need to check for transmissivity values of zero.

3) If there is a cell to the right, calculate the branch conductance (CR) along the row.

4) If there is a cell in front, calculate the conductance (CC) along the column.

5) Return.

Flow chart for module SBCF5I



List of variables for module SBCF5I

Variable	Range	Definition
CC	Package	DIMENSION (NCOL,NROW,NLAY), Conductance in the column direction. CC(J,I,K) contains conductance between nodes (J,I,K) and (J,I+1,K). (real array)
CR	Package	DIMENSION (NCOL,NROW,NLAY), Conductance in the row direction. CR(J,I,K) contains conductance between nodes (J,I,K) and (J+1,I,K). (real array)
DELC	Package	DIMENSION (NROW), Cell dimension in the column direction. DELC(I) contains width of row I. (real array)
DELR	Package	DIMENSION (NCOL), Cell dimension in the row direction. DELC(J) contains width of column J. (real array)
I	Module	Index (integer*4)
ITRPY	Package	Flag (integer*4) = 0, a single value of the anisotropy ratio will be used for each layer = 2, X- and Y-direction transmissivities or hydraulic conductivities will be specified for each cell in each layer ≠ 0 or 2, a value of the anisotropy ratio will be used for each cell in each layer
J	Module	Index (integer*4)
K	Package	Index for layers (integer*4)
NCOL	Global	Number of columns in the grid (integer*4)
NLAY	Global	Number of layers in the grid (integer*4)
NROW	Global	Number of rows in the grid (integer*4)
T1	Module	Temporary store for conductance in the row direction (real)
T2	Module	Temporary store for conductance in the column direction (real)
TRPY	Package	DIMENSION (NCOL,NROW,NLAY), Anisotropy ratio: ratio of transmissivity in the column direction to transmissivity in the row direction (real array)
YX	Module	Temporary store for the anisotropy ratio (real)
ZERO	Module	Variable set to value of zero (real)

2.4.14 Narrative for module B12DRI

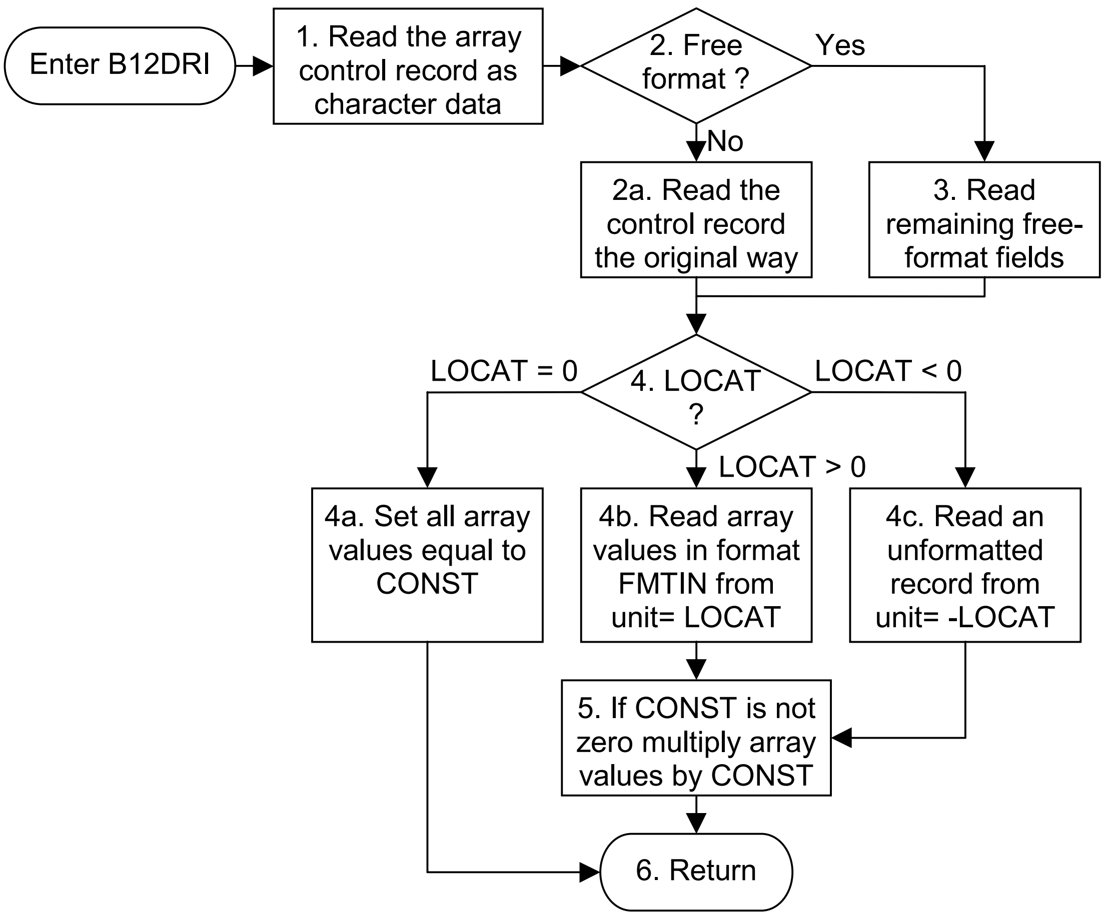
The module B12DRI reads values for a two-dimensional real array. It is based on module U2DREL in the utilities package, but has been modified to suppress printing of arrays to the listing file and to enable format codes to be stored for future use. The module B12DRI is called from module BCF5VK and calls the utility routine URWORD. It performs its functions in the following order:

1. Read the array control record as character data.
2. Look for alphabetic word that indicates that the record is free format. Set a flag specifying if free format or fixed format.

2a. If a recognised word was not found, read the control record the original way (LOCAT, CONST, FMTIN, IPRN).

3. For free format control record, read remaining fields.
4. Test LOCAT to see how to define array values.
   a. LOCAT=0; set all array values equal to CONST and return.
   b. LOCAT>0; read formatted records using format FMTIN.
   c. LOCAT<0; read unformatted record containing array values.
5. If CONST is not zero then multiply array values by CONST.
6. Return

Flow chart for module B12DRI



List of variables for module B12DRI

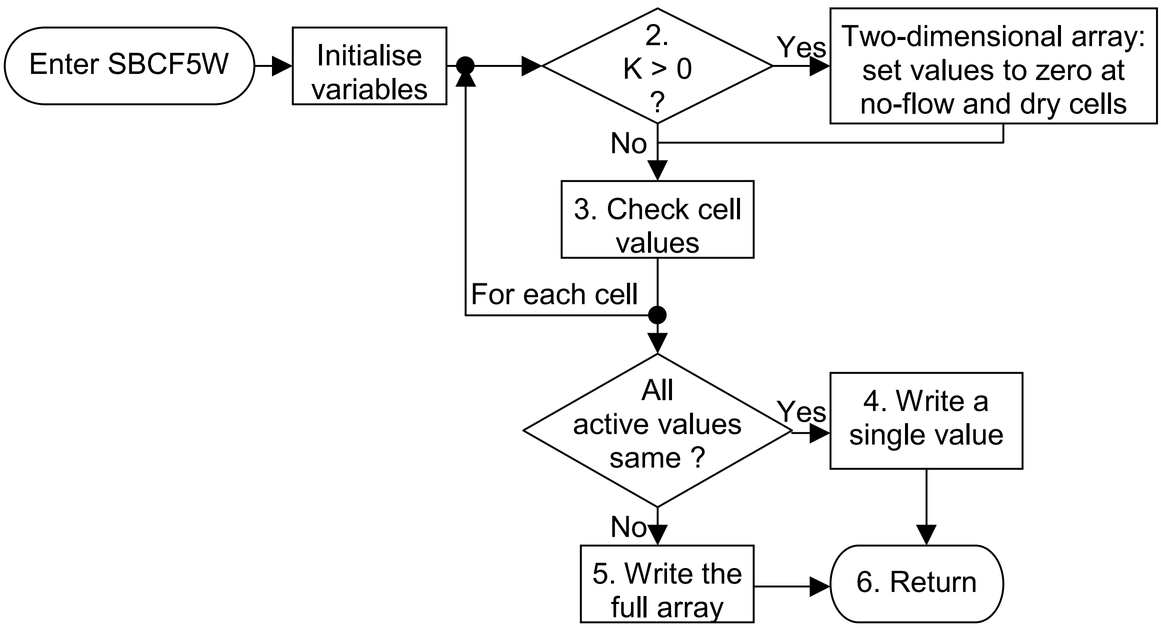
Variable	Range	Definition
A	Module	DIMENSION (JJ,II), Input array (real array)
CNTRL	Module	Character string used to read array control record (character*80)
CONST	Module	Constant to which all array values are set if LOCAT is equal to zero or by which all array values are multiplied if LOCAT is not equal to zero (real)
FMTIN	Module	Format under which array values will be read (character*20)
FNAME	Module	Filename of file containing array if OPEN/CLOSE option is used to read array (character*80)
I	Module	Index for rows (integer*4)
ICLOSE	Module	Flag used to specify whether an OPEN/CLOSE file has been opened (integer*4)
ICOL	Module	Index for position along text string CNTRL (integer*4)
IFREE	Module	Flag used to indicate if free format is being used (integer*4)
II	Module	Number of rows (integer*4)
IN	Module	Unit number from which the array control record will be read (integer*4)
IOUT	Global	Primary unit number for all printed output (integer*4)
IP	Module	Flag to indicate whether array values are written to the output file (0 suppresses output) (integer*4)
IPRN	Module	Code for format to be used when printing the arrays (not used in this routine) (integer*4)
ISTART	Module	Location on a line of the name file of the start of a word or number (integer*4)
ISTOP	Module	Location on a line of the name file of the end of a word or number (integer*4)
J	Module	Index for columns (integer*4)
JJ	Module	Number of columns (integer*4)
LOCAT	Module	Location of values to fill in the array (integer*4) < 0, read an unformatted record containing the array values = 0, set all the array values equal to constant (CONST) > 0, read the formatted records containing the array values
N	Module	Temporary integer value (integer*4)
NUNOPN	Module	Unit number for OPEN/CLOSE files (99) (integer*4)
R	Module	Temporary real value (real)

2.4.15 Narrative for module SBCF5W

The module SBCF5W writes values for a two-dimensional real array that can be read by the array readers in MODFLOW (including the array-control record with a specified format). It is called from module BCF5VK to write the arrays for the new BAS and BCF packages when the auto-conversion option is activated. It performs its functions in the following order:

0. Initialise variables.
1. Loop through all the cells in the layer.
2. If K > 0 then 2-d array, set values to zero at dry/no-flow cells (this avoids the likelihood of cells with unrealistic VKD parameters becoming active due to rewetting in subsequent simulations).
3. Check if all the active values in the array are the same.
4. If all active values are the same write a single value.
5. Otherwise write out the full array.
6. Return.

Flow chart for module SBCF5W



List of variables for module SBCF5W

Variable	Range	Definition
A	Module	DIMENSION (JJ,II), Output array (real array)
ALAST	Module	Temporary store for comparing values in an array (real)
ANAME	Module	Label for printout of output array (character*24)
FMT	Module	Format that will be used to write arrays (character*20)
HDRY	Package	Head value assigned to nodes that have become unsaturated and hence inactive (real)
HNEW	Global	DIMENSION (NCOL,NROW,NLAY), Most recent estimate of groundwater head in each cell (double precision array)
I	Module	Index for rows (integer*4)
IBCFOUT	Module	Unit number of the file that arrays will be written to (integer*4)
IBOUND	Global	DIMENSION (NCOL,NROW,NLAY), Status of each cell. (integer*4 array) < 0, constant-head cell = 0, inactive (no-flow) cell > 0, variable-head cell
ICONST	Module	Flag used to indicate whether all active values in the array are the same (integer*4)
IFIRST	Module	Flag used to determine whether the first active value in the array has been read (integer*4)
INBCF	Package	Unit number to write in the array control record (integer*4)
J	Module	Index for columns (integer*4)
K	Module	Layer number (integer*4)
NCOL	Package	Number of columns (integer*4)
NLAY	Package	Number of layers (integer*4)
NROW	Package	Number of rows (integer*4)

STEP 1

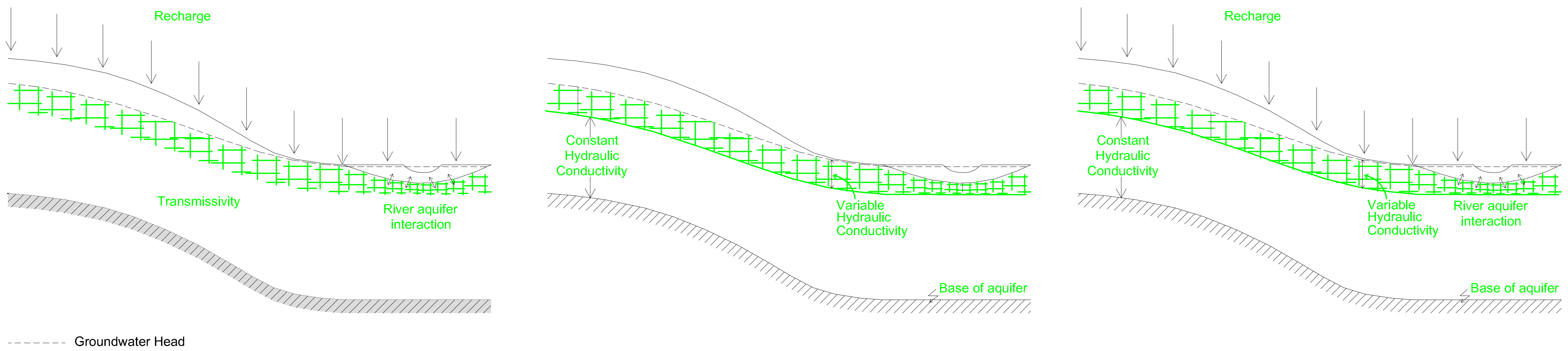
INITIAL CONDITIONS WITH SPECIFIED TRANSMISSIVITY

STEP 2

DEVELOPING PARAMETERS FOR VKD

STEP 3

INITIAL CONDITIONS WITH VKD



Specify: Transmissivity, recharge, river-aquifer interaction parameters
Determine: groundwater heads and flows

Calculate: hydraulic conductivity in lower (const K) part of aquifer and elevation of base of variable hydraulic conductivity zone
From: Transmissivity, groundwater heads, thickness of variable hydraulic conductivity zone, base of aquifer, rate of increase of hydraulic conductivity conductivity, and maximum hydraulic conductivity

Specify: base of aquifer, hydraulic conductivity, elevation of base of variable hydraulic conductivity zone, rate of increase of hydraulic conductivity, maximum hydraulic conductivity, recharge, river-aquifer interaction parameters
Determine: groundwater heads and flows. These should be equal to those calculated in Step 1

3 STREAM PACKAGE

3.1 Introduction

The Streamflow-Routing (STR) package (Prudic, 1989) is a modification of the River Package (McDonald & Harbaugh, 1988 & 1996) designed to route flow through one or more rivers, streams, canals or ditches (hereafter referred to as streams) in addition to computing the leakage between the streams and the aquifer system.

The processes represented by the stream package are given below prior to the description of the new features.

Streams that interact with the modelled aquifer are divided into reaches and segments. A segment is a stream or diversion in which stream flow from surface sources are added at the beginning of the segment or subtracted (in the case of a diversion) at the end of the segment. A reach is the part of a segment that corresponds to an individual cell in the finite-difference grid used to simulate groundwater flow in the aquifer. Segments are numbered sequentially from the farthest upstream segment to the last downstream segment, as are reaches within each segment. Both must also be read into the program in sequential order.

Streamflow is accounted for by specifying flow for the first reach in each segment that enters the modelled area and then computing streamflow to adjacent downstream reaches in each segment as equal to inflow in the upstream reach plus or minus leakage from or to the aquifer in the upstream reach. The accounting scheme used assumes that streamflow entering a stream reach is instantly available to downstream reaches. This assumption is generally reasonable because of the relatively slow rates of groundwater flow.

Streamflow into a segment that is formed from tributary streams is computed by adding the outflows from the last reach in each of the specified tributary segments. If a segment is a diversion, then the specified flow into the first reach of the segment is subtracted from flow in the main stream. However, if the specified flow of the diversion is greater than the flow out of the segment from which flow is to be diverted, then no flow is diverted from that segment.

Reaches in a segment are permitted to go dry whenever downward leakage to the aquifer exceeds stream inflow into the reach. The downstream reaches also go dry (streamflow into the downstream reaches is zero) and leakage into the aquifer is not permitted. Upward leakage (from the aquifer) is permitted even when there is no streamflow in a reach. This happens when the head in the aquifer exceeds the top of the streambed in the dry reach. Flow in the stream resumes and water may leak downwards back into the aquifer again in downstream reaches.

3.2 Conceptualisation and implementation of new features

The following discussion of the conceptualisation and implementation of the new features included in the stream package is divided into three sections: specified discharges or abstractions at any stream cell, tributary inflows at any stream cell, and a small correction to cell-by-cell stream flow output.

3.2.1 Specified discharges or abstractions at any stream cell

The original version of the stream package allows a specified inflow rate, i.e. a discharge,to be assigned only at the first reach (ie, first cell) of a segment, and does not allow surface water abstraction.. The input format includes a column for this flow rate for all stream reaches, but the value was ignored in all but the first reach in each segment. It looks as if the original version of the package was intended to allow inflows at any cell but was later changed.

The revised version of the stream package allows the user to specify additional discharges or abstractions at any reach of any stream segment. This means that contributions to surface water flow from discharges or runoff can be specified at any stream cell. The use of the –1 flag to denote tributary inflows is no longer used, allowing negative flows (abstractions) to be specified (connections to tributaries are now handled entirely by input block 5 – see below). If the specified abstraction rate is greater than the flow in the stream, then the abstraction is set equal to the inflow to that reach (drying the reach) and a message giving the reduced abstraction rate is written to the output file. To determine whether this option is used, a new flag (ISWABS) is read from the first line of the input file. This flag should be set to a non-zero value to activate the option.

3.2.2 Tributary inflows at any stream cell

The original version of the stream package allows tributary inflows to be assigned only at the first reach (ie, first cell) of a segment. The second modification of the stream package allows tributary inflows to be specified for any reach in a segment. This modification allows a major river to be specified with a single segment number, with tributary inflows from smaller streams at various points along its length. This change has been made to make pre- and post-processing easier. Setting the new flag (ISWABS) to a positive value activates the option. It may be noticed that this modification applies only to tributary inflows and not to diversions.

3.2.3 Small correction to cell-by-cell stream flow output

The final modification was made to the routine that writes the streamflows to the binary cell-by-cell output file. This change only affects simulations where more than one stream reach is defined in a single model cell. This would not normally be done, except perhaps at a confluence. The original version of the stream package summed the total streamflow in all the reaches in the cell and wrote this summed value to the output file, giving an erroneous value for the accreted streamflow at that point. A correction was made to the code so that only the stream flow from the furthest downstream reach was recorded in the binary output file.

3.3 Input instructions

3.3.1 Summary of new options

This new version of the MODFLOW modelling code allows two new options to be set (shown in **bold**) in the input instructions for the stream package. The new code should still be compatible with old MODFLOW input files (versions up to MODFLOW-96). The new options included are:

- **Allow surface water discharges and abstractions to be specified in any reach -** set **ISWABS <> 0** and specify positive (discharges) or negative (abstractions) values for the Flow into each reach
- **Allow tributary inflows to be specified in any reach -** set **ISWABS > 0** and include reach numbers in record 5a (below)

3.3.2 Input format

For each simulation:

Read in subroutine STR1AL

1.	Data:	MXSTRM	NSS	NTRIB	NDIV	ICALC	CONST	ISTCB1	ISTCB2	**ISWABS**
	Format:	I10	I10	I10	I10	I10	F10.0	I10	I10	**I10**

For each stress period:

Read in subroutine STR1RP

2.	Data:	ITMP	IRDFLG	IPTFLG
	Format:	I10	I10	I10

The following data set is read in sequential order of segment and reach.

3.	Data:	Layer	Row	Col	Seg	Reach	Flow	Stage	Cond	Sbot	Stop
	Format:	I5	I5	I5	I5	I5	F15.0	F10.0	F10.0	F10.0	F10.0

If stream stages for each reach are to be calculated (ICALC>0), then the following data set is read in sequential order of segment and reach.

4.	Data:	Width	Slope	Rough
	Format:	F10.0	F10.0	F10.0

If the maximum number of tributaries (NTRIB) that can join a segment is greater than zero, **and ISWABS$\leq$0,** then the following data set is read. One record for each segment is read in sequential order. A record is necessary even for segments that do not have tributaries. In this case a blank record or a record with all zeros is read.

5.	Data:	Itrib(1)	Itrib(2)	…	Itrib(NTRIB)
	Format:	I5	I5	…	I5

If the maximum number of tributaries (NTRIB) that can join a segment is greater than zero, **and ISWABS>0,** then the following data set is read. One record for each segment is read in sequential order. A record is necessary even for segments that do not have tributaries. In this case a blank record or a record with all zeros is read.

5a.	**Data:**	**Inrch(1)**	**Itrib(1)**	**Inrch(2)**	**Itrib(2)**	**…**	**Inrch(NTRIB)**	**Itrib(NTRIB)**
	Format:	**I5**	**I5**	**I5**	**I5**	**…**	**I5**	**I5**

If diversions are specified (NDIV>0), then the following data set is read. One record is read for each segment in sequential order. For segments that are not diversions, zeros or blanks are specified for each input item.

6.	Data:	Iupseg
	Format:	I10

3.3.3 Explanation of parameters used in input instructions

MXSTRM is the maximum number of stream reaches that can be active during the simulation.

NSS is the maximum number of segments that can be used during the simulation.

NTRIB is the maximum number of tributary segments that can join during a simulation. Ten is the maximum number allowed as currently specified in the program.

NDIV is a flag, which when positive, specifies that diversions from segments are to be simulated.

ICALC is a flag, which when positive, specifies that stream stages in reaches are to be calculated.

CONST is a constant value used in calculating stream stage in reaches. It is specified whenever ICALC is greater that zero. This constant is 1.486 for flow units of cubic feet per second and 1.0 for units of cubic meters per second. The constant must be multiplied by 86,400 when using time units of days in the simulation. (For an explanation of time units, see McDonald and Harbaugh, 1988, p. 4-10.)

ISTCB1 is a flag and a unit number.

If ISTCB1>0, it is the unit number to which leakage between each stream reach and the corresponding model cell will be saved on disk whenever the variable ICBCFL is specified. (See McDonald and Harbaugh, 1988, p. 4-15, for details about the Output Control Package used to specify ICBCFL.)

If ISTCB1=0, leakage between each reach and corresponding model cell will not be printed or filed on disk.

If ISTCB1<0, streamflows into and out of each reach and leakage between each reach and corresponding river cell will be printed whenever the variable ICBCFL is specified.

ISTCB2 is a flag and unit number for an option of storing accreted streamflow out of each reach instead of having the results printed.

If ISTCB2>0, it is the unit number where streamflow out of each reach will be saved on disk whenever the variable ICBCFL is specified.

If ISTCB2≤0, streamflows out of each reach will not be saved on disk.

ISWABS is a flag.

If ISWABS<0, surface water discharges and abstractions are allowed in ANY stream reach (abstraction rates are limited to the amount of flow available in the stream reach). Tributary inflows are only allowed in the first reach of a stream segment.

If ISWABS=0, surface water discharges and tributary inflows are only allowed in the FIRST reach of a stream segment. Surface water abstractions are not allowed. This option allows the stream package to be used in the original way.

If ISWABS>0, surface water discharges, abstractions and tributary inflows are allowed in ANY stream reach (abstraction rates are limited to the amount of flow available in the stream reach).

ITMP is a flag and a counter.

If ITMP<0, stream data from the last stress period will be reused.

If ITMP≥0, ITMP will be the number of reaches active during the current stress period.

IRDFLG is a flag, which when positive, suppresses printing of the input data set specified for a stress period. The input data set is printed for a stress period if the value is zero or blank.

IPTFLG is a flag, which when positive, suppresses printing of results for a stress period. The results are printed for a stress period if the value is zero or blank and whenever the variable ICBCFL is specified.

Layer is the layer number of the cell containing the stream reach.

Row is the row number of the cell containing the stream reach.

Col is the column number of the cell containing the stream reach.

Seg is a number assigned to a group of reaches. Segments must be numbered in downstream order and are read into the program in sequential order.

Reach is a sequential number in a segment that begins with one for the farthest upstream reach and continues in downstream order to the last reach in the segment. Reaches must be read in sequentially as the order reaches are read into the program determines the order of connection of inflows and outflows.

Flow is the surface water abstraction/discharge or diversion flow $[L^3T^{-1}]$ entering the segment.

If ISWABS=0, Flow is specified only for the first reach in each segment. The value is either a zero or a blank when the reach number (Reach) is not 1. When inflow into a segment is the sum of outflow from a specified number of tributary segments, the segment inflow values are specified as a –1.

If ISWABS<>0, Flow can be specified for any reach in any segment. Surface water abstractions can be simulated by entering a negative value (if the specified abstraction is greater than the available stream flow, all the flow is removed from the stream and a message is written to the output file detailing the reduction to the abstraction). Tributary inflows are handled entirely by the flow routing section in input block 5 (Itrib).

(Note: if the specified inflow to a diversion is greater than the flow in the reach from which flow is to be diverted, then no flow is diverted from the stream.)

Stage is the stream stage, in units of length.

Cond is the streambed hydraulic conductance, in units of length squared per time.

Sbot is the elevation of the bottom of the streambed, in units of length.

Stop is the elevation of the top of the streambed, in units of length.

Width is the width of the stream channel, in units of length. It is used only when the stream stage in each reach is to be calculated.

Slope is the slope of the stream channel in each reach, in units of length per length. It is used only when stream stage in each reach is calculated.

Rough is Mannings roughness coefficient for each stream reach. It is used only when stream stage in each reach is calculated.

Itrib(1) for a segment that has tributary segments, Itrib(1) is the number of the first tributary segment. For a segment with no tributaries, Itrib(1) must be specified as zero.

Itrib(2) for a segment that has tributary segments, Itrib(2) is the number of the second tributary segment. For a segment with no or only one tributary segment, Itrib(2) must be specified as zero.

Inrch(1) for a segment that has tributary segments, if tributaries are allowed in any stream reach (ISWABS>0), Inrch(1) is the number of the reach that the first tributary flows into.

Inrch(2) for a segment that has tributary segments, if tributaries are allowed in any stream reach (ISWABS>0), Inrch(2) is the number of the reach that the second tributary flows into.

Iupseg for a diversion segment, Iupseg is the number of the upstream segment from which water is diverted. For a segment that is not a diversion, Iupseg must be specified as zero.

3.3.4 Example stream input file

```
    150      12       2       0       0   86400      50      51        1
    150       0       0  Inflow   Stage   Cond.  Bottom     Top
1   54   57   1   1          10   85.28  2.50E+04   84.28   85.28
1   55   57   1   2          10  85.217  2.50E+04  84.217  85.217
1   55   56   1   3          10   84.43  2.50E+04   83.43   84.43
1   55   55   1   4          10  83.303  2.50E+04  82.303  83.303
1   56   55   1   5          10  82.579  2.50E+04  81.579  82.579
1   56   54   1   6          10  81.629  [illegible]
1   [illegible]              10  [illegible].209  [illegible]
1   [illegible]              10  [illegible].364  [illegible]
1   [illegible]              10  [illegible].683  [illegible]
1   59   52   1  10          10  76.418  [illegible]
1   60   52   1  11          10  74.752  [illegible]
1   61   52   1  12          10  73.204  [illegible]
1   62   51   1  13          10  71.734  [illegible]
1   63   51   1  14          10  69.944  [illegible]
1   64   51   1  15          10  68.695  [illegible]
1   65   51   1  16          10  67.446  [illegible]
1   66   51   1  17          10  66.411  [illegible]
1   66   52   1  18          10  63.489  2.50E+04  62.489  63.489
1   67   52   1  19          10  64.192  2.50E+04  63.192  64.192
1   68   52   1  20          10  61.216  2.50E+04  60.216  61.216
1   69   52   1  21          10  58.591  2.50E+04  55.182  58.591
1   70   52   1  22          10  56.655  2.50E+04   51.31  56.655
1   67   54   [illegible]                       [illegible]0E+04  68.455  69.455
1   68   54   [illegible]                       [illegible]0E+04  65.457  66.457
1   70   61   [illegible]                       [illegible]0E+04   77.85   78.85
1   70   60   [illegible]                       [illegible]0E+04  76.425  77.425
1   70   59   [illegible]                       [illegible]0E+04  72.874  73.874
1   69   58   [illegible]                       [illegible]0E+04   69.82   70.82
1   68   57   [illegible]                       [illegible]0E+04  68.802  69.802
1   68   56   [illegible]                       [illegible]0E+04  66.901  67.901
1   69   57   [illegible]                       [illegible]0E+04  66.837  67.837
1   69   56   [illegible]                       [illegible]0E+04  65.173  66.173
1   69   55   [illegible]                       [illegible]0E+01      63      64
1   69   54   [illegible]                       [illegible]0E+04   61.23   62.23
1   80   56   [illegible]                       [illegible]0E+04  70.348  71.348
1   79   56   8   2          10  70.757  2.50E+04  69.757  70.757
1   78   56   8   3          10  69.716  2.50E+04  68.716  69.716
1   78   55   8   4          10  68.756  2.50E+04  67.756  68.756
1   77   55   8   5          10  67.646  2.50E+04  66.646  67.646
[illegible]                                                   [illegible]668
[illegible]                                                   [illegible]278
[illegible]                                                   [illegible]272
[illegible]                                                   [illegible]277
1   72   53   8  10          10  59.745  2.50E+04  58.745  59.745
1   71   53   8  11          10  58.342  2.50E+04  57.342  58.342
1   70   53   8  12          10  57.321  2.50E+04  56.321  57.321
1   71   52   8  13          10  55.535  2.50E+04  54.535  55.535
1   71   51   8  14          10   55.01  2.50E+04   54.01   55.01
1   70   50   8  15          10  53.936  2.50E+04  52.936  53.936
1   70   49   8  16          10  52.932  2.50E+04  51.932  52.932
1   70   48   8  17          10  51.936  2.50E+04  50.936  51.936
1   69   48   8  18          10  51.191  2.50E+04  50.191  51.191
1   69   47   8  19          10  50.218  2.50E+04  49.218  50.218
1   69   46   8  20          10  49.244  2.50E+04  48.244  49.244
1   69   45   8  21          10  48.307  2.50E+04  47.307  48.307
1   69   44   8  22          10  47.542  2.50E+04  46.542  47.542
1   70   44   8  23          10  47.029  2.50E+04  46.029  47.029
1   70   43   8  24          10  46.361  2.50E+04  45.361  46.361
1   69   42   8  25          10  45.454  2.50E+04  44.454  45.454
1   69   41   8  26          10  45.164  2.50E+04  44.164  45.164
1   70   40   8  27          10   43.82  2.50E+04   42.82   43.82
1   69   39   8  28          10  42.498  2.50E+04  41.498  42.498
1   70   38   8  29          10  41.332  2.50E+04  40.332  41.332
1   71   37   8  30          10  40.361  2.50E+04  39.361  40.361
1   54   45   9   1          10  75.623  2.50E+04  74.623  75.623
1   54   44   9   2          10  74.618  2.50E+04  73.618  74.618
1   54   43   9   3          10  73.693  2.50E+04  72.693  73.693
1   55   43   9   4          10  72.586  2.50E+04  71.586  72.586
1   55   42   9   5          10  71.371  2.50E+04  70.371  71.371
1   55   41   9   6          10  74.932  2.50E+04  73.932  74.932
1   56   41   9   7          10  73.817  2.50E+04  72.817  73.817
1   56   40   9   8          10  72.491  2.50E+04  71.491  72.491
1   56   39   9   9          10  71.417  2.50E+04  70.417  71.417
1   56   38   9  10          10  69.297  2.50E+04  68.297  69.297
1   56   37   9  11          10   67.58  2.50E+04   66.58   67.58
1   56   36   9  12          10  66.361  2.50E+04  65.361  66.361
1   55   36   9  13          10  65.224  2.50E+04  64.224  65.224
1   55   35   9  14          10  64.419  2.50E+04  63.419  64.419
```

Item: 2
Variables: ITMP, IRDFLG & IPTFLG

Item: 1
Variable: **ISWABS**
Value: **1**
Option: **Surface water discharges, abstraction & tributary inflows allowed in any stream reach.**

Item: 3
Variable: Flow
Value: **10**
Option: **Flow of 10 m^3/d added to each stream reach (can be used to represent inflow from surface water runoff).**

Item: 3
Variables: Layer, Row, Col, Seg, Reach, Flow, Stage, Cond, Sbot, Stop

```
1   55   34   9   15          10   62.809  2.50E+04   61.809   62.809
1   55   33   9   16          10   61.502  2.50E+04   60.502   61.502
1   55   32   9   17          10   59.619  2.50E+04   58.619   59.619
1   54   32   9   18          10   58.363  2.50E+04   57.363   58.363
1   53   32   9   19          10   56.553  2.50E+04   55.553   56.553
1   52   31   9   20          10   55.778  2.50E+04   54.778   55.778
1   52   30   9   21          10   55.522  2.50E+04   54.522   55.522
1   52   29   9   22          10   55.011  2.50E+04   54.011   55.011
1   52   28   9   23          10   54.467  2.50E+04   53.467   54.467
1   51   28   9   24          10   53.964  2.50E+04   52.964   53.964
1   51   27   9   25          10   53.515  2.50E+04   52.515   53.515
1   51   26   9   26          10   53.004  2.50E+04   52.004   53.004
1   51   25   9   27          10   52.524  2.50E+04   51.524   52.524
1   50   25   9   28          10   52.066  2.50E+04   51.066   52.066
1   50   24   9   29          10   51.599  2.50E+04   50.599   51.599
1   50   23   9   30          10   51.172  2.50E+04   50.172   51.172
1   51   22   9   31          10       51  5.00E+01       50       51
1   35   45  12    1          10   88.321  2.50E+04   87.321   88.321
1   34   44  12    2          10   86.553  2.50E+04   85.553   86.553
1   34   43  12    3          10   85.111  2.50E+04   84.111   85.111
1   34   42  12    4          10    83.88  2.50E+04    82.88    83.88
1   35   42  12    5          10   82.542  2.50E+04   81.542   82.542
1   35   41  12    6          10   81.443  2.50E+04   80.443   81.443
1   35   40  12    7          10   80.296  2.50E+04   79.296   80.296
1   35   39  12    8          10   79.594  2.50E+04   78.594   79.594
1   36   39  12    9          10   78.481  2.50E+04   77.481   78.481
1   36   38  12   10          10   77.524  2.50E+04   76.524   77.524
1   37   38  12   11          10    76.53  2.50E+04    75.53    76.53
1   37   37  12   12          10   75.219  2.50E+04   74.219   75.219
1   37   36  12   13          10   73.817  2.50E+04   72.817   73.817
1   37   35  12   14          10   72.559  2.50E+04   71.559   72.559
1   38   35  12   15          10    71.19  2.50E+04    70.19    71.19
1   38   34  12   16          10    69.88  2.50E+04    68.88    69.88
1   39   34  12   17          10   69.082  2.50E+04   68.082   69.082
1   39   33  12   18          10   67.985  2.50E+04   66.985   67.985
1   38   32  12   19          10   65.571  2.50E+04   64.571   65.571
1   38   31  12   20          10   64.485  2.50E+04   63.485   64.485
1   39   31  12   21          10   63.257  2.50E+04   62.257   63.257
1   40   31  12   22          10   61.971  2.50E+04   60.971   61.971
1   40   30  12   23          10   61.204  2.50E+04   60.204   61.204
1   41   30  12   24          10   60.668  2.50E+04   59.668   60.668
1   41   29  12   25          10   60.329  2.50E+04   59.329   60.329
1   41   28  12   26          10   59.594  2.50E+04   58.594   59.594
1   42   28  12   27          10     58.9  2.50E+04     57.9     58.9
1   43   28  12   28          10    58.22  2.50E+04    57.22    58.22
1   44   28  12   29          10   57.561  2.50E+04   56.561   57.561
1   45   28  12   30          10   56.942  2.50E+04   55.942   56.942
1   45   27  12   31          10   56.394  2.50E+04   55.394   56.394
1   45   26  12   32          10   55.669  2.50E+04   54.669   55.669
1                                  54.912  2.50E+04   53.912   54.912
1                                  54.167  2.50E+04   53.167   54.167
1                                  53.527  2.50E+04   52.527   53.527
1                                  52.971  2.50E+04   51.971   52.971
1                                  52.336  2.50E+04   51.336   52.336
1                                  51.741  2.50E+04   50.741   51.741
1                                  51.199  2.50E+04   50.199   51.199
1                                  50.592  2.50E+04   49.592   50.592
1                                  49.395  2.50E+04   48.395   49.395
1                                   48.55  2.50E+04    47.55    48.55
1                                   47.66  2.50E+04    46.66    47.66
1                                  46.832  2.50E+04   45.832   46.832
1                                  46.203  2.50E+04   45.203   46.203
1   52   18  12   46          10   45.552  2.50E+04   44.552   45.552
1   52   17  12   47          10   44.972  2.50E+04   43.972   44.972
1   53   17  12   48          10   44.063  2.50E+04   43.063   44.063
1   53   16  12   49          10    43.12  2.50E+04    42.12    43.12
1   54   16  12   50          10   42.047  2.50E+04   41.047   42.047
1   55   16  12   51          10   41.245  2.50E+04   40.245   41.245
1   55   15  12   52      -50000   40.733  2.50E+04   39.733   40.733
1   56   15  12   53          10   40.494  2.50E+04   39.494   40.494
1   57   15  12   54          10   40.022  2.50E+04   39.022   40.022
1   57   14  12   55          10   39.558  2.50E+04   38.558   39.558
     0        0    :Segments that flow into Segment 1
     0        0                                    t 2
     0        0                                    t 3
10   3        0                                    t 4
     0        0                                    t 5
     0        0                                    t 6
     0        0                                    t 7
13   1   12   4                                    t 8
     0        0                                    t 9
     0        0                                    t 10
     0        0                                    t 11
43   9        0    :Segments that flow into Segment 12
    -1        1        0        Stress period 2
    -1        1        0        Stress period 3
    -1        1        0        Stress period 4
```

Item: 3
Variable: Flow
Value: **-50,000**
Option: **Flow of 50,000 m³/d taken from stream reach (can be used to represent surface water abstraction).**

Item: **5a**
Variables: **Inrch(1)**, Itrib(1),…
Values: **10**, 3,…
Option: Stream segment 3 is a tributary flowing into **reach 10 of** segment 4.

Item: **5a**

Item: 2
Variables: ITMP, IRDFLG & IPTFLG

(Data repeated for stress periods 2-4)

3.4 Module documentation for stream package

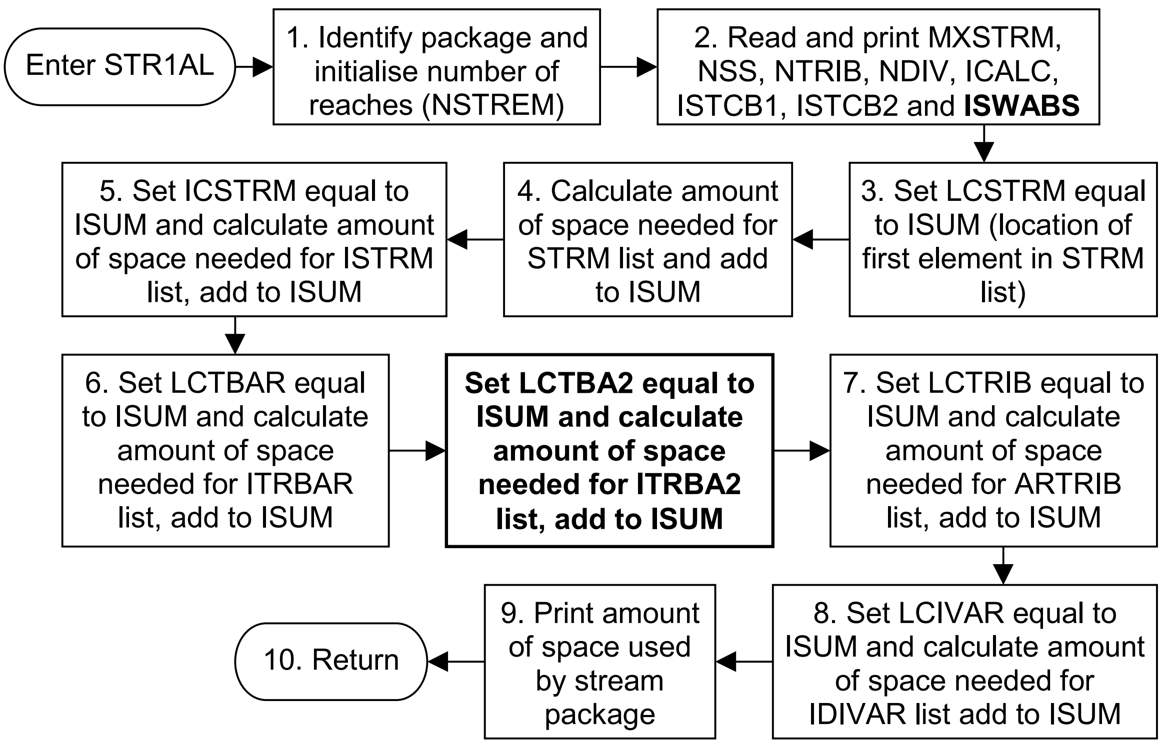
Modifications have been made to all four of the original modules of the stream routing package (STR-VKD1). No extra modules have been added. All of the modules are called by the MAIN program. The modules are:

STR1AL Allocates space for arrays used in the package.
STR1RP Reads all data needed by the package and initialises reach inflow, outflow, and leakage arrays.
STR1FM Calculates leakage to and from stream reaches and adds appropriate terms to finite-difference equations used to calculate heads in aquifer and simulate groundwater flow.
STR1BD Calculates rates and accumulated volumes of stream leakage into and out of aquifer. Results are either saved on files or printed.

3.4.1 Changes to module STR1AL

- Different title printed to listing file
- Read in new input flag (ISWABS - activates new options)
- Print message to output file if new options are activated
- Allocate storage space for new array (ITRBA2 – holds extra tributary routing info)

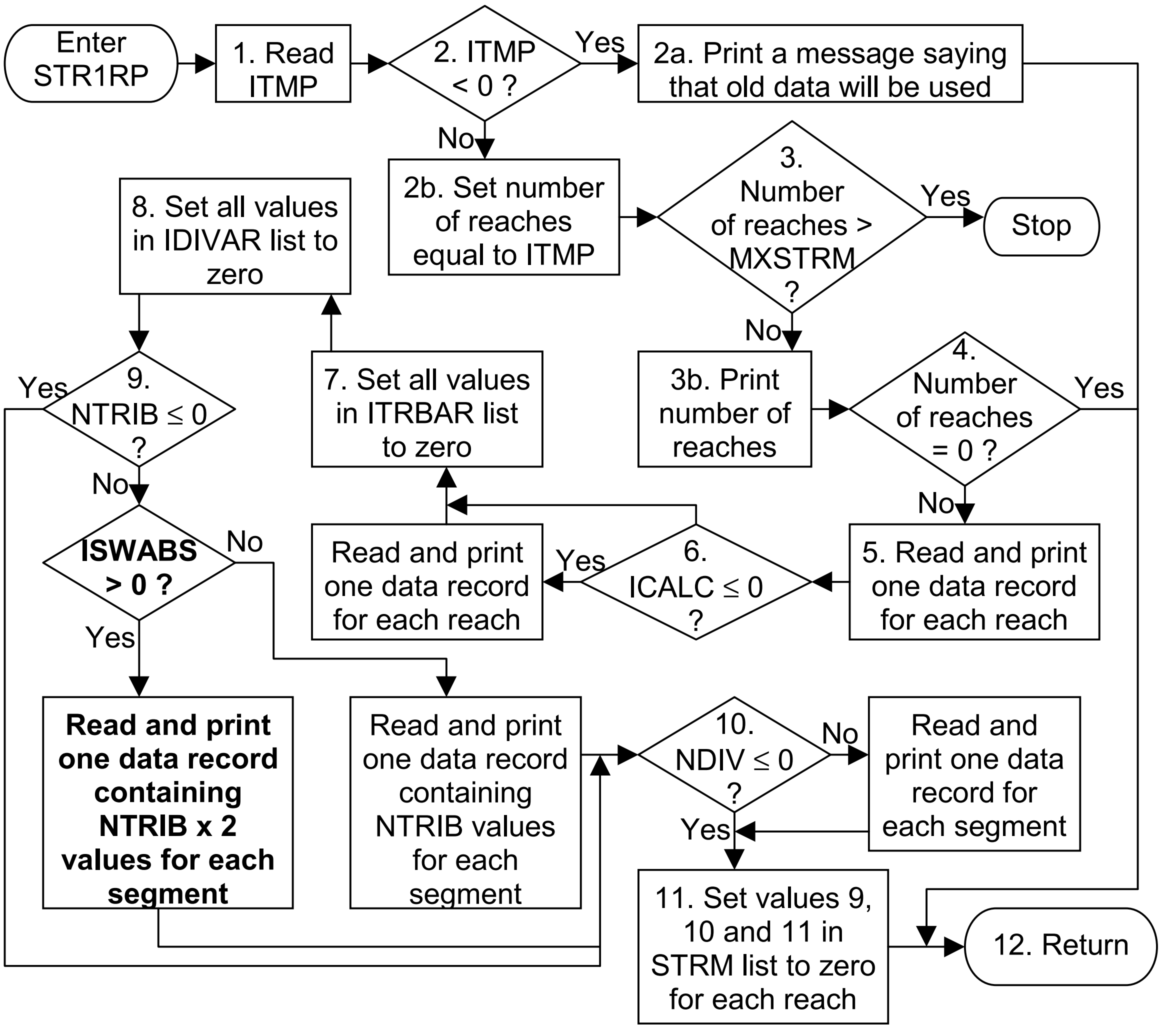
Flow Chart for module STR1AL



3.4.2 Changes to module STR1RP

- Dimension new array ITRBA2
- Change input format if ISWABS is greater than zero
- Change format of input echoed to listing file

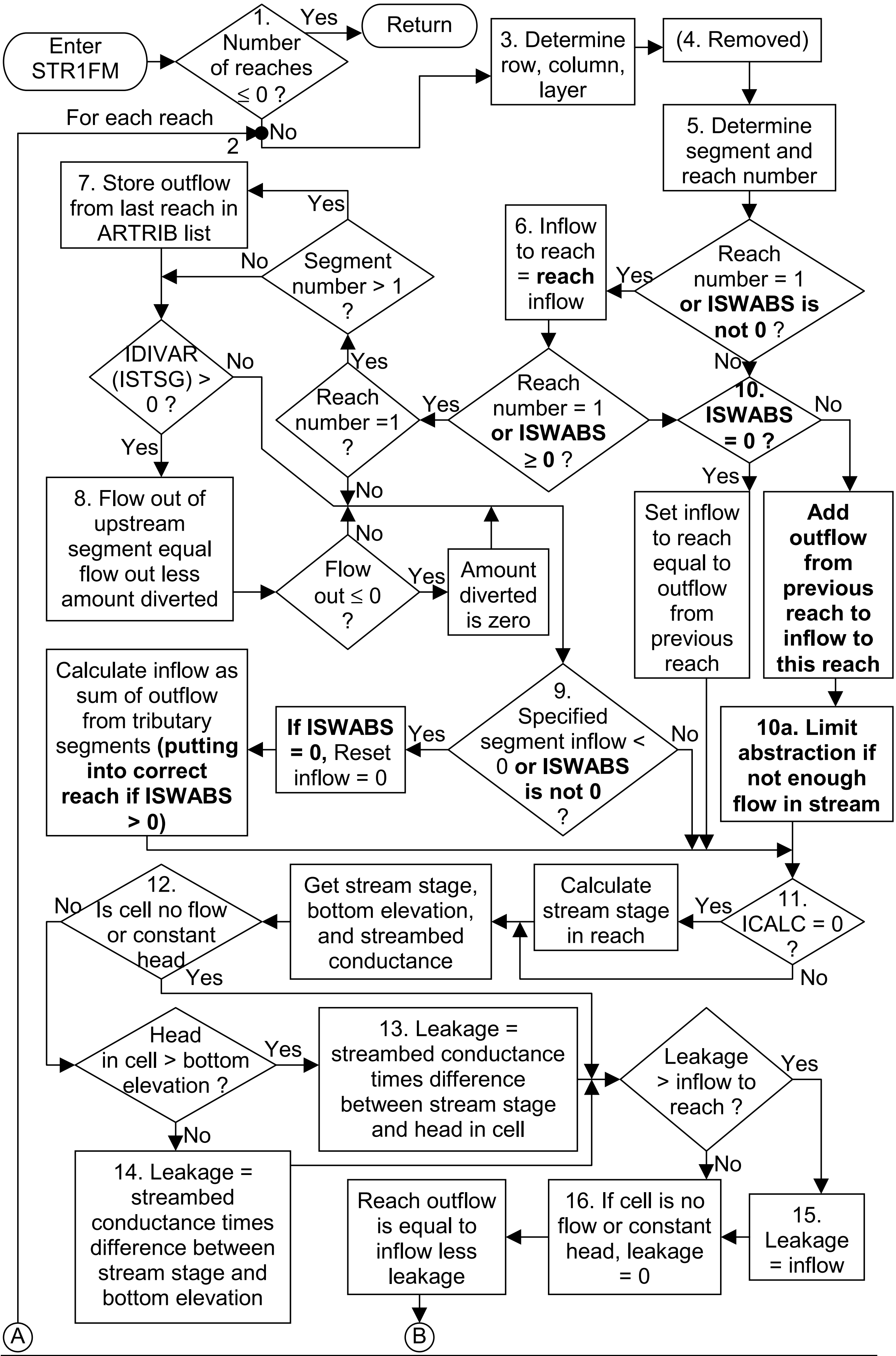
Flow chart for module STR1RP

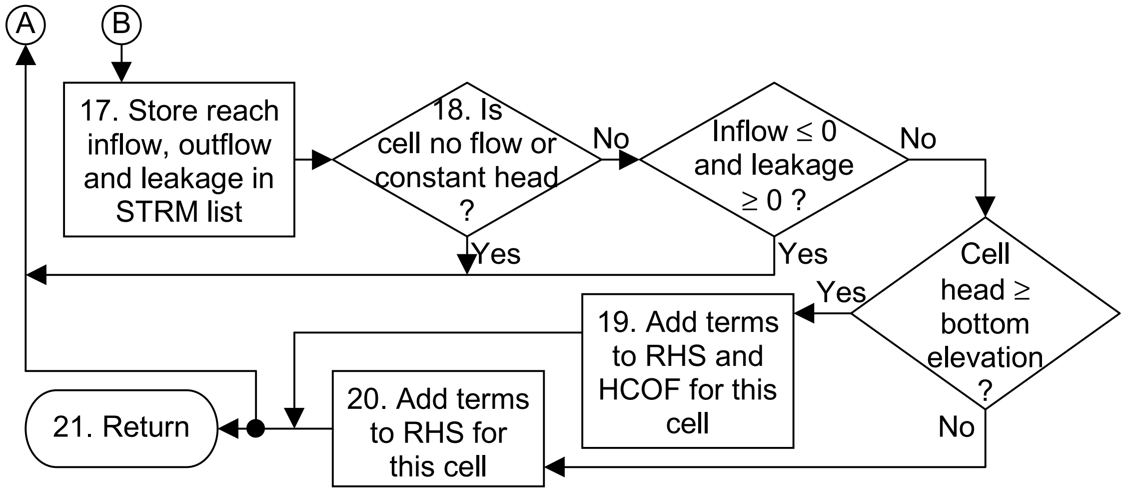


3.4.3 Changes made to STR1FM

- Dimension new array ITRBA2.
- Allow surface water abstraction or discharge if ISWABS is not zero.
- Allow tributary inflows at any reach in a segment if ISWABS is greater than zero.
- Limit surface water abstraction if there is not enough flow available in the stream.

Flow chart for module STR1FM

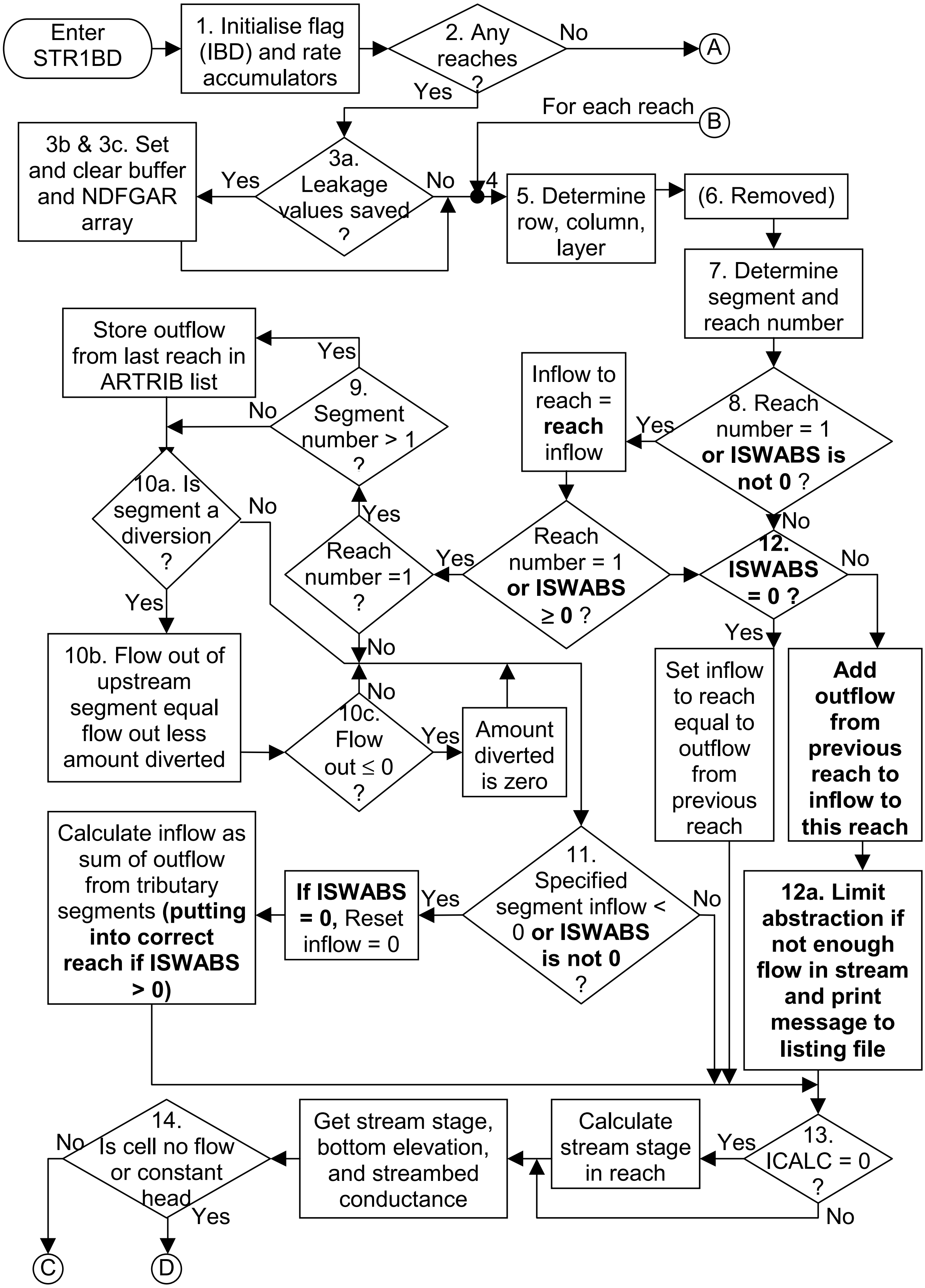


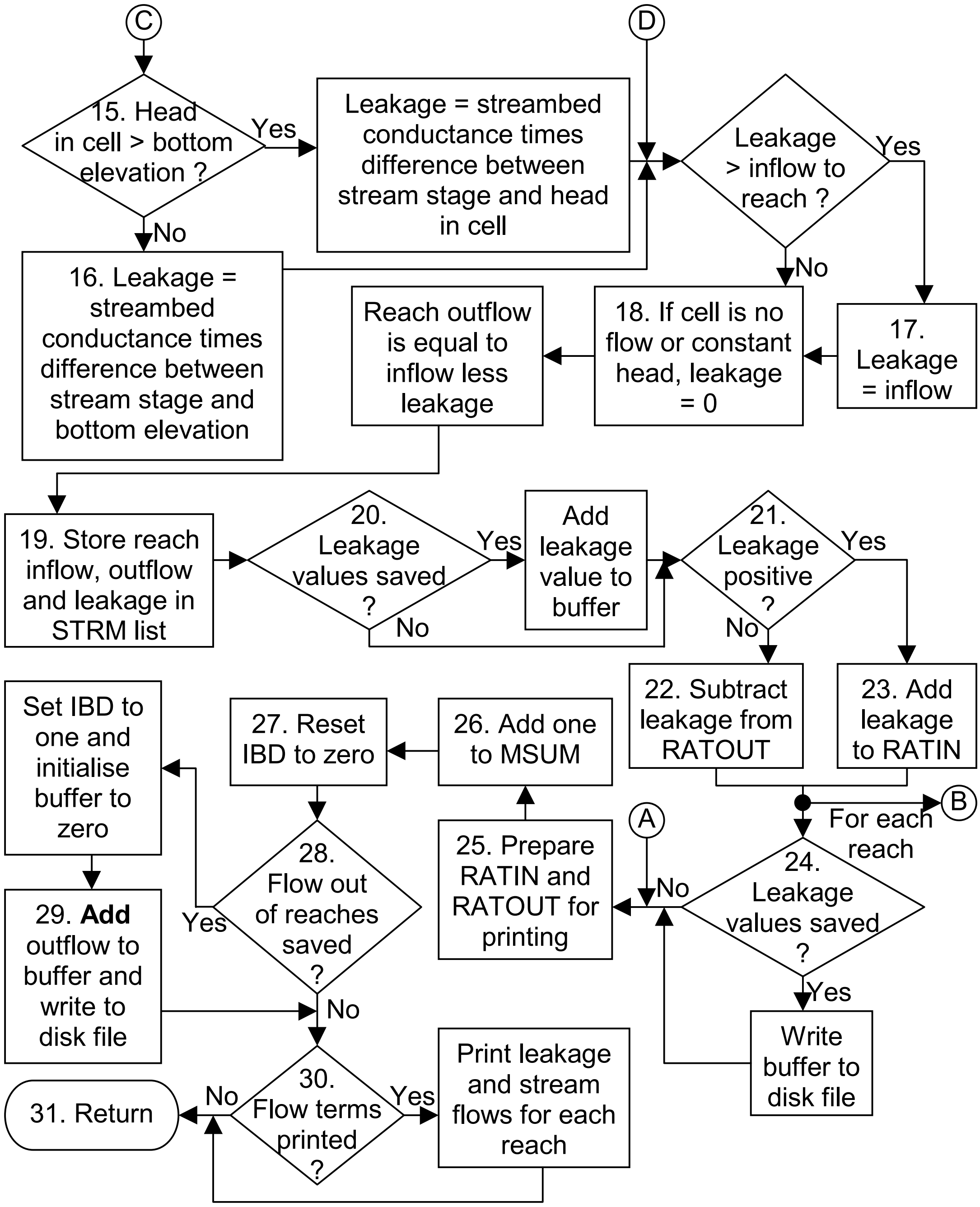


3.4.4 Changes made to STR1BD

- Dimension new array ITRBA2
- Allow surface water abstraction or discharge if ISWABS is not zero
- Allow tributary inflows at any reach in a segment if ISWABS is greater than zero
- Limit surface water abstraction if there is not enough flow available in the stream and print a message to the listing file

Flow chart for module STR1BD




C
D
15. Head in cell > bottom elevation ?
Yes
No
Leakage = streambed conductance times difference between stream stage and head in cell
Leakage > inflow to reach ?
Yes
No
16. Leakage = streambed conductance times difference between stream stage and bottom elevation
Reach outflow is equal to inflow less leakage
18. If cell is no flow or constant head, leakage = 0
17. Leakage = inflow
19. Store reach inflow, outflow and leakage in STRM list
20. Leakage values saved ?
Yes
No
Add leakage value to buffer
21. Leakage positive ?
Yes
No
22. Subtract leakage from RATOUT
23. Add leakage to RATIN
Set IBD to one and initialise buffer to zero
27. Reset IBD to zero
26. Add one to MSUM
B
For each reach
A
25. Prepare RATIN and RATOUT for printing
24. Leakage values saved ?
No
Yes
28. Flow out of reaches saved ?
Yes
No
29. Add outflow to buffer and write to disk file
Write buffer to disk file
31. Return
No
30. Flow terms printed ?
Yes
Print leakage and stream flows for each reach

4 PRECONDITIONED CONJUGATE-GRADIENT SOLVER PACKAGE

4.1 Introduction

The Preconditioned Conjugate-Gradient (PCG) Solver Package (Hill MC, 1990) is one of the more powerful solvers available for MODFLOW. The solver allows a solution to be found that satisfies criteria of both minimum head change and flow residual.

The solver converges on a solution through a series of inner and outer iterations. The coefficients of the finite difference equations (some of which are head-dependant) are recalculated at the start of each outer iteration, based on the current estimate of the head distribution. Within each outer iteration a number of inner iterations are carried out during which the coefficients remain unchanged.

4.2 New features included

The changes that have been made to this version of the solver package do not affect the way that the finite difference equations are solved.

4.2.1 Progress monitor

The first modification prints convergence information to the screen whilst the model is running so that the progress of the simulation can be monitored. An input flag does not control this modification; the progress monitor will always be displayed when the PCG solver is used. The information written to the screen includes the stress period, the time step, the iteration and the current degree of convergence in terms of the head difference and the flow residual.

4.2.2 Force convergence if criteria met for a specified number of iterations

The second modification has been made to make this version of the code compatible with the Windows version of MODFLOW supplied with Groundwater Vistas (the Agency's preferred MODFLOW user interface (Environmental Simulations Inc, 2001)). ***It should be stressed that the use of this option can adversely affect the accuracy of the solutions obtained and it is not recommended that the option be used without great caution and independent checks of the water balance.*** This option has been included so that the modified code is compatible with models created by Groundwater Vistas.

The original version of the package only reaches convergence if the criteria are met during the first inner iteration of an outer iteration. This means that the convergence criteria can be met at the end of an outer iteration but if the criteria are not met immediately once the coefficients are updated, the iterations continue. The new option allows the user to specify the maximum number of consecutive outer iterations (NOUTC) during which the criteria are met before convergence is 'forced'. This can sometimes lead to large water balance errors.

4.2.3 Debugging option

The final modification can be used to identify problem areas in a simulation that fails to converge. To activate the option a unit number is entered at the top of the input file (IPCGDEBUG) and file name specified in the NAME file as DATA(BINARY). The simulation will then produce a heads-type binary output file containing the heads calculated in each iteration. This file is created for every time step of the simulation, and is cleared each time the convergence criteria are met, so only the heads from the unconverged time step remain in the file. This option will slow simulations down and quite large files can be created. The contents of the file produced are as follows:

Position in file (bytes)	Data in heads file	Data in debug file	Data type
0	Time step	Inner iteration	Integer
4	Stress period	Outer iteration	Integer
8	Stress period time	Largest head difference	Real
12	Total time	Largest residual	Real
16	"HEAD"	"Unconverged Head"	Text
32	Number of columns	Number of columns	Integer
36	Number of rows	Number of rows	Integer
40	Layer number	Layer number	Integer
44	Head values	Unconverged head values	Real array

4.3 Input instructions

4.3.1 Summary of new options

This new version of the MODFLOW modelling code allows three new options to be set (shown in **bold**) in the input instructions for the PCG package. The new code should still be compatible with old MODFLOW input files (versions up to MODFLOW-96). The new options included are:

- **Progress monitor -** automatically runs when using PCG solver
- **Force convergence if criteria met for specified number of outer iterations -** set **NOUTC > 0**
- **Debugging option -** set IPCGDEBUG > 0 and specify a file name in the NAME file

4.3.2 Input format

For each simulation:

Read in subroutine PCG2AL

1.	Data:	MXITER	ITER1	NPCOND	**NOUTC**	**IPCGDEBUG**
	Format:	I10	I10	I10	**I10**	**I10**

Read in subroutine PCG2RP

2.	Data:	HCLOSE	RCLOSE	RELAX	NBPOL	IPRPCG	MUTPCG	DAMP
	Format:	F10.0	F10.0	F10.0	I10	I10	I10	F10.0

4.3.3 Explanation of parameters used in input instructions

MXITER is the maximum number of outer iterations – that is, calls to the solution routine. For a linear problem MXITER should be 1, unless more than 50 inner iterations are required, when MXITER could be as large as 10. A larger number (generally less than 100) is required for a nonlinear problem.

ITER1 is the maximum number of inner iterations. For nonlinear problems, ITER1 usually ranges from 3 to 10; a value of 30 will be sufficient for most linear problems.

NPCOND is the flag used to select the matrix preconditioning method. The following options are available.

NPCOND	Preconditioning method
1	Modified Incomplete Cholesky (for use on scalar computers (PCs))
2	Polynomial (for use on vector computers or to conserve computer storage)

NOUTC is the maximum number of consecutive outer iterations during which the criteria are met before convergence is 'forced' (can lead to invalid results). If NOUTC is 0 then convergence will not be forced.

IPCGDEBUG is a flag and a unit number.

If IPCGDEBUG > 0, it is the unit number on which unconverged head values will be recorded. A record is written every iteration, and is cleared each time a time step converges. This option slows the simulation down as output is written every iteration. The size of the file will be large if a large number of iterations are allowed.

If IPCGDEBUG = 0, unconverged head values will not be recorded.

HCLOSE is the head change criterion for convergence, in units of length. When the maximum absolute value of the head change at all nodes during an iteration is less than or equal to HCLOSE, and the criterion for RCLOSE is satisfied (see below), iteration stops.

RCLOSE is the residual criterion for convergence, in units of cubic length per time. When the maximum absolute value of the residual at all nodes during an iteration is less than or equal to RCLOSE, and the criterion for HCLOSE is satisfied (see above), iteration stops.

For nonlinear problems, convergence is achieved when the convergence criteria are satisfied on the first inner iteration.

RELAX is the relaxation parameter used with NPCOND=1 (MICCG). Usually, RELAX=1.0, but for some problems a value of 0.99 or 0.97 will reduce the number of iterations required for convergence. RELAX is not used if NPCOND is not 1.

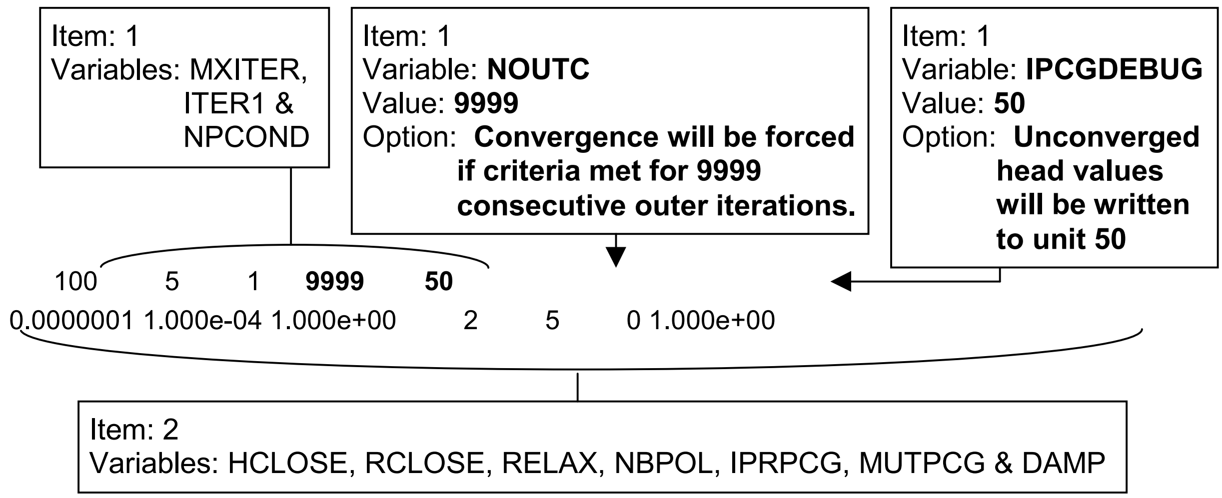
NBPOL is used when NPCOND=2 to indicate whether the estimate of the upper bound on the maximum eigenvalue is 2.0, or whether the estimate will be calculated. NBPOL=2 is used to specify the value as 2.0; for any other value of NBPOL, the estimate is calculated. Convergence is generally insensitive to this parameter. NBPOL is not used if NPCOND does not equal 2.

IPRPCG is the printout interval for PCG. If IPRPCG is equal to zero, it is changed to 999. The extreme head change and residual (positive or negative) are printed for each iteration of a time step whenever the time step is an even multiple of IPRPCG. The printout also occurs at the end of each stress period regardless of the value of IPRPCG.

MUTPCG is a flag which controls printing from the solver. If MUTPCG is not 0, printing from, the solver is suppressed. If MUTPCG=1, the number of iterations is printed, but the lists of extreme head changes and residuals is suppressed. If MUTPCG=2, all printing is suppressed.

DAMP is the damping parameter. DAMP can be used to reduce oscillation when the solver is having difficulty converging due to excessive oscillation. The value of DAMP is a multiplier for the head change calculated each iteration at all cells. Thus, if DAMP is 0.5, the head change is cut in half. If DAMP is 1.0, then PCG2 behaves as it did prior to the addition of this capability. DAMP should be set equal to 1.0 except when there is indication of excessive oscillation. If the value of DAMP is specified as 0.0 or less, it is automatically changed to 1.0.

4.3.4 Example PCG input file



4.4 Module documentation for PCG package

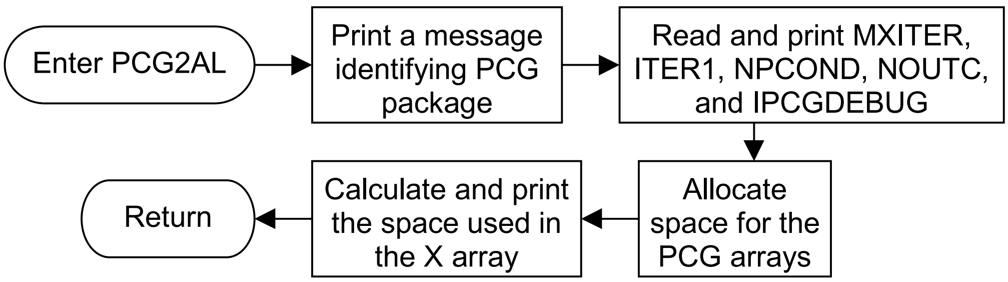
The modified Preconditioned Conjugate Gradient package (PCG-VKD1) has had changes made to two of the three primary modules. These modules are:

PCG2AL Allocates space for arrays used in the package & reads new input parameters
PCG2RP Reads remaining input parameters – not changed
PCG2AP Solves the groundwater flow equations

4.4.1 Changes to module PCG2AL

- Changed tittle printed to listing file
- Read in new input parameters and write options to listing file

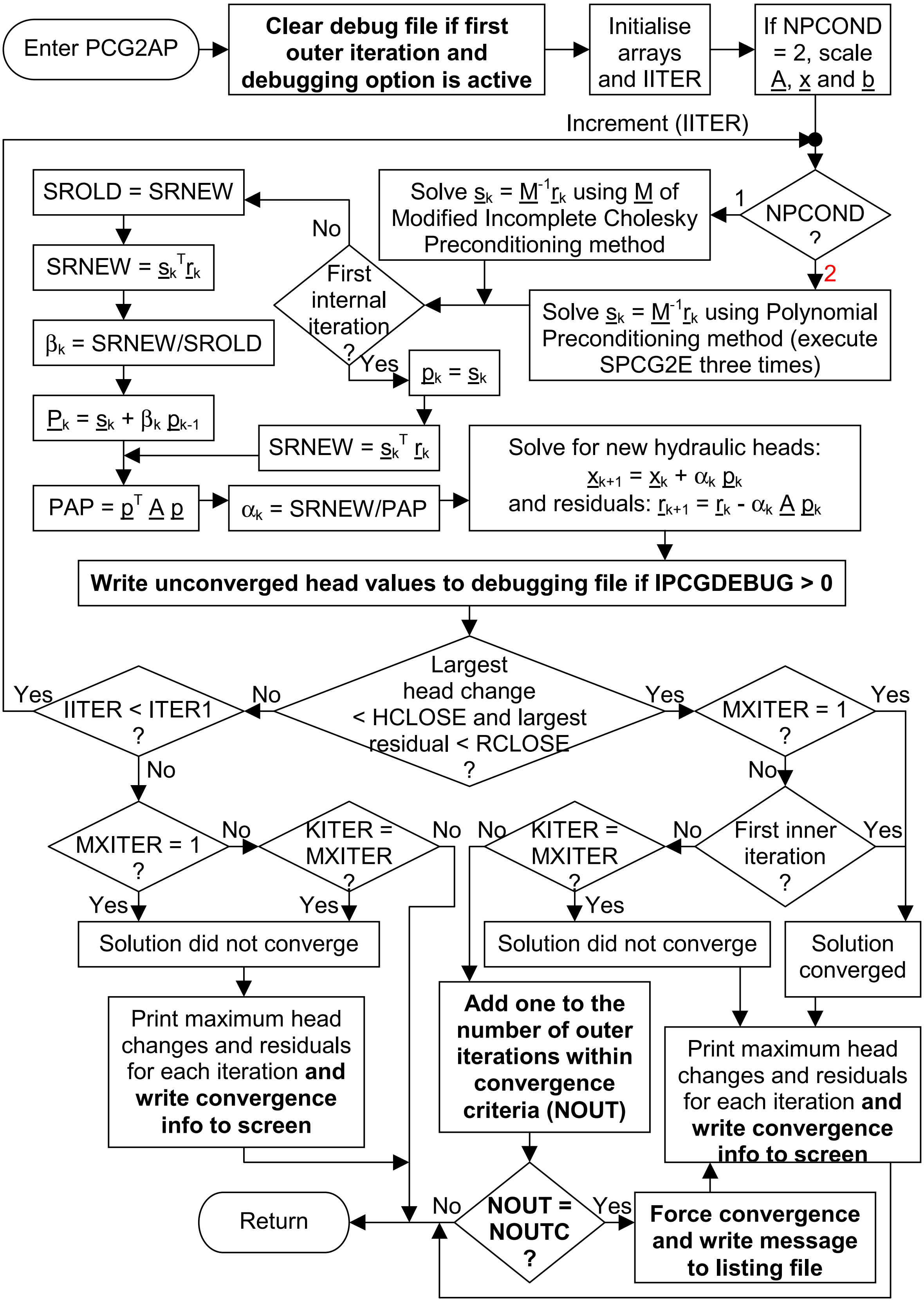
Flow chart for module PCG2AL



4.4.2 Changes to module PCG2AP

- Write unconverged head values to debugging file if IPCGDEBUG > 0
- Force convergence if NOUTC outer iterations are within the convergence criteria and write warning to listing file
- Write convergence information to the screen

Flow chart for module PCG2AP



5 MODFLOW UTILITIES PACKAGE

5.1 Introduction

The utilities package contains routines to read data from and write data to data files. Its main functions include reading one and two dimensional real and integer arrays, and writing arrays to the output file or to binary files. A description of the array input formats is included here for completeness (the BCF package uses the array readers). This is followed by a description of the modifications to allow the use of direct access binary files.

5.2 Array input

The routines for reading arrays have not been modified. The input format for array inputs is reproduced here for completeness. For more details see Harbaugh & McDonald, 1996.

The real two-dimensional array reader (U2DREL), the integer two-dimensional array reader (U2DINT), and the real one-dimensional array reader (U1DREL) read one array-control record and, optionally, a data array in a format specified on the array control record. The control record is read from the input unit number specified for the major option that is requesting the array. For example, the Recharge Package uses U2DREL to read the RECH array. The input unit for the recharge option is contained in IUNIT (8), and accordingly, the RECH array control record is read on this input unit.

For real array reader (U2DREL or U1DREL)

Data:	LOCAT	CNSTNT	FMTIN	IPRN
Format:	I10	F10.0	5A4	I10

For integer array reader (U2DINT)

Data:	LOCAT	ICONST	FMTIN	IPRN
Format:	I10	I10	5A4	I10

5.2.1 Explanation of fields used in input instructions

LOCAT indicates the location of the data which will be put in the array.

- If LOCAT < 0, the sign is reversed to give the unit number from which an unformatted record will be read.
- If LOCAT = 0, every element in the array will be set equal to the value CNSTNT or ICONST.
- If LOCAT > 0, it is the unit number from which data values will be read in the format specified in the third field of the array-control record (FMTIN).

CNSTNT/ICONST is a constant. Its use depends on the value of LOCAT.
If LOCAT = 0, every element in the array is set equal to CNSTNT/ICONST.
If LOCAT is not 0, and if CNSTNT/ICONST is not 0, every element in the array is multiplied by CNSTNT/ICONST.

FMTIN is the format of records containing the array values. It is used only if the first field in the array-control record (LOCAT) contains a positive number. The format must be enclosed in parentheses; for example, (15F5.0) for real data and (15I5) for integer data. The format must be either a standard Fortran format, “(FREE)” which indicates free format, or “(BINARY)” which indicates binary (unformatted) data. When using a free-format control record, the format must be enclosed in apostrophes if it contains one or more blanks or commas. There are only 2 ways to create a binary file that can be read by MODFLOW. The first way is to use MODFLOW to create the file by saving heads in a binary file (but not a direct access binary file – see Section 5.3.1). This is commonly done when it is desired to use computed heads from one simulation as initial heads for a subsequent simulation. The other way to create a binary file is to write a special program that generates a binary file. “(FREE)” and “(BINARY)” can only be specified in free-format control records (see below). Also, “(BINARY)” can only be specified when using U2DREL or U2DINT, and only when the control record is EXTERNAL or OPEN/CLOSE. When the “(FREE)” option is used, be sure that all array elements have a non-blank value and that a comma or at least one blank separates adjacent values.

IPRN is a flag indicating that the array being read should be printed and a code for indicating the format that should be used. It is used only if LOCAT is not equal to zero. The format codes are different for each of the three modules. IPRN is set to zero when the specified value exceeds those defined in the chart below. If IPRN is less than zero, the array will not be printed.

IPRN	U2DREL	U2DINT	U1DREL
0	10G11.4	10I11	10G12.5
1	11G10.3	60I1	
2	9G13.6	40I2	
3	15F7.1	30I3	
4	15F7.2	25I4	
5	15F7.3	20I5	
6	15F7.4		
7	20F5.0		
8	20F5.1		
9	20F5.2		
10	20F5.3		
11	20F5.4		
12	10G11.4		

5.2.2 Free-format control records for array readers

In addition to the fixed format control records described above, free-format can also be used. Values in italics are key words that can be specified as uppercase or lowercase. Each control record is limited to a length of 79 characters.

1. *CONSTANT* CNSTNT
All values in the array are set equal to CNSTNT.

2. *INTERNAL* CNSTNT FMTIN IPRN
The individual array elements will be read from the same file that contains the control record.

3. *EXTERNAL* Nunit CNSTNT FMTIN IPRN
The individual array elements will be read from the file unit number specified by Nunit. The name of the file associated with this file unit must be contained in the name file.

4. *OPEN/CLOSE* FNAME CNSTNT FMTIN IPRN
The array will be read from the file whose name is specified by FNAME. This file will be opened on unit 99 just prior to reading the array and closed immediately after the array is read. This file should not be included in the name file. A file that is read using this control record can contain only a single array. The OPEN/CLOSE option is particularly useful for running simulations that require more than 99 files using a computer that allows only 99 files to be opened simultaneously.

5.3 New features

The only change made to the utilities package was to allow the use of direct access binary output files.

5.3.1 Allow use of direct access binary files

The routine in the Basic package that opens all the files in the NAME file (SBAS5O) allows the user to specify a binary file as a direct access file by entering the word ‘DIRECT’ after the filename. A number specifying the record length of the direct access file (in bytes) should follow this keyword (a value of 1 will ensure that the file opens without an error – see Section 6.2 for details).

This modification to the utilities package allows the head, drawdown and cell-by-cell flow files to be written in direct access format. This is the same format as the files created by the Windows version of MODFLOW (MFWin32) supplied with Groundwater Vistas (Environmental Simulations Inc, 2001) and enables output from the modified MODFLOW code to be processed by Groundwater Vistas. The record length of the file is calculated from the model dimensions when the first record is written, any previous contents of the file are cleared and the file is re-opened with the correct record length. Subsequent records are then added to the file as specified in the Output Control file.

Note: *This option works with code compiled using the Salford FTN77 compiler; it does not work using the Compaq Fortran compiler.*

5.4 Module documentation for utilities package

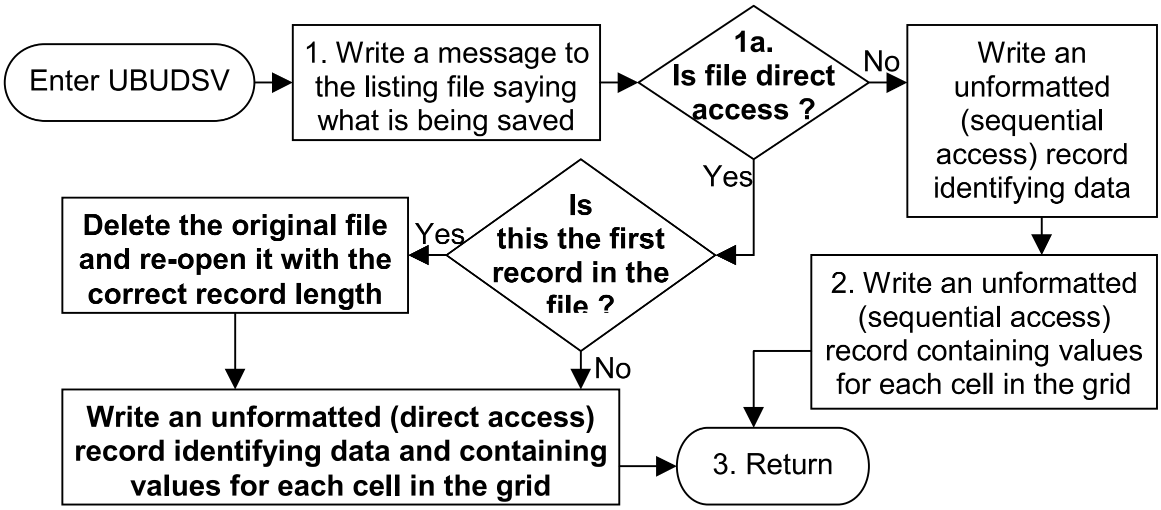
The modified Utility package (UTL-VKD1) has had changes made to two of the modules. These modules are:

UBUDSV Records cell-by-cell flow terms for one component of flow
ULASAV Saves a one-layer array on disk (used for head and drawdown output)

5.4.1 Changes to module UBUDSV

- Allow use of direct access binary files

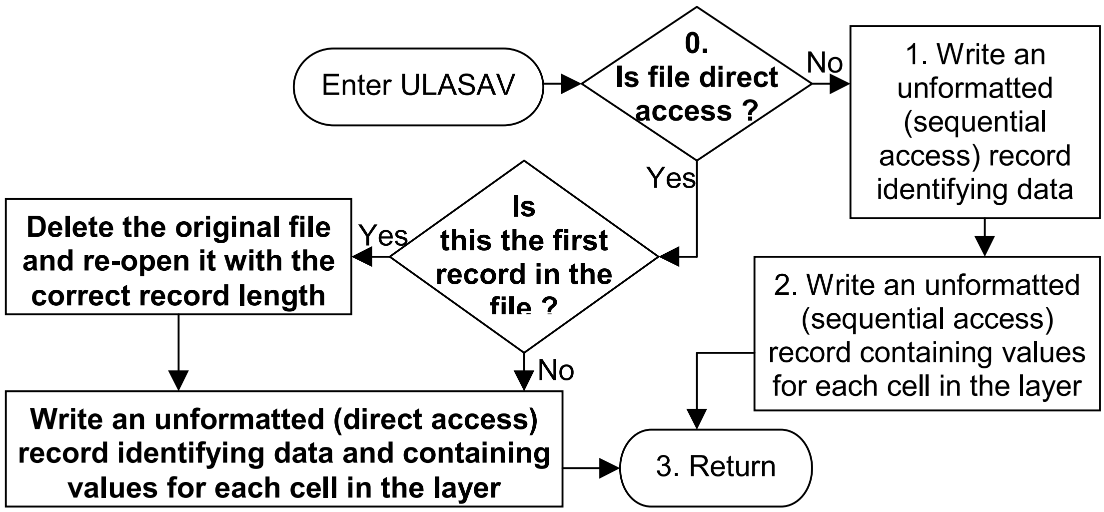
Flow chart for module UBUDSV



5.4.2 Changes to module ULASAV

- Allow use of direct access binary files

Flow chart for module ULASAV



6 MAIN PROGRAM

6.1 Description of changes to main program

The changes to the main program (MF-VKD1.for) mainly consist of changes to the calls to the subroutines of the modified packages (BCF, Stream and PCG). Other changes that have been made include:

- Increased the size of the X array from 1,500,000 to 10,000,000. (This allows the code to use a maximum of 38 MB of memory, however, many operating systems use dynamic storage meaning that the memory requirements will depend on the size of the model being simulated.)
- Allow the progress monitor to be printed to the screen when using the PCG solver (see Section 4.2.1).
- Allow the code to loop back and run a second simulation when using the auto conversion option (see Section 2.2.3).
- Allow the output of calculated transmissivity values to the listing file (see Section 2.2.7).
- Change all the ‘real’ variables to ‘double precision’ variables. This allows for the greater precision required for the calculation of variations in storage with depth.

6.2 Input instructions for the name file

The name file is a special file that is read by the Basic Package. The name file is read on unit 99, as specified in the MAIN program. The name file must specify the names of all files being used in a model simulation except for OPEN/CLOSE files (see Section 6.2.1). As a minimum, there must be a listing file to which all model output intended for printing will be written and one input file for each package being used. (Output that goes to the listing file is frequently referred to as printed output because it is expected that all or parts of this file will be printed.) There may also be additional input files containing array data and output files for saving head, drawdown, and cell-by-cell budget data.

This new version of the MODFLOW modelling code allows some new options to be set (shown in **bold**) in the name file.

For each simulation

1. Ftype Nunit Fname [*DIRECT* Recl] **[> Fname2 [*DIRECT* Recl]]**

The name file must contain one of the above records (item 1) for each file used in the simulation. The first record must be for the listing file; the remaining records can be in any order. All parameters are free format. The length of each record must be 79 characters or less. The items in square brackets are optional.

Any number of comment records, which are indicated by the # character in column one, can be interspersed among the item-1 records. Any text characters can follow the # character. Comment records have no effect on the simulation; their purpose is to allow users to provide documentation about a particular simulation. All comment records that come after the first item-1 record are written in the listing file.

6.2.1 Explanation of parameters in the name file

Ftype is the file type, which must be one of the following character values. Ftype may be entered in uppercase or lowercase:

LIST for the listing file (main output file)
BAS for the Basic Package
BCF for the Block Centred Flow Package
WEL for the Well Package
DRN for the Drain Package
RIV for the River Package
EVT for the Evapotranspiration Package
TLK for the Transient Leakage Package
GHB for the General Head Boundary Package
RCH for the Recharge Package
SIP for the Strongly Implicit Procedure (solver) Package
DE4 for the D4 Direct Solution (solver) Package
SOR for the Slice Successive Over Relaxation (solver) Package
OC for the Output Control Package
PCG for the Preconditioned Conjugate Gradient (PCG2 solver) Package
GFD for the General Finite Difference Package
HFB for the Horizontal Flow Barrier Package
RES for the Reservoir Package
STR for the Stream Routing Package
IBS for the Interbed Storage Package
CHD for the Constant Head Package
FHB for the Specified Flow Package
DATA for formatted (text) files such as those used to save formatted head and drawdown and for input of array data from files that are separate from the primary package input files.
DATA(BINARY) for binary (unformatted) files such as those used to save cell-by-cell budget data and binary (unformatted) head and drawdown data.

Nunit is the Fortran unit to be used when reading from or writing to the file. Any legal unit number on the computer being used can be specified except units 94-99. Unit 99 is used for the name file and for reading arrays using the OPEN/CLOSE option. Units 97 and 98 are used for batch files as explained below. **Units 94-96 are used to create new input files when using the auto-conversion option (see Section 2.2.3)**. Each file must have a unique unit number.

Fname is the filename of the file to be opened. No spaces or commas are allowed in the filename unless the filename is enclosed by single quotes (‘).

DIRECT is a keyword that can be used with DATA(BINARY) files to specify the file as a direct access binary file. DIRECT must be followed by Recl. (*Note: the Groundwater Vistas user interface (Environmental Simulations Inc, 2001), which is currently the Agency’s preferred interface for MODFLOW modelling, only uses direct access files. The tick boxes specifying that model results are in double precision should all be ticked when importing model results. – This option does not work if the code has been compiled using the Compaq Fortran compiler*)

Recl is the record length of the direct access binary file. **For head, drawdown and cell-by-cell flow files the actual record length is calculated in the code, but a number should still be entered so that the file can be opened (recommended value: 1).**

Fname2 is used when an auto-conversion simulation is being run (see Section 2.2.3) to specify the names of the new Basic and Block Centred Flow files that are to be created by the code, and the names of the output files for the second simulation. Fname2 should be preceded by a ‘greater than’ character, ‘>’, and a space. If the second set of binary output files is to be direct access, the keyword DIRECT and the record length Recl should be specified after Fname2.

Example NAME file:

LIST 10 Run38.out **> Run38-part2.out**
Run 38 – VKD auto-conversion run using direct access binary
files
BAS 20 Run38.bas **> Run38-part2.bas**
BCF 11 Run38.bcf **> Run38-part2.bcf**
DRN 13 ..\Run17\Run17.drn
RIV 14 ..\Run17\Run17.riv
WEL 12 ..\Run17\Run17.wel
RCH 18 ..\Run17\Run17.rch
OC 22 ..\Run17\Run17.oc
SIP 19 ..\Run17\Run17.sip
DATA(BINARY) 50 Run38.cbc **DIRECT 1 > Run38-part2.cbc DIRECT 1**
DATA(BINARY) 30 Run38.hds **DIRECT 1 > Run38-part2.hds DIRECT 1**
DATA(BINARY) 31 Run38.ddn **DIRECT 1 > Run38-part2.ddn DIRECT 1**

(note: the “..\Run17\” parts of the filenames above refer to input files in different directories)

6.3 Batch mode

Except as described below, MODFLOW will prompt the user to enter the name of the name file when MODFLOW is started. Once a name file is entered, MODFLOW will run without further user interaction.

Provision has been made to allow a series of simulations to be run without the need to manually restart MODFLOW for each simulation and respond to the prompt for the name file. This is called batch mode and is done through the use of a special file named modflow.bf. If this file exists in the user's current directory (where MODFLOW is run from) when MODFLOW is invoked, the user is not prompted for a name file. Instead, MODFLOW reads the names of the name files from modflow.bf. Each line of the file contains the name of a single name file, and each name can be up to 80 characters long.

When batch mode is used, a report file, named modbatch.rpt, is created. Before starting a simulation, the name of the name file is written to the report file. If the name file exists, the simulation is executed; when done, another message is written to the report file. If a name file does not exist, a message is written to the report file and processing continues with the next name file in modflow.bf.

When batch mode is used to run MODFLOW, two additional file units are used – 97 and 98. These units can not be used by any of the MODFLOW simulations. However, the file units used by one simulation can be the same as those used in any of the other simulations that are executed because each simulation is executed separately.

7 COMPILATION OPTIONS (SALFORD & COMPAQ)

The modified code has been compiled and tested using the Salford (Salford Software Ltd, 1996) and Compaq (Compaq Computer Corporation, 2000) FORTRAN compilers. The compiler control files are included on the CD which accompanies this report.

MODFLOW stores most of its variable arrays within a general array called the X-array. This array contains integer, real and double precision variables. This may cause problems with some FORTRAN compilers; for instance, the 16-bit Salford compiler (DBOS) requires an additional flag (NOCHECK) to be set in order for the code to work.

7.1 Salford

The following options have been used to compile the code using the Salford compiler (FTN77):

/INTL	Sets default integer length to four bytes.
/LIS	Produces a source listing file (to aid debugging).
/DEBUG	Causes the output of information to allow the use of source level debugging facilities.
/SAVE	Do not use the stack for storage of local variables and arrays. Otherwise dynamic storage is used for all local variables and arrays.
/ZEROISE	All static variables and arrays set to zero at start of execution.
/DREAL	changes all real variables (four byte) to double precision (eight byte)

All of the options above are optional, with the exception of INTL when compiling with the 16-bit compiler, and DREAL which is required to allow accurate calculation of variations in storage with depth.

7.2 Compaq

The only change made to the standard compiler options for the Compaq Visual Fortran compiler (Version 6.5) was to change the ‘default real kind’ from 4 to 8 (double precision).

Note: *The option to write direct access binary output files does not work if the code has been compiled using the Compaq Fortran compiler.*

To set up a workspace select:

- ‘File’ then ‘New’ then ‘Projects’ then ‘FORTRAN Console Application’.
- Ensure ‘Location’ is path to directory containing source code.
- Ensure ‘Project Name’ is the same as directory name containing source code.
- Press return.
- Accept all other options to complete new workspace.
- Select ‘Project’ then ‘Add to Project’ then ‘Files’.
- Select all source code files.
- The compiler options can be found in ‘Project’ then ‘Settings’ then ‘FORTRAN’. Select the ‘Fortran Data’ category and change the ‘default real kind’ from 4 to 8 and click ‘OK’.
- The executable can now be compiled under the ‘build’ menu (or by pressing F7).

8 PRACTICAL APPLICATION

8.1 Introduction

This section of the report is intended to provide guidance for modellers who are already fairly familiar with MODFLOW, who wish to use this modified version of the code to simulate aquifers where parameters vary with depth. The guidance presented here is based on the experience gained in the initial testing of the code and also on some modelling projects where the code is being further tested.

It should be noted that in some of the following subsections guidance has been aimed towards modellers using the Groundwater Vistas user interface (Environmental Simulations Inc, 2001), which is currently the Agency's preferred interface for MODFLOW modelling. It should be noted here that model results can only be imported into Groundwater Vistas if the binary output files are written as direct access files (see Section 6.2), and that the tick boxes specifying that the model output is in double precision should be checked on the Groundwater Vistas 'Import model results' dialogue box.

The following subsections include guidance on getting started, obtaining initial conditions for time variant simulations, converting existing steady-state MODFLOW models to VKD models, and converting steady-state models to time-variant models. This is followed by a summary of the effects of VKD and VSD on simulation results.

8.2 Getting started

The MODFLOW executable should either be located in the directory containing the model input files and run from there, or else can be located on the system path defined in the "autoexec.bat" file on the "C:\" drive (or for windows NT/2000; under "Start" → "Settings" → "Control Panel" → "System" → [→ "Advanced"] "Environment"). To run the program bring up a command prompt or "DOS box", go to the directory containing the input files, and type in the name of the MODFLOW executable (eg, "MF-VKD1"). The user is then asked to enter name of the name file before the simulation commences.

The user should either use the executable included on the accompanying CD, or can compile the code themselves (see Section 7).

8.3 Obtaining initial conditions for a time variant VKD simulation

When undertaking time-variant groundwater simulations it is important that the initial conditions used to start the simulation are realistic. Often, initial conditions are taken from the results of a steady-state simulation that represents average conditions in the aquifer. However, as groundwater conditions change with the seasons and respond to changes in abstraction, real aquifers are rarely in a condition that could be described as a 'steady state'.

The problem of initial conditions can be approached in a couple of different ways. The first, most commonly used method is to run the model through a couple of years, before the time period of interest, so that the model reaches 'dynamic balance'. A second, less commonly used method is to estimate the change in groundwater storage over the model over a time period of about a month, and add this to the recharge in a 'time-instant' steady state model.

The problem of initial conditions becomes more of an issue when attempting to model an aquifer where hydraulic conductivity varies with depth. The problem is in how to define a realistic hydraulic conductivity profile, when several different profiles can give the same transmissivity values, and hence the same steady state solution. *The effect of different hydraulic conductivity profiles on simulation results can only be assessed by looking at both high and low water table conditions.*

Environment Agency staff and consultants that have worked with the code have suggested different approaches to obtain a hydraulic conductivity profile and initial conditions for a time variant simulation where the water table and therefore transmissivity has a high seasonal variation.

One way to get an idea of how the hydraulic conductivity varies with depth is to obtain estimates of the transmissivity for high and low water table conditions, and then construct a hydraulic conductivity profile that gives the same high and low transmissivity values for the different water table levels. The transmissivity values can be estimated in the normal ways (ie, from observed groundwater gradients and flow estimates) and can be tested through model simulations. For example, the approach suggested to simulate the Upper Lee catchment involves bypassing the steady state phase of the modelling, and instead undertaking two time variant simulations with different distributions of specified transmissivities. The time variant simulations start with some representative initial heads (ie, contoured observed heads, or heads at surface elevation), and simulate a shortened historical sequence of recharge and abstraction. The first model has lower transmissivities, and is designed to reproduce the observed low flows over two or three years. The second model has higher transmissivities, designed to reproduce high flows. The transmissivity distributions are then compared, and combined to derive the VKD parameters. The advantages of this approach are likely to be:

- The problems associated with steady state simulations are avoided.
- The effect of storage can be evaluated in the same simulations.

The disadvantages are likely to be:

- The effects of the storage coefficients on simulation results may not be so easily differentiated from the effects of the VKD parameters. Some users report that this differentiation is relatively straightforward.
- It takes a relatively long time for the time-variant models to run and for any parameter changes to be assessed. This is not such a large problem with modern computers

Care needs to taken to ensure that the model has reached dynamic balance.

Another approach, proposed for modelling the River Bourne catchment, is to construct two time-instant steady state (TISS) models, one for maximum and one for minimum water table conditions. The idea would be to avoid the auto-conversion option available in the code (and hence avoid a 'measles plot' of hydraulic conductivity values) and use the same VKD parameters directly in each model. Selecting times of maximum and minimum groundwater heads (when there is little or no change in groundwater storage) avoids the difficulty in estimating the change in storage across the model. The advantages of this approach are likely to be:

- The ability of the model to reproduce maximum and minimum heads can be assessed before moving to a time variant model.
- The effect of the VKD parameters can be evaluated independently of the storage coefficients.
- The run times of the steady state models should be quick compared to those of time-variant models (provided there are no convergence problems).
- The results of either of the steady state models could be used as stable initial conditions for a time-variant simulation.

The disadvantages of this approach are likely to be:

- There may be difficulties in getting the steady state models to converge without using the auto-conversion option (although the PCG2 solver in MODFLOW is more robust than the point SOR solver used in the Birmingham University code).
- The assumption of zero change in storage at maximum and minimum water table conditions may not be valid.

A third approach, which was employed for the Itchen and Mimram catchment models, consists in employing as initial conditions the heads produced in a time variant run where the model reaches dynamic balance based on a representative recharge pattern.

The above approaches avoid the use of the auto-conversion conversion option, and allow the VKD parameters to be input directly (rather than the base hydraulic conductivity being calculated by the MODFLOW code, resulting in 'measles plots' of hydraulic conductivity). From the points discussed above it is not immediately clear which of the three approaches is more useful. It was suggested that both the approaches should be investigated, and the relative merits of each could be compared once they had both been attempted.

8.4 Converting an existing steady-state MODFLOW-96 model to a VKD model: auto-conversion option

8.4.1 Setting up the MODFLOW input files

These notes describe how to convert an existing MODFLOW model to VKD model. The instructions below should be used in conjunction with the Input Instructions (Section 2.3).

All the following changes are made to the Block-Centred Flow (BCF) input file:

- Set the Steady State Flag (ISS) to 2 (if converting specified transmissivity layer(s) – LAYCON = 0 or 2) or 3 (if converting specified average hydraulic conductivity layer(s) – LAYCON = 1 or 3). This flag is the first number in the BCF file (row 1, columns 1 to 10).

- Set the Transmissivity Output Flag (ITRANS) to one. This allows the (inter-nodal) transmissivities calculated by VKD run to be compared against those from the specified transmissivity run (output to the listing file). This flag is located in columns 91 to 100 of row 1.

- Change the Layer Type (LAYCON) for the layer(s) of interest to 4 (unconfined VKD) or 5 (confined / unconfined VKD). Note: all other layer types (0 to 3) will be simulated in the normal way. These flags are located on the second line of the BCF file.

- Enter an array for the Bottom Elevations (BOT) if the original layer type was 0 or 2. If the original layer type was 1 or 3, the BCF file should already contain an array of bottom elevations. This array is located after the Transmissivity / Hydraulic Conductivity array.

- Enter an array for the Top Elevations (TOP) if specifying a confined / unconfined VKD type layer (LAYCON=5), and the original layer type was 0 or 1. If the original layer type was 2 or 3, the BCF file should already contain an array of top elevations. This array is located after the "Vcont" array if this is not the lowest layer in the model, or after the BOT array if it is the lowest layer – see the Input Instructions.

- Enter an array for the Thickness of the Upper Zone of the layer (the zone where hydraulic conductivity increases – commonly associated with the zone of fluctuation of groundwater heads). This array should be entered after all other arrays for the layer.

- Enter an array for the Hydraulic Conductivity Gradient Factor (VKGRAD – defined as the increase in hydraulic conductivity per unit rise in elevation, divided by the base hydraulic conductivity). This array should be placed after the upper thickness array.

- Enter an array for the Maximum Hydraulic Conductivity Factor (VKMAX – defined as the maximum hydraulic conductivity allowed in the upper zone of the layer, divided by the base hydraulic conductivity). This array should be placed after the VKGRAD array.

- It is a good idea to test out values for the parameters above, by checking the transmissivity given at different groundwater heads. This can be done by:

    (i) calculating the transmissivity distribution for different heads in a spreadsheet and exporting it in matrix form as a text file

    (ii) visualising the transmissivity distribution by importing the matrix contained in the text file in a spare property array in Groundwater Vistas (Environmental Simulations Inc, 2001)

Arrays of property values (upper thickness, hydraulic conductivity gradient factor, and maximum hydraulic conductivity factor) can be stored in spare property arrays in Groundwater Vistas (Environmental Simulations Inc, 2001) (eg, dispersivity, chemical reactions, and initial concentrations). The arrays can then be inserted into the BCF package by exporting them to a file [Props → Export → Property values (Matrix)…], and copying the array from that file to the relevant part of the BCF package. Note: if values are exported from Groundwater Vistas in exponential format, there will be 3 figures in the exponential part of the number. Although FORTRAN (the code MODFLOW is written in) usually reads and writes exponentials with two figures, it will also read three, so the values do not need to be changed.

Having made these changes, the BCF file should be ready for use. Save the file with a new filename. It is also good practice to create a new Basic (BAS) file with an updated title in the first two lines. The next step is to create a "name file" which contains a file descriptor ("LIST", "BAS", "WEL", etc), a unit number (as shown in the [Model → Modflow → Packages] section in Groundwater Vistas), and a file name for each file used in the simulation (see Section 6.2). Binary output files (head, drawdown, and cell-by-cell flow files) should be identified by the descriptor "DATA(BINARY)". Groundwater Vistas (Version 2) creates a name file (*.nam) when the MODFLOW input files are created, this file can be edited to use the filenames of the newly created files. All the other MODFLOW input files (WEL, RIV, DRN, etc) should be the same as for the original model.

Now run the MODFLOW-VKD application (using a DOS or command Box), and enter the name of the "name file" when prompted. Two simulations should run: the first a specified transmissivity or average hydraulic conductivity simulation, the second a VKD simulation.

8.4.2 Checking the simulation results

The outputs of the two simulations should be compared to verify that they give the same results (eg, inspect the drawdown file to check that the largest difference in head is small – say, less than 0.1 m). These results should also be compared against the original MODFLOW model (use the [Plot → File Operations → Compare Two Files…] option in Groundwater Vistas). Cell-by-cell flows can be compared in a similar way, and inter-nodal transmissivities can be compared in the output files from each simulation (if the Transmissivity Output Flag is not zero).

If there is a significant difference between the results of the two simulations, or if the second simulation does not converge, try changing the solver, or the solver options (PCG solver recommended). For example, the drawdowns produced in the second simulation will have values equal or very close to the opposite of the default value used for dry cells if dry cells are generated in the second simulation. In this case, decreasing the damping parameter reduces the oscillation of heads during the solving process and can avoid the generation of dry cells. Another way of avoiding differences between the results of the two simulations or non convergence is to reduce the Hydraulic Conductivity Gradient Factor and/or to increase the Upper Thickness, as this should reduce the non-linearity of the problem.

As well as checking the results of the two simulations, it is necessary to check the parameter values calculated by MODFLOW-VKD to make sure that the values themselves and their distribution are sensible. These parameters are the base hydraulic conductivities and the elevation of the change from constant to varying hydraulic conductivity (VMID – point of inflection). One way to do this is to copy the relevant arrays from the BCF package created by MODFLOW-VKD (second BCF file) into separate files, and import them into property arrays in Groundwater Vistas [Props → Import → Matrix…] (eg, the hydraulic conductivity and dispersivity property arrays). Then sort the arrays [Props → Property Values → Sort Zones], reset colours to grey scale [Props → Property Values → Reset Colors to Gray Scale], and reset the zone colours [Props → Property Values → Reset Zone Colors]. The property values can then be checked in either the database or by tracking the mouse pointer over the model cells.

8.5 Creating a time-variant VKD/VSD model from a steady-state VKD model

This Section describes how to set up the BCF file for a time variant VKD / VSD model. All the other packages (BAS, WEL, RIV, etc) should be created in the usual way (eg, using Groundwater Vistas (Environmental Simulations Inc, 2001), text editors, spreadsheets or FORTRAN routines). Make sure that the initial heads in the Basic (BAS) file are those calculated in the steady state VKD simulation (strictly speaking, these should be the heads produced from the second simulation, however, as the results of the two simulations should be the same, the heads produced from the first simulation can be used instead – reproduced in the 2nd Basic file).

Using the BCF package created in the steady state simulation (second BCF file) as a starting point, the following changes should be made:

- Change the steady state flag (ISS) to zero. This flag is the first number in the file (row 1, columns 1 to 10).
- Set the Transmissivity Output Flag (ITRANS) to zero. If this flag is non-zero, inter-nodal transmissivities are written to the output file every time step, leading to very large output files. This flag is located in columns 91 to 100 of row 1.
- Add the Primary Storage Coefficient array (Sf1). If the layer is unconfined (LAYCON = 1 or 4), this array contains values of Specific Yield. If the layer is confined or convertible (LAYCON = 0, 2, 3 or 5), this array contains values of Confined Storage Coefficient. This array should be placed before the (base) Hydraulic Conductivity array.
- If the layer type is convertible (LAYCON = 2, 3 or 5), add the Secondary Storage Coefficient (Sf2). This array is always Confined Storage Coefficient[1]. This array should be placed before the Top Elevation array.
- Enter an array for the elevation of the point of inflection of unconfined storage (VSMID – defined as the elevation where storage changes from being constant to varying with depth). This array should be placed after the Maximum Hydraulic Conductivity Factor array.

[1] A possible exception to this might be in the case where a reduction in specific yield towards the top of the aquifer is being modelled (e.g. when simulating groundwater flooding). In this case the reduced Specific Yield should be entered in this array, and the "top" of the layer should be set to the elevation where the specific yield is believed to reduce. *Important note: using this approach means that the transmissivity will not increase once heads rise above the "top" of the layer. The modeller must verify that this approximation is valid (i.e. the contribution to the layer's transmissivity above this point is negligible – however high the groundwater heads rise) before implementing it in the model.*

- Enter an array for the Storage Gradient Factor (VSGRAD – defined as the increase in specific yield per unit rise in elevation, divided by the base specific yield). This array should be placed after the VSMID array.
- Enter an array for the Maximum Storage Factor (VSMAX – defined as the maximum specific yield allowed in the upper zone of the layer, divided by the base specific yield). This array should be placed after the VSGRAD array.
- If no variation in storage is required, set VSMAX to one. It is good practice to examine the effects of VKD on simulation results before including variation in storage (VSD) as well.

Having made these changes, the BCF file should be ready for use. Save the file with a new filename. It is also good practice to create a new Basic (BAS) file with an updated title in the first two lines. The next step is to create a new “name file”. All the other MODFLOW input files (WEL, RIV, DRN, etc) should be the same as for a normal time-variant MODFLOW model.

Now run the MODFLOW-VKD application (using a DOS or command Box), and enter the name of the “name file” when prompted. A single VKD / VSD simulation should run. If the PCG solver is being used, the progress of the model is written to the screen (progress is not displayed for other solvers).

Due to the non-linearities of VSD, simulations with high values of VSGRAD are prone to instabilities, and may not converge easily. To avoid this, change the solver options. Alternatively, use low values of VSGRAD (between 0 and 0.5) and / or low values of VSMAX (between 1 and 1.5).

8.6 Effect of VKD/VSD on simulation results

Including variations in hydraulic properties with depth (VKD or VSD) in a numerical model effects the time-variant behaviour of the model, specifically the variability of the groundwater heads and flows. It should be recognised, however, that other model parameters can have similar effects on head and flow variability. One such parameter is the aquifer storage. Increasing the (non-depth dependant) storage in a model has the effect of reducing the variability of both heads and flows. Although these effects are different from those produced by a model with VKD (which increases flow variability and reduces head variation), it is not always easy to distinguish which parameters need changing without detailed analysis of the field data. Another important model input that has a direct effect of head and flow variability is the distribution of recharge through time. There are many different methods available for estimating recharge rates, but their time dependence is often harder to determine due especially to the effects of storage and throughflow in the unsaturated zone. In addition to the storage and recharge other parameters such as pumping rates, river coefficients and leakance between model layers can also affect head and flow variability, although some of these may only provide local changes.

8.6.1 Effect of VKD on simulation results

The effects of including a variation of hydraulic conductivity with depth are on the variation through time of groundwater heads and flows. Provided that the average transmissivity values remain the same, increasing the rate at which hydraulic conductivity reduces with depth has the effect of reducing head variations, whilst increasing the variation of flows to rivers and streams. However, although head variations are reduced in most areas, they are increased slightly near to outflow points such as rivers, streams and wells with variable pumping rates.

8.6.2 Effect of VSD on simulation results

Including variations in storage with depth also affect the variations of heads and flows through time. Provided that the storage at the average groundwater head level remains the same, increasing the rate at which storage reduces with depth has the effect of reducing heads slightly at both high and low points through time, whilst raising them slightly at intermediate points. The same is true for flows to rivers and streams.

9 POST PROCESSING UTILITIES

9.1 Introduction

As the modifications to the MODFLOW code have been tested using various different models, a number of post-processing utilities have been written and modified to aid the investigation of model results. This section includes descriptions of these utilities and instructions on how to use them.

9.2 Modifications to the USGS Zonebudget program (Zonebdgt8.exe)

Zonebudget is a FORTRAN program developed by the USGS to “compute and print volumetric budgets over subregions of a flow system that is being simulated using the USGS Modular Three-Dimensional Finite-Difference Ground-Water Flow Model” (Harbaugh, 1990).

At an early stage of the project it was necessary to know the water balance error at every node of the test models to insure that the acceptance criteria had been met. This was attempted using Zonebudget, with a different zone specified for each node in the model. However, the output files produced by the original version of the code were found to be very large and not very easy to interrogate, especially when investigating the results of time-variant models.

To avoid the problem of large files, and to make the program easier to use, the following changes were made:

- The format of the output file was changed to a table format that could be easily imported into a spreadsheet such as Excel.
- The option was added for the time step information to be read from the corresponding basic (BAS) file so that times can be included in the output files.
- The option was added for creating budget files for each zone and/or for each flow component.
- Modifications were made to allow either sequential or direct access cell-by-cell flow files to be read (although double precision cell-by-cell flow files are not recognised and must first be converted using the program ‘mf-convert.exe’).
- Additional changes were made to allow the program to read zone number outputs from Groundwater Vistas (Environmental Simulations Inc, 2001).

9.2.1 Inputs

Zonebudget should be run from a command prompt or DOS box by typing in the name of the executable file ('Zonebdgt8.exe'). Once the program starts the user is prompted to provide the following inputs:

1. ' Enter the name of a LISTING FILE for results:'

The user should enter the name of the main output file (eg, 'Run001.zbd'). The names of the individual files for each zone and flow component will be based on this filename (eg, 'Run001-Zone001.csv', 'Run001_________Storage.csv', etc).

2. ' Enter the name of the MODFLOW BAS file (reads timestep info; CR to skip):'

The user can either enter the name of the MODFLOW basic (BAS) file that was used in the simulation, or the input can be left blank (press the carriage return key). The bas file is used to calculate the simulation time corresponding to each timestep, if a bas file is not entered then zeros are entered in the time column of the output tables.

3. ' Enter the name of the file containing CELL-BY-CELL BUDGET TERMS:'

Here the name of the binary file containing cell-by-cell flows should be entered (eg, 'Run001.cbc'). This file can be either sequential access (as produced by MODFLOW-96) or direct access (as produced by MFWin32). MODFLOW-VKD can produce either sequential or direct access files.

If a bas file was not entered at step 2 then the following prompt is given:

3a. ' Enter number of rows, columns and layers:'

The user should then enter the number of rows, columns and layers in the model simulation (this information can be found in the bas file).

There is then a pause whilst the program scans the contents of the cell-by-cell flow file. This can sometimes take some time if it is a large file. A summary of the information in the cell-by-cell flow file is then printed to the screen.

4. ' Enter a TITLE to be printed in the listing:'

This title will be written in all the output files and can be used to provide information on the simulation and the zones used.

5. ' Enter the name of your ZONE INPUT FILE (CR for interactive):'

The zone input file can either be in the original format used for the USGS zonebudget program (Harbaugh, 1990), or a zone number file such as those produced by Groundwater Vistas (Environmental Simulations Inc, 2001). The zone number file contains the row, column and layer number of each cell followed by the zone number for that cell. The values for each cell should start on a new line of the file. All or some of the cells in the model domain can be specified (cells omitted from the zone number file will be considered as zone zero). The zone number file can be exported from Groundwater Vistas (Select the property zone to export then choose [Props]→[Export]→[Zone Numbers…] from the menu. Don't tick the 'Export in Matrix Format' box).

If no zone input file name is entered then the user is prompted to enter zones interactively in blocks, specifying minimum and maximum rows, columns and layers for each block.

The highest zone number in the zone file is then written to the screen.

6. ' Remove obsolete columns from output files (Y/N)? '

If ‘yes’ is chosen for this option the output is initially written to temporary files. Once the program has finished running, these temporary files are scanned for columns that contain only zeros. These columns are removed from the final sets of output files. Any files in which all the columns contain zero values are permanently deleted.

7. ‘Create output file for each zone?’
If ‘yes’ is typed for this option a different output file is produced for each specified zone.

8. ‘Create output file for each boundary type?’
If ‘yes’ is typed for this option a different output file is produced for each component of the flow budget.

9. ' Choose the option for specifying when budgets are calculated:'
   ' A = ALL times stored in the budget file.'
   ' P = For each time stored in the budget file, PROMPT user.'
   ' L = Enter a LIST of times.'

The user should then specify which option is required. If the last option is chosen, up to ten different time steps can be specified.

The program now has enough information to process the cell-by-cell flow file. Messages are printed to the screen telling the user how far through the file the program is.

9.2.2 Outputs

The main output file (the listing file entered at the start of the program) contains a record of the inputs to the zonebudget program, which can be checked at a later date to verify all the inputs.

The remaining output files contain tables of flows for each zone and component of flow. Each file contains a header with the title that was entered at step 4, a description of which zone or component of flow the table is for, and a record of the name of the zone file. This is followed by a table of flows in and out of each zone and boundary condition against the stress period, time step and simulation time. The files are given the extension ‘.csv’ so that they are automatically associated with Excel.

9.3 Program to calculate nodal water balance errors (WB-Error-dp.exe)

In order to facilitate the calculation of nodal water balance errors across a model, a FORTRAN program was written that reads the contents of a (direct access) binary cell-by-cell flow file, calculates both the absolute and percentage flow balance errors at every model node, and writes them to another cell-by-cell flow type binary file.

The program assumes that all the cell-by-cell flows (ie, from the BCF, stream and well packages, etc) have been written to the same file (on the same unit number). It also assumes that the file is in direct access format rather than sequential access format (see Section 6.2 and Section 9.4 below), and that it contains double precision values.

The program first asks the user to enter the name of the CBC (cell-by-cell flow) file to read. Once this has been entered (eg, 'Run001.cbb') the user is prompted for the name of the CBC type file of water balance errors that they want to create (eg, 'Run001-err.cbx'). This is followed by a request to choose from three different output options: absolute errors, percentage errors, or both. The program then reads the input CBC file, calculates the water balance errors for each node at each time step in the file, and writes the values to the output file. The output CBC file is in the same format as the input CBC file, but with title texts of 'Flow Balance Err' for absolute errors, and '% Flow Bal Err ' for percentage errors.

9.4 Programs to change the formats of MODFLOW binary output files

To avoid conflicts between versions of MODFLOW that use direct or sequential access binary files and utilities that use only one or the other, five FORTRAN programs have been written to convert MODFLOW binary output files from one format to the other. In addition to these two other utilities have been written to convert MODFLOW binary heads files to formats compatible with spreadsheets and plotting software. These seven programs are:

'cbc2dos.exe'	Converts CBC files from direct access to sequential access
'cbc2gv2.exe'	Converts CBC files from sequential access to direct access
'heads2dos.exe'	Converts heads (or drawdown) files from direct access to sequential access
'heads2gv2.exe'	Converts heads (or drawdown) files from sequential access to direct access
'mf-convert.exe'	Converts head, drawdown and cell-by-cell flow files between different combinations of direct and sequential access, and single and double precision.
'heads2csv.exe'	Converts heads files from direct/sequential format to comma separated variable (*.csv) format (text file for importing into spreadsheets)
'heads2tp.exe'	Converts heads files from direct/sequential format to TecPlot input format (*.plt)

All the programs require an input filename (of an existing MODFLOW output file) and an output filename. The first four programs also have the option to create a text (ASCII) file of head, drawdown or flow values. The last two programs require additional information on the specific time step or layer to export.

9.5 Programs to combine or split components of cell-by-cell flow files

The output options in MODFLOW allow the user to save the cell-by-cell flow terms from different packages to either one single file or many different files. Different post processing packages, such as those described above and others such as MODPATH, require all the cell-by-cell flows to be in a single file. Two FORTRAN programs have been written to either combine or split up cell-by-cell flow files depending on what is required. These programs are:

'cbc-split.exe'	Splits a single CBC flow file into different files containing one or more components of the flow.
'combine-cbb.exe'	Combines up to ten CBC flow files with the same model dimensions into one single CBC flow file.

Each program will prompt for the names of input and output filenames, and may require additional information or print an error message if a problem occurs. Both programs only work with single precision, direct access cell-by-cell flow files.

9.6 Program to extract time variant flows from the MODFLOW listing file (TVF-win5.exe)

This program reads the MODFLOW listing file and extracts the flow rates or cumulative volumes from the volumetric budget reported at the end of each time step. These values are then written to a new file in a table format, separated by commas, which can be opened in a spreadsheet, where the values can easily be plotted to show the variation of flows over time.

REFERENCES

Compaq Computer Corporation, 2000. *Compaq Visual Fortran Standard Version 6.5 User Guide.* Compaq Computer Corporation.

Cross G A, Rushton K R & Tomlinson L M, 1995. *The East Kent Chalk Aquifer during the 1988-92 Drought.* Journal of the Institution of Water and Environmental Management, 9, pp. 37-48.

Environment Agency, 1999. *Representation of the Variation of Hydraulic Conductivity with Saturated Thickness in MODFLOW. Stages I & II. Code Changes and Testing Against Birmingham University Code.* Environment Agency. ISBN: 0 857 05194 7.

Environment Agency, 2002. *Enhancements to Modflow, variations in hydraulic conductivity and storage with depth.* Environment Agency. ISBN: 1 857 05765 1.

Environmental Simulations Inc, 2001. Guide to Using Groundwater Vistas, Rendon, Virginia, USA.

Foster S S D & Milton V A, 1974. *The permeability and storage of an Unconfined Chalk Aquifer.* Hydrological Sciences – Bulletin – des Sciences Hydrologiques, XIX, 4, pp. 485-499.

Harbaugh, A W, 1990, A computer program for calculating subregional water budgets using results from the U.S. Geological Survey modular three-dimensional ground-water flow model: U.S. Geological Survey Open-File Report 90-392, 46 p.

Harbaugh A W & McDonald M G, 1996, *User's documentation for the U.S. Geological Survey modular finite-difference ground-water flow model*: U.S. Geological Survey Open-File Report 96-485.

Hill M C, 1990. *Preconditioned Conjugate-Gradient 2 (PCG2), A Computer Program for Solving Ground-Water Flow Equations.* U.S. Geological Survey.

Kladias, M P & Ruskauff G J, 1996. *Computer Note on Implementing Spatially Variable Anisotropy in MODFLOW.* Geraghty and Miller, Inc and INTERA, Inc.

Lahey Computer Systems, Inc, 1995-1998. *FORTRAN 95 User Guide.* Lahey Computer Systems, Inc.

McDonald M G & Harbaugh A W, 1988. *A modular three-dimensional finite-difference ground-water flow model.* U.S. Geological Survey, Techniques of Water Resources Investigations. Book 6, Chapter A1.

Owen M, and Robinson V K, 1978. *Characteristics and Yield in Fissured Chalk.* Institution of Civil Engineers, Symposium on Thames Groundwater Scheme, Paper 2, 33-49.

Prudic D E, 1989. *Documentation of a Computer Program to Simulate Stream-Aquifer Relations Using A Modular, Finite-Difference, Ground-Water Flow Model*. U.S. Geological Survey.

Rushton K R, & Chan Y K, 1976. *Pumping test analysis when parameters vary with depth.* Ground Water. **14**, 82-87.

Rushton K R, Connorton B J & Tomlinson L M, 1989. *Estimation of the groundwater resources of the Berkshire Downs supported by mathematical modelling.* Quarterly Journal of Engineering Geology, London, Vol. 22, pp. 329-341.

Rushton K R & Fawthrop N P, 1991. *Groundwater Support Of Stream Flows In Cambridge Area, UK*. Hydrological Basis of Ecologically Sound Management of Soil and Groundwater. IAHS Publication No. 202, 367-376.

Rushton K R & Rathod K S, 1980. *Aquifer response due to zones of higher permeability and storage coefficient*. Journal of Hydrology, 50, 299-316.

Rushton, K R, & Redshaw, S C, 1979. *Seepage and Groundwater Flow*. John Wiley & Sons, Ltd.

Rushton K R, Smith E J & Tomlinson L M, 1982. *An Improved Understanding Of Flow in a Limestone Aquifer Using Field Evidence and Mathematical Models.* Journal of the Institution of Water Engineers and Scientists, Vol. 36, pp. 369 -387.

Salford Software Ltd, 1996. *FTN77 User's Guide.* Salford Software Ltd.

Salmon S, Chadha D & Smith D, 1996. *Development of a Groundwater Resource Model for the Yorkshire Chalk.* Journal of the Chartered Institution of Water and Environmental Management, 10, pp. 37-48.

Southern Water Authority, 1979. *Itchen Groundwater Regulation Scheme; Final report on the Candover Pilot Scheme.* Southern Water Authority, Hampshire.

APPENDIX A
Modified computer code for BCF package with changes highlighted

COMPILER OPTIONS: listing intl no_persist nodclvar nomap nocheck logl dynm offset dreal noansi nopagethrow
                   nosilent no_optimise warn73 nolink no_link

```
C-Changes to the code are highlighted with the following headings:
C
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
C-Environment Agency (UK)-2.Variable storage with depth (VSD)>>>>>>>>>>>
C-Environment Agency (UK)-3.Auto-conversion for steady state VKD>>>>>>>>
C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
C-Environment Agency (UK)-5.Inter-nodal transmissivities>>>>>>>>>>>>>>>>
C-Environment Agency (UK)-6.X & Y-dir transmissivities>>>>>>>>>>>>>>>>>>
C-Environment Agency (UK)-7.Explicit transmissivity calculation>>>>>>>>>
C-Environment Agency (UK)-8.Output calculated transmissivities>>>>>>>>>>
C-Environment Agency (UK)-9.Bug fix>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
C-Environment Agency (UK)-10.K gradient independant of kbase>>>>>>>>>>>>
C-Environment Agency (UK)-11.Output calc trans to binary file>>>>>>>>>>>
C-Environment Agency (UK)-12.Bug fix>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
C
C These headings are followed by the original USGS MODFLOW-96 code
C (commented out), followed by the modified code. All the code changes
C are followed by the following:
C
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
c      SUBROUTINE BCF5AL(ISUM,LENX,LCSC1,LCHY,LCBOT,LCTOP,LCSC2,LCTRPY,
c     1  IN,ISS,NCOL,NROW,NLAY,IOUT,IBCFCB,LCWETD,IWDFLG,LCCVWD,
c     2  WETFCT,IWETIT,IHDWET,HDRY,IAPART,IFREFM)
C-Environment Agency (UK)-10.K gradient independant of kbase>>>>>>>>>>>>
c      SUBROUTINE BCF5AL(ISUM,LENX,LCSC1,LCHY,LCBOT,LCTOP,LCSC2,LCTRPY,
c     1  IN,ISS,NCOL,NROW,NLAY,IOUT,IBCFCB,LCWETD,IWDFLG,LCCVWD,
c     2  WETFCT,IWETIT,IHDWET,HDRY,IAPART,IFREFM,itrpy,lckgrad,ihold,
c     3  itrans,lcmid,lckmax,lcsgrad,lcsmax,lcsmid)
      SUBROUTINE BCF5AL(ISUM,LENX,LCSC1,LCHY,LCBOT,LCTOP,LCSC2,LCTRPY,
     1  IN,ISS,NCOL,NROW,NLAY,IOUT,IBCFCB,LCWETD,IWDFLG,LCCVWD,
     2  WETFCT,IWETIT,IHDWET,HDRY,IAPART,IFREFM,itrpy,lckgrad,ihold,
     3  itrans,lcmid,lckmax,lcsgrad,lcsmax,lcsmid,ikbase)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C-----VERSION 1431 20FEB1996 BCF5AL
C     ******************************************************************
C     ALLOCATE ARRAY STORAGE FOR BLOCK-CENTERED FLOW PACKAGE
C     ******************************************************************
C
C        SPECIFICATIONS:
C     ------------------------------------------------------------------
      COMMON /FLWCOM/LAYCON(200)
      COMMON /FLWAVG/LAYAVG(200)
C-Environment Agency (UK)-5.Inter-nodal transmissivities>>>>>>>>>>>>>>>>
c      CHARACTER*12 AVGNAM(4)
c      DATA AVGNAM/'HARMONIC    ','ARITHMETIC  ',
c     1            'LOGARITHMIC ','*UNCONFINED*'/
      CHARACTER*12 AVGNAM(5)
      DATA AVGNAM/'HARMONIC    ','ARITHMETIC  ',                           AT 001B
     1            'LOGARITHMIC ','*UNCONFINED*',
     2            'INTERNODAL  '/
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C     ------------------------------------------------------------------
C
C1------IDENTIFY PACKAGE
      WRITE(IOUT,1) IN                                                     AT 001B
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
c     1 FORMAT(1X,/1X,'BCF5 -- BLOCK-CENTERED FLOW PACKAGE, VERSION 5',
c      1', 9/1/93',' INPUT READ FROM UNIT',I3)
C-Environment Agency (UK)-10.K gradient independant of kbase>>>>>>>>>>>>
c     1 FORMAT(1X,/1X,'BCF-VKD1 -- BLOCK-CENTERED FLOW PACKAGE WITH ',
c      1'VARIABLE HYDRAULIC CONDUCTIVITY WITH DEPTH (VKD), VERSION 1',
c      1', 18/7/01',' INPUT READ FROM UNIT',I3)
    1 FORMAT(1X,/1X,'BCF-VKD3 -- BLOCK-CENTERED FLOW PACKAGE WITH ',       AT 0051
     1'VARIABLE HYDRAULIC CONDUCTIVITY WITH DEPTH (VKD), VERSION 5',
     1', 16/5/03',' INPUT READ FROM UNIT',I3)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C2------READ AND PRINT ISS (STEADY-STATE FLAG), IBCFCB (FLAG FOR
C2------PRINTING OR UNIT# FOR RECORDING CELL-BY-CELL FLOW TERMS), HDRY
```

```
C2------(HEAD AT CELLS THAT CONVERT TO DRY), WETTING PARAMETERS,
c2------anisotropy option, explicit transmissivity calculation option,
c2------and transmissivity output option.
c
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
c       IF(IFREFM.EQ.0) THEN
c          READ(IN,'(2I10,F10.0,I10,F10.0,2I10)')
c     1              ISS,IBCFCB,HDRY,IWDFLG,WETFCT,IWETIT,IHDWET
c       ELSE
c          READ(IN,*) ISS,IBCFCB,HDRY,IWDFLG,WETFCT,IWETIT,IHDWET
c       END IF
C-Environment Agency (UK)-10.K gradient independant of kbase>>>>>>>>>>>>
c       IF(IFREFM.EQ.0) THEN
c          READ(IN,'(2I10,F10.0,I10,F10.0,5I10)')
c     1              ISS,IBCFCB,HDRY,IWDFLG,WETFCT,IWETIT,IHDWET,itrpy
c     2             ,ihold,itrans
c       ELSE
c          READ(IN,*) ISS,IBCFCB,HDRY,IWDFLG,WETFCT,IWETIT,IHDWET,itrpy
c     1             ,ihold,itrans
c       END IF
      IF(IFREFM.EQ.0) THEN                                              AT 010C
         READ(IN,'(2I10,F10.0,I10,F10.0,6I10)')                         AT 0118
     1              ISS,IBCFCB,HDRY,IWDFLG,WETFCT,IWETIT,IHDWET,itrpy
     2             ,ihold,itrans,ikbase
      ELSE                                                              AT 021F
         READ(IN,*) ISS,IBCFCB,HDRY,IWDFLG,WETFCT,IWETIT,IHDWET,itrpy   AT 0224
     1             ,ihold,itrans,ikbase
      END IF                                                            AT 02EE
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      IF(ISS.EQ.0) WRITE(IOUT,3)                                        AT 02EE
    3 FORMAT(1X,'TRANSIENT SIMULATION')                                 AT 0320
      IF(ISS.NE.0) WRITE(IOUT,4)                                        AT 034B
    4 FORMAT(1X,'STEADY-STATE SIMULATION')                              AT 037D
      IF(IBCFCB.LT.0) WRITE(IOUT,8)                                     AT 03AC
    8 FORMAT(1X,'CONSTANT-HEAD CELL-BY-CELL FLOWS WILL BE PRINTED',     AT 03DE
     1     ' WHEN ICBCFL IS NOT 0')
      IF(IBCFCB.GT.0) WRITE(IOUT,9) IBCFCB                              AT 043F
    9 FORMAT(1X,'CELL-BY-CELL FLOWS WILL BE SAVED ON UNIT',I3)          AT 0481
      WRITE(IOUT,11) HDRY                                               AT 04C6
   11 FORMAT(1X,'HEAD AT CELLS THAT CONVERT TO DRY=',G13.5)             AT 04FC
      IF(IWDFLG.NE.0) GO TO 35                                          AT 053D
      WRITE(IOUT,12)                                                    AT 0549
   12 FORMAT(1X,'WETTING CAPABILITY IS NOT ACTIVE')                     AT 056F
      GO TO 39                                                          AT 05A6
C
   35 WRITE(IOUT,36)                                                    AT 05AB
   36 FORMAT(1X,'WETTING CAPABILITY IS ACTIVE')                         AT 05D1
      IF(IWETIT.LE.0) IWETIT=1                                          AT 0604
      WRITE(IOUT,37)WETFCT,IWETIT                                       AT 0616
   37 FORMAT(1X,'WETTING FACTOR=',F10.5,                                AT 065C
     1     '     WETTING ITERATION INTERVAL=',I4)
      WRITE(IOUT,38)IHDWET                                              AT 06B5
   38 FORMAT(1X,'FLAG THAT SPECIFIES THE EQUATION TO USE FOR HEAD',     AT 06EB
     1    ' AT WETTED CELLS=',I4)
C
C3------STOP THE SIMULATION IF THERE ARE MORE THAN 200 LAYERS.
   39 IF(NLAY.LE.200) GO TO 50                                          AT 074E
      WRITE(IOUT,41)                                                    AT 075D
   41 FORMAT(1X,/1X,'YOU HAVE SPECIFIED MORE THAN 200 MODEL LAYERS'/1X, AT 0783
     1 'SPACE IS RESERVED FOR A MAXIMUM OF 200 LAYERS IN ARRAYS LAYCON',
     2 ' AND LAYAVG')
      STOP                                                              AT 082A
C
C4------READ LAYCON & PRINT TITLE FOR LAYCON TABLE.
C-Environment Agency (UK)-10.K gradient independant of kbase>>>>>>>>>>>>
c   50 IF(IFREFM.EQ.0) THEN
   50 if(ikbase.ne.0.and.(iss.eq.2.or.iss.eq.3))then                    AT 083D
        write(iout,51)                                                  AT 085E
   51 format(1X,/1X,'YOU HAVE SPECIFIED VKGRAD TO BE INDEPENDANT OF'/1X, AT 0884
     1 'K BASE, BUT USING THE AUTOMATIC CONVERSION OPTION. THIS'/1X,
     2 'COMBINATION IS NOT SUPPORTED. VKGRAD WILL BE A MULTIPLE OF'/1X,
     3 'K BASE')
        ikbase=0                                                        AT 096D
      endif                                                             AT 0976
      IF(IFREFM.EQ.0) THEN                                              AT 0976
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
         READ(IN,'(40I2)') (LAYCON(I),I=1,NLAY)                         AT 0982
      ELSE                                                              AT 0A01
         READ(IN,*) (LAYCON(I),I=1,NLAY)                                AT 0A06
      END IF                                                            AT 0A64
      WRITE(IOUT,52)                                                    AT 0A64
```

```
   52 FORMAT(1X,5X,'LAYER  LAYER-TYPE CODE     INTERBLOCK T',            AT 0A8A
     1      /1X,5X,44('-'))
C
C5------LOOP THROUGH LAYERS CALCULATING LAYAVG, PRINTING THE LAYER-TYPE
C5------CODE, AND COUNTING LAYERS THAT NEED TOP & BOT ARRAYS.
      NBOT=0                                                             AT 0AE9
      NTOP=0                                                             AT 0AF3
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
      nkgrad=0                                                           AT 0AFD
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      DO 100 I=1,NLAY                                                    AT 0B07
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
c      IF(LAYCON(I).EQ.30 .OR. LAYCON(I).EQ.32) LAYCON(I)=LAYCON(I)-10
      IF(LAYCON(I).EQ.30 .OR. LAYCON(I).EQ.32 .or. laycon(i).eq.34 .or.  AT 0B34
     1      laycon(i).eq.35) LAYCON(I)=LAYCON(I)-10
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      INAM=LAYCON(I)/10                                                  AT 0B7A
      LAYAVG(I)=INAM*10                                                  AT 0B94
C-Environment Agency (UK)-5.Inter-nodal transmissivities>>>>>>>>>>>>>>>>
c      IF(LAYAVG(I).LT.0 .OR. LAYAVG(I).GT.30) THEN
      IF(LAYAVG(I).LT.0 .OR. LAYAVG(I).GT.40) THEN                       AT 0BA1
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
         WRITE(IOUT,53) LAYAVG(I)                                        AT 0BBD
   53    FORMAT(1X,'INVALID INTERBLOCK T CODE:',I4)                      AT 0BFE
         STOP                                                            AT 0C35
      END IF                                                             AT 0C48
      LAYCON(I)=LAYCON(I)-LAYAVG(I)                                      AT 0C48
      L=LAYCON(I)                                                        AT 0C5C
      INAM=INAM+1                                                        AT 0C69
      WRITE(IOUT,55) I,L,LAYAVG(I),AVGNAM(INAM)                          AT 0C6F
   55 FORMAT(1X,I9,I13,I11,' -- ',A)                                     AT 0D05
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
c      IF(LAYCON(I).LT.0 .OR. LAYCON(I).GT.3) THEN
      IF(LAYCON(I).LT.0 .OR. LAYCON(I).GT.5) THEN                        AT 0D38
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
         WRITE(IOUT,56) LAYCON(I)                                        AT 0D5A
   56    FORMAT(1X,'INVALID LAYER TYPE:',I4)                             AT 0D9B
         STOP                                                            AT 0DCC
      END IF                                                             AT 0DDF
C
C5A-----ONLY THE TOP LAYER CAN BE UNCONFINED(LAYCON=1).
      IF(L.NE.1 .OR. I.EQ.1) GO TO 70                                    AT 0DDF
      WRITE(IOUT,57)                                                     AT 0DF9
   57 FORMAT(1X,/1X,'LAYER TYPE 1 IS ONLY ALLOWED IN TOP LAYER')         AT 0E1F
      STOP                                                               AT 0E68
C
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
c   70 IF(L.EQ.1 .OR. L.EQ.3) NBOT=NBOT+1
c      IF(L.EQ.2 .OR. L.EQ.3) NTOP=NTOP+1
c
c5a2----only the top layer can have unconfined variable hydraulic
c5a2----conductivity with depth (VKD) - (LAYCON=4).
   70 if(l.ne.4 .or. i.eq.1) go to 72                                    AT 0E7B
      write(iout,58)                                                     AT 0E95
   58 format(1x,/1x,'LAYER TYPE 4 IS ONLY ALLOWED IN TOP LAYER')         AT 0EBB
      stop                                                               AT 0F04
c
C5B-----LAYER TYPES 1, 3, 4 and 5 NEED A BOTTOM. ADD 1 TO KB.
   72 IF(L.EQ.1 .OR. L.EQ.3 .or. l.eq.4 .or. l.eq.5) NBOT=NBOT+1         AT 0F17
C
C5C-----LAYER TYPES 2, 3 and 5 NEED A TOP. ADD 1 TO KT.
      IF(L.EQ.2 .OR. L.EQ.3 .or. l.eq.5) NTOP=NTOP+1                     AT 0F51
c
c5c2----layer types 4 and 5 need a hydraulic conductivity gradient,
c5c2----middle elevation and maximum hyd cond value. Add 1 to KG
      if(l.eq.4 .or. l.eq.5) nkgrad=nkgrad+1                             AT 0F7E
c
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C5D-----IF LAYAVG=30, BUFF MUST BE SEPARATE FROM RHS (IAPART NOT 0).
      IF(IAPART.EQ.0 .AND. LAYAVG(I).EQ.30) THEN                         AT 0F9E
         WRITE(IOUT,75)                                                  AT 0FBE
   75    FORMAT(1X,'IAPART IN BAS PACKAGE MUST BE NONZERO',              AT 0FE4
     1       ' WHEN INTERBLOCK T IS *UNCONFINED*')
         STOP                                                            AT 1047
      END IF                                                             AT 105A
  100 CONTINUE                                                           AT 105A
C
C
C6------COMPUTE THE NUMBER OF CELLS IN THE ENTIRE GRID AND IN ONE LAYER.
      NRC=NROW*NCOL                                                      AT 105F
      ISIZ=NRC*NLAY                                                      AT 1070
```

```
C
C7------ALLOCATE SPACE FOR ARRAYS.
      ISOLD=ISUM                                                          AT 107C
      LCSC1=ISUM                                                          AT 108A
      IF(ISS.EQ.0) ISUM=ISUM+ISIZ                                         AT 1092
      LCSC2=ISUM                                                          AT 10AC
      IF(ISS.EQ.0) ISUM=ISUM+NRC*NTOP                                     AT 10B9
      LCTRPY=ISUM                                                         AT 10D4
C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
C      ISUM=ISUM+NLAY
      if(itrpy.ne.2)then                                                  AT 10E1
        ISUM=ISUM+isiz                                                    AT 10ED
      else                                                                AT 10F5
        isum=isum+nrc*nbot                                                AT 10FA
      endif                                                               AT 110F
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      LCBOT=ISUM                                                          AT 110F
      ISUM=ISUM+NRC*NBOT                                                  AT 111F
      LCHY=ISUM                                                           AT 112E
      ISUM=ISUM+NRC*NBOT                                                  AT 1138
      LCTOP=ISUM                                                          AT 1147
      ISUM=ISUM+NRC*NTOP                                                  AT 1151
      LCWETD=ISUM                                                         AT 1160
      IF(IWDFLG.NE.0)ISUM=ISUM+NRC*NBOT                                   AT 1167
      LCCVWD=ISUM                                                         AT 1188
      IF(IWDFLG.NE.0)ISUM=ISUM+NRC*(NLAY-1)                               AT 1195
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
      lckgrad=isum                                                        AT 11B0
      isum=isum+nrc*nkgrad                                                AT 11BD
      lcmid=isum                                                          AT 11CC
      isum=isum+nrc*nkgrad                                                AT 11D3
      lckmax=isum                                                         AT 11E2
      isum=isum+nrc*nkgrad                                                AT 11E9
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-Environment Agency (UK)-2.Variable storage with depth (VSD)>>>>>>>>>>>
      lcsmid=isum                                                         AT 11F8
      if(iss.eq.0) isum=isum+nrc*nkgrad                                   AT 11FF
      lcsgrad=isum                                                        AT 1220
      if(iss.eq.0) isum=isum+nrc*nkgrad                                   AT 122D
      lcsmax=isum                                                         AT 1248
      if(iss.eq.0) isum=isum+nrc*nkgrad                                   AT 1255
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C8------PRINT THE AMOUNT OF SPACE USED BY THE BCF PACKAGE.
      ISP=ISUM-ISOLD                                                      AT 1270
      WRITE(IOUT,101) ISP                                                 AT 1284
  101 FORMAT(1X,I10,' ELEMENTS IN X ARRAY ARE USED BY BCF')               AT 12BE
      ISUM1=ISUM-1                                                        AT 12FF
      WRITE(IOUT,102) ISUM1,LENX                                          AT 130E
  102 FORMAT(1X,I10,' ELEMENTS OF X ARRAY USED OUT OF ',I10)              AT 135B
      IF(ISUM1.GT.LENX) WRITE(IOUT,103)                                   AT 13A0
  103 FORMAT(1X,'   ***X ARRAY MUST BE DIMENSIONED LARGER***')            AT 13DA
C
C9------RETURN.
      RETURN                                                              AT 141D
      END                                                                 AT 1422
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
c      SUBROUTINE BCF5RP(IBOUND,HNEW,SC1,HY,CR,CC,CV,DELR,DELC,BOT,TOP,
c     1 SC2,TRPY,IN,ISS,NCOL,NROW,NLAY,IOUT,WETDRY,IWDFLG,CVWD)
C-Environment Agency (UK)-10.K gradient independant of kbase>>>>>>>>>>>>
c      SUBROUTINE BCF5RP(IBOUND,HNEW,SC1,HY,CR,CC,CV,DELR,DELC,BOT,TOP,
c     1 SC2,TRPY,IN,ISS,NCOL,NROW,NLAY,IOUT,WETDRY,IWDFLG,CVWD,itrpy,
c     2 vkgrad,vmid,vkmax,vsgrad,vsmax,vsmid)


      SUBROUTINE BCF5RP(IBOUND,HNEW,SC1,HY,CR,CC,CV,DELR,DELC,BOT,TOP,
     1 SC2,TRPY,IN,ISS,NCOL,NROW,NLAY,IOUT,WETDRY,IWDFLG,CVWD,itrpy,
     2 vkgrad,vmid,vkmax,vsgrad,vsmax,vsmid,ikbase)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C-----VERSION 0917 17JULY1992 BCF5RP
C     ******************************************************************
C     READ AND INITIALIZE DATA FOR BLOCK-CENTERED FLOW PACKAGE
C     ******************************************************************
C
C        SPECIFICATIONS:
C     ------------------------------------------------------------------
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
c      CHARACTER*24 ANAME(11)
      CHARACTER*24 ANAME(21)
```

```
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      DOUBLE PRECISION HNEW
C
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
c       DIMENSION HNEW(NCOL,NROW,NLAY),SC1(NCOL,NROW,NLAY),
c      1    HY(NCOL,NROW,NLAY),CR(NCOL,NROW,NLAY),CC(NCOL,NROW,NLAY),
c      2    CV(NCOL,NROW,NLAY),DELR(NCOL),DELC(NROW),BOT(NCOL,NROW,NLAY),
c      3    TOP(NCOL,NROW,NLAY),SC2(NCOL,NROW,NLAY),TRPY(NLAY),
c      4    IBOUND(NCOL,NROW,NLAY),WETDRY(NCOL,NROW,NLAY),
c      5    CVWD(NCOL,NROW,NLAY)
      DIMENSION HNEW(NCOL,NROW,NLAY),SC1(NCOL,NROW,NLAY),
     1    HY(NCOL,NROW,NLAY),CR(NCOL,NROW,NLAY),CC(NCOL,NROW,NLAY),
     2    CV(NCOL,NROW,NLAY),DELR(NCOL),DELC(NROW),BOT(NCOL,NROW,NLAY),
     3    TOP(NCOL,NROW,NLAY),SC2(NCOL,NROW,NLAY),TRPY(ncol,nrow,nlay),
     4    IBOUND(NCOL,NROW,NLAY),WETDRY(NCOL,NROW,NLAY),
     5    CVWD(NCOL,NROW,NLAY),vkgrad(ncol,nrow,nlay),
     6    vmid(ncol,nrow,nlay),vkmax(ncol,nrow,nlay),
     7    vsgrad(ncol,nrow,nlay),vsmax(ncol,nrow,nlay),
     8    vsmid(ncol,nrow,nlay)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
      COMMON /FLWCOM/LAYCON(200)
C
      DATA ANAME(1) /'    PRIMARY STORAGE COEF'/                         AT 0085
      DATA ANAME(2) /'    TRANSMIS. ALONG ROWS'/                         AT 0085
      DATA ANAME(3) /'   HYD. COND. ALONG ROWS'/                         AT 0085
      DATA ANAME(4) /'VERT HYD COND /THICKNESS'/                         AT 0085
      DATA ANAME(5) /'                  BOTTOM'/                         AT 0085
      DATA ANAME(6) /'                     TOP'/                         AT 0085
      DATA ANAME(7) /'  SECONDARY STORAGE COEF'/                         AT 0085
      DATA ANAME(8) /'COLUMN TO ROW ANISOTROPY'/                         AT 0085
      DATA ANAME(9) /'                    DELR'/                         AT 0085
      DATA ANAME(10)/'                    DELC'/                         AT 0085
      DATA ANAME(11)/'        WETDRY PARAMETER'/                         AT 0085
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
      data aname(12)/'HYD COND GRADIENT FACTOR'/                         AT 0085
      data aname(13)/' THICKNESS OF UPPER ZONE'/                         AT 0085
      DATA ANAME(14)/'                  BOTTOM'/                         AT 0085
      data aname(15)/' TRANSMIS. ALONG COLUMNS'/                         AT 0085
      data aname(16)/'HYD. COND. ALONG COLUMNS'/                         AT 0085
      data aname(17)/'ELEVATION OF CHANGE IN K'/                         AT 0085
      data aname(18)/' MAXIMUM HYD COND FACTOR'/                         AT 0085
      data aname(19)/' STORAGE GRADIENT FACTOR'/                         AT 0085
      data aname(20)/'  MAXIMUM STORAGE FACTOR'/                         AT 0085
      data aname(21)/'ELEVATION OF CHANGE IN S'/                         AT 0085
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C     ------------------------------------------------------------------
C
C1------READ TRPY,DELR,DELC.
C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
c       CALL U1DREL(TRPY,ANAME(8),NLAY,IN,IOUT)
      if(itrpy.eq.0)then                                                 AT 0085
        loc=1                                                            AT 0091
        call u2drel(trpy(1,1,1),aname(8),loc,nlay,loc,in,iout)           AT 009B
      elseif(itrpy.ne.2)then                                             AT 00E9
        do 23 k=1,nlay                                                   AT 00FA
          call u2drel(trpy(1,1,k),aname(8),nrow,ncol,k,in,iout)          AT 0127
 23     continue                                                         AT 0175
      endif                                                              AT 0177
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      CALL U1DREL(DELR,ANAME(9),NCOL,IN,IOUT)                            AT 0177
      CALL U1DREL(DELC,ANAME(10),NROW,IN,IOUT)                           AT 0198
C
C2------READ ALL PARAMETERS FOR EACH LAYER.
      KT=0                                                               AT 01B9
      KB=0                                                               AT 01C3
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
      kg=0                                                               AT 01CD
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      DO 200 K=1,NLAY                                                    AT 01D7
      KK=K                                                               AT 0204
C
C2A-----FIND ADDRESS OF EACH LAYER IN THREE DIMENSION ARRAYS.
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
C       IF(LAYCON(K).EQ.1 .OR. LAYCON(K).EQ.3) KB=KB+1
C       IF(LAYCON(K).EQ.2 .OR. LAYCON(K).EQ.3) KT=KT+1
      IF(LAYCON(K).EQ.1 .OR. LAYCON(K).EQ.3 .or. laycon(k).eq.4 .or.     AT 020A
     1      laycon(k).eq.5) KB=KB+1
      IF(LAYCON(K).EQ.2 .OR. LAYCON(K).EQ.3 .or. laycon(k).eq.5) KT=KT+1 AT 0248
      if(laycon(k).eq.4 .or. laycon(k).eq.5) kg=kg+1                     AT 027E
C
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
```

```
C-Environment Agency (UK)-3.Auto-conversion for steady state VKD>>>>>>>>
      if(iss.eq.2.and.(laycon(k).eq.4.or.laycon(k).eq.5))then              AT 02A6
        aname(3)=aname(2)                                                  AT 02D4
        aname(5)=aname(14)                                                 AT 02E7
        aname(17)=aname(13)                                                AT 02FA
        aname(16)=aname(15)                                                AT 030D
      elseif(iss.eq.3.and.(laycon(k).eq.4.or.laycon(k).eq.5))then          AT 0320
        aname(17)=aname(13)                                                AT 0353
      else                                                                 AT 0366
        ANAME(3) ='   HYD. COND. ALONG ROWS'                               AT 036B
        ANAME(5) ='                 BOTTOM'                                AT 037E
        aname(17)='ELEVATION OF CHANGE IN K'                               AT 0391
        aname(16)='HYD. COND. ALONG COLUMNS'                               AT 03A4
      endif                                                                AT 03B7
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C2B-----READ PRIMARY STORAGE COEFFICIENT INTO ARRAY SC1 IF TRANSIENT.
      IF(ISS.EQ.0)CALL U2DREL(SC1(1,1,K),ANAME(1),NROW,NCOL,KK,IN,IOUT)    AT 03B7
C
C2C-----READ TRANSMISSIVITY INTO ARRAY CC IF LAYER TYPE IS 0 OR 2.
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
c      IF(LAYCON(K).EQ.3 .OR. LAYCON(K).EQ.1) GO TO 100
      IF(LAYCON(K).EQ.3 .OR. LAYCON(K).EQ.1 .or. laycon(k).eq.4 .or.       AT 0415
     1      laycon(k).eq.5) GO TO 100
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      CALL U2DREL(CC(1,1,K),ANAME(2),NROW,NCOL,KK,IN,IOUT)                 AT 0453
C-Environment Agency (UK)-6.X & Y-dir transmissivities>>>>>>>>>>>>>>>>>>
      if(itrpy.ne.2)goto 110                                               AT 04A1
c2c2----Move X-dir transmissivity to array CR and read Y-dir trans into
c2c2----array CC if trans is specified in each direction (ITRPY=2).
      do 14 i=1,nrow                                                       AT 04AD
      do 14 j=1,ncol                                                       AT 04DA
         cr(j,i,k)=cc(j,i,k)                                               AT 0507
 14   continue                                                             AT 053F
      call u2drel(cc(1,1,k),aname(15),nrow,ncol,kk,in,iout)                AT 0546
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      GO TO 110                                                            AT 0598
C
C2D-----READ HYDRAULIC CONDUCTIVITY(HY) AND BOTTOM ELEVATION(BOT)
C2D-----IF LAYER TYPE IS 1, 3, 4 or 5.
  100 CALL U2DREL(HY(1,1,KB),ANAME(3),NROW,NCOL,KK,IN,IOUT)                AT 059D
C-Environment Agency (UK)-6.X & Y-dir transmissivities>>>>>>>>>>>>>>>>>>
c      CALL U2DREL(BOT(1,1,KB),ANAME(5),NROW,NCOL,KK,IN,IOUT)
      if(itrpy.ne.2)goto 105                                               AT 05EF
c2d2----Read Y-dir hyraulic conductivity into array TRPY if specified in
c2d2----each direction (ITRPY=2).
      call u2drel(trpy(1,1,kb),aname(16),nrow,ncol,kk,in,iout)             AT 05FB
 105  CALL U2DREL(BOT(1,1,KB),ANAME(5),NROW,NCOL,KK,IN,IOUT)               AT 064D
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C2E-----READ VERTICAL HYCOND/THICK INTO ARRAY CV IF NOT BOTTOM LAYER;
C2E-----MULTIPLIED BY CELL AREA TO CONVERT TO CONDUCTANCE LATER.
  110 IF(K.EQ.NLAY) GO TO 120                                              AT 069F
      CALL U2DREL(CV(1,1,K),ANAME(4),NROW,NCOL,KK,IN,IOUT)                 AT 06B0
C
C2F-----READ SECONDARY STORAGE COEFFICIENT INTO ARRAY SC2 IF TRANSIENT
C2F-----AND LAYER TYPE IS 2, 3 or 5.
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
c  120 IF(LAYCON(K).NE.3 .AND. LAYCON(K).NE.2) GO TO 130
  120 IF(LAYCON(K).NE.3.AND.LAYCON(K).NE.2.and.laycon(k).ne.5) GO TO 130   AT 06FE
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      IF(ISS.EQ.0)CALL U2DREL(SC2(1,1,KT),ANAME(7),NROW,NCOL,KK,IN,IOUT)   AT 072E
C
C2G-----READ TOP ELEVATION(TOP) IF LAYER TYPE IS 2, 3 or 5.
      CALL U2DREL(TOP(1,1,KT),ANAME(6),NROW,NCOL,KK,IN,IOUT)               AT 078C
C
C2H-----READ WETDRY CODES IF LAYER TYPE IS 1, 3, 4 or 5 AND WETTING
C2H-----CAPABILITY HAS BEEN INVOKED (IWDFLG NOT 0).
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
c  130 IF(LAYCON(K).NE.3.AND.LAYCON(K).NE.1)GO TO 200
c      IF(IWDFLG.EQ.0)GO TO 200
  130 IF(LAYCON(K).NE.3 .AND. LAYCON(K).NE.1 .and. laycon(k).ne.4 .and.    AT 07DE
     1      laycon(k).ne.5)GO TO 200
      IF(IWDFLG.EQ.0)GO TO 140                                             AT 081C
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      CALL U2DREL(WETDRY(1,1,KB),ANAME(11),NROW,NCOL,KK,IN,IOUT)           AT 0828
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
c2i-----Read VKD parameters if layer type is 4 or 5.
 140  if(laycon(k).ne.4 .and. laycon(k).ne.5) go to 200                    AT 087A
      call u2drel(vmid(1,1,kg),aname(17),nrow,ncol,kk,in,iout)             AT 089C
      call u2drel(vkgrad(1,1,kg),aname(12),nrow,ncol,kk,in,iout)           AT 08EE
      call u2drel(vkmax(1,1,kg),aname(18),nrow,ncol,kk,in,iout)            AT 0940
```

```
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-Environment Agency (UK)-10.K gradient independant of kbase>>>>>>>>>>>>
      if(ikbase.ne.0)then                                                AT 0992
        do i=1,nrow                                                      AT 099E
          do j=1,ncol                                                    AT 09CB
C-Environment Agency (UK)-12.Bug fix>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
c            vkgrad(j,i,kg)=vkgrad(j,i,kg)/hy(j,i,kb)
c            vkmax(j,i,kg)=vkmax(j,i,kg)/hy(j,i,kb)
            if(hy(j,i,k).eq.0)then                                       AT 09F8
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
              vkgrad(j,i,kg)=0                                           AT 0A37
              vkmax(j,i,kg)=0                                            AT 0A70
            else                                                         AT 0AA9
              vkgrad(j,i,kg)=vkgrad(j,i,kg)/hy(j,i,kb)                   AT 0AAE
              vkmax(j,i,kg)=vkmax(j,i,kg)/hy(j,i,kb)                     AT 0B0D
            endif                                                        AT 0B69
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
          enddo                                                          AT 0B69
        enddo                                                            AT 0B6E
      endif                                                              AT 0B73
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-Environment Agency (UK)-2.Variable storage with depth (VSD)>>>>>>>>>>>
c2j-----Read VSD parameters if transient and layer type is 4 or 5.
      if(iss.ne.0)goto 200                                               AT 0B73
      call u2drel(vsmid(1,1,kg),aname(21),nrow,ncol,kk,in,iout)          AT 0B7F
      call u2drel(vsgrad(1,1,kg),aname(19),nrow,ncol,kk,in,iout)         AT 0BD1
      call u2drel(vsmax(1,1,kg),aname(20),nrow,ncol,kk,in,iout)          AT 0C23
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-Environment Agency (UK)-10.K gradient independant of kbase>>>>>>>>>>>>
      if(ikbase.ne.0)then                                                AT 0C75
        if(laycon(k).eq.4)then                                           AT 0C81
          do i=1,nrow                                                    AT 0C95
            do j=1,ncol                                                  AT 0CC2
C-Environment Agency (UK)-12.Bug fix>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
c             vsgrad(j,i,kg)=vsgrad(j,i,kg)/sc1(j,i,k)
c             vsmax(j,i,kg)=vsmax(j,i,kg)/sc1(j,i,k)
              if(sc1(j,i,k).eq.0)then                                    AT 0CEF
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
                vsgrad(j,i,kg)=0                                         AT 0D2E
                vsmax(j,i,kg)=0                                          AT 0D67
              else                                                       AT 0DA0
                vsgrad(j,i,kg)=vsgrad(j,i,kg)/sc1(j,i,k)                 AT 0DA5
                vsmax(j,i,kg)=vsmax(j,i,kg)/sc1(j,i,k)                   AT 0E04
              endif                                                      AT 0E60
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
            enddo                                                        AT 0E60
          enddo                                                          AT 0E65
        else                                                             AT 0E6A
          do i=1,nrow                                                    AT 0E6F
            do j=1,ncol                                                  AT 0E9C
C-Environment Agency (UK)-12.Bug fix>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
c             vsgrad(j,i,kg)=vsgrad(j,i,kg)/sc2(j,i,kt)
c             vsmax(j,i,kg)=vsmax(j,i,kg)/sc2(j,i,kt)
              if(sc2(j,i,k).eq.0)then                                    AT 0EC9
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
                vsgrad(j,i,kg)=0                                         AT 0F08
                vsmax(j,i,kg)=0                                          AT 0F41
              else                                                       AT 0F7A
                vsgrad(j,i,kg)=vsgrad(j,i,kg)/sc2(j,i,kt)                AT 0F7F
                vsmax(j,i,kg)=vsmax(j,i,kg)/sc2(j,i,kt)                  AT 0FDE
              endif                                                      AT 103A
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
            enddo                                                        AT 103A
          enddo                                                          AT 103F
        endif                                                            AT 1044
      endif                                                              AT 1044
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
  200 CONTINUE                                                           AT 1044
C
C3------PREPARE AND CHECK BCF DATA.
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
c      CALL SBCF5N(HNEW,IBOUND,SC1,SC2,CR,CC,CV,HY,TRPY,DELR,DELC,ISS,
c     1         NCOL,NROW,NLAY,IOUT,WETDRY,IWDFLG,CVWD)
      CALL SBCF5N(HNEW,IBOUND,SC1,SC2,CR,CC,CV,HY,TRPY,DELR,DELC,ISS,    AT 1049
     1         NCOL,NROW,NLAY,IOUT,WETDRY,IWDFLG,CVWD,itrpy,vkgrad,
     2         vkmax,vsgrad,vsmax)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C4------RETURN
      RETURN                                                             AT 109F
      END                                                                AT 10A4
```

```
      SUBROUTINE BCF5AD(IBOUND,HOLD,BOT,WETDRY,IWDFLG,ISS,
     1                  NCOL,NROW,NLAY)
C
C-----VERSION 1659 30OCT1992 BCF5AD
C     ******************************************************************
C     SET HOLD TO BOT WHENEVER A WETTABLE CELL IS DRY
C     ******************************************************************
C
C        SPECIFICATIONS:
C     ------------------------------------------------------------------
C
      DIMENSION IBOUND(NCOL,NROW,NLAY),HOLD(NCOL,NROW,NLAY),
     1          BOT(NCOL,NROW,NLAY),WETDRY(NCOL,NROW,NLAY)
C
      COMMON /FLWCOM/LAYCON(200)
C     ------------------------------------------------------------------
C
C1------RETURN IF STEADY STATE OR IF NOT USING WETTING CAPABILITY
      IF(IWDFLG.EQ.0 .OR. ISS.NE.0) RETURN                              AT 004C
C
C2------LOOP THROUGH ALL LAYERS TO SET HOLD=BOT IF A WETTABLE CELL IS DRY
      ZERO=0.                                                           AT 0064
      KB=0                                                              AT 0072
      DO 100 K=1,NLAY                                                   AT 0079
C
C2A-----SKIP LAYERS THAT CANNOT CONVERT BETWEEN WET AND DRY
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
c      IF(LAYCON(K).NE.3 .AND. LAYCON(K).NE.1) GO TO 100
      IF(LAYCON(K).NE.3 .AND. LAYCON(K).NE.1 .and. laycon(k).ne.4 .and.  AT 0092
     1      laycon(k).ne.5) GO TO 100
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      KB=KB+1                                                           AT 00CA
      DO 90 I=1,NROW                                                    AT 00CD
      DO 90 J=1,NCOL                                                    AT 00EB
C
C2B-----SKIP CELLS THAT ARE CURRENTLY WET OR ARE NOT WETTABLE
      IF(IBOUND(J,I,K).NE.0) GO TO 90                                   AT 0109
      IF(WETDRY(J,I,KB).EQ.ZERO) GO TO 90                               AT 012D
C
C2C-----SET HOLD=BOT
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
      HOLD(J,I,K)=BOT(J,I,KB)                                           AT 0157
   90 CONTINUE                                                          AT 0195
  100 CONTINUE                                                          AT 019F
C
C3-----RETURN
      RETURN                                                            AT 01A4
      END                                                               AT 01A9
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
c      SUBROUTINE BCF5FM(HCOF,RHS,HOLD,SC1,HNEW,IBOUND,CR,CC,CV,HY,TRPY,
c     1                BOT,TOP,SC2,DELR,DELC,DELT,ISS,KITER,KSTP,KPER,
c     2                NCOL,NROW,NLAY,IOUT,WETDRY,IWDFLG,CVWD,
c     3                WETFCT,IWETIT,IHDWET,HDRY,BUFF)
```

```
      SUBROUTINE BCF5FM(HCOF,RHS,HOLD,SC1,HNEW,IBOUND,CR,CC,CV,HY,TRPY,
     1               BOT,TOP,SC2,DELR,DELC,DELT,ISS,KITER,KSTP,KPER,
     2               NCOL,NROW,NLAY,IOUT,WETDRY,IWDFLG,CVWD,
     3               WETFCT,IWETIT,IHDWET,HDRY,BUFF,itrpy,vkgrad,ihold,
     4               vmid,vkmax,vsgrad,vsmax,vsmid)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-----VERSION 1500 29JUNE1993 BCF5FM
C     ******************************************************************
C     ADD LEAKAGE CORRECTION AND STORAGE TO HCOF AND RHS, AND CALCULATE
C     CONDUCTANCE AS REQUIRED
C     ******************************************************************
C
C        SPECIFICATIONS:
C     ------------------------------------------------------------------
C-Environment Agency (UK)-2.Variable storage with depth (VSD)>>>>>>>>>>>
c      DOUBLE PRECISION HNEW
      DOUBLE PRECISION HNEW,htmp,vm,vsm,vsg,esm,bt,stmp,sold,snew,scof,
     1                rho1,rho2,tp
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
c      DIMENSION HCOF(NCOL,NROW,NLAY),RHS(NCOL,NROW,NLAY),
c     1    HOLD(NCOL,NROW,NLAY),SC1(NCOL,NROW,NLAY),HNEW(NCOL,NROW,NLAY),
c     2    IBOUND(NCOL,NROW,NLAY),CR(NCOL,NROW,NLAY),
c     3    CC(NCOL,NROW,NLAY),CV(NCOL,NROW,NLAY),HY(NCOL,NROW,NLAY),
```

```
c      4    TRPY(NLAY),BOT(NCOL,NROW,NLAY),TOP(NCOL,NROW,NLAY),DELR(NCOL),
c      5    DELC(NROW),SC2(NCOL,NROW,NLAY),WETDRY(NCOL,NROW,NLAY),
c      6    CVWD(NCOL,NROW,NLAY),BUFF(NCOL,NROW,NLAY)
      DIMENSION HCOF(NCOL,NROW,NLAY),RHS(NCOL,NROW,NLAY),
     1    HOLD(NCOL,NROW,NLAY),SC1(NCOL,NROW,NLAY),HNEW(NCOL,NROW,NLAY),
     2    IBOUND(NCOL,NROW,NLAY),CR(NCOL,NROW,NLAY),
     3    CC(NCOL,NROW,NLAY),CV(NCOL,NROW,NLAY),HY(NCOL,NROW,NLAY),
     4    TRPY(ncol,nrow,nlay),BOT(NCOL,NROW,NLAY),TOP(NCOL,NROW,NLAY),
     5    DELR(NCOL),DELC(NROW),SC2(NCOL,NROW,NLAY),
     6    WETDRY(NCOL,NROW,NLAY),CVWD(NCOL,NROW,NLAY),
     7    BUFF(NCOL,NROW,NLAY),vkgrad(ncol,nrow,nlay),
     8    vmid(ncol,nrow,nlay),vkmax(ncol,nrow,nlay)
      dimension vsgrad(ncol,nrow,nlay), vsmax(ncol,nrow,nlay),
     1    vsmid(ncol,nrow,nlay)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
      COMMON /FLWCOM/LAYCON(200)
C     ------------------------------------------------------------------
      KB=0                                                          AT 0085
      KT=0                                                          AT 008F
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
      kg=0                                                          AT 0099
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      ONE=1.                                                        AT 00A3
      TLED=ONE/DELT                                                 AT 00B7
C
C1------FOR EACH LAYER: IF T VARIES CALCULATE HORIZONTAL CONDUCTANCES
      DO 100 K=1,NLAY                                               AT 00C9
      KK=K                                                          AT 00F1
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
c      IF(LAYCON(K).EQ.3 .OR. LAYCON(K).EQ.2) KT=KT+1
      IF(LAYCON(K).EQ.3 .OR. LAYCON(K).EQ.2 .or. laycon(k).eq.5) KT=KT+1   AT 00F7
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C1A-----IF LAYER TYPE IS NOT 1, 3, 4 or 5 THEN SKIP THIS LAYER.
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
c      IF(LAYCON(K).NE.3 .AND. LAYCON(K).NE.1) GO TO 100
      IF(LAYCON(K).NE.3 .AND. LAYCON(K).NE.1 .and. laycon(k).ne.4 .and.    AT 0127
     1      laycon(k).ne.5) GO TO 100
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      KB=KB+1                                                       AT 0165
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
      if(laycon(k).eq.4 .or. laycon(k).eq.5) kg=kg+1                AT 016B
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C1B-----FOR LAYER TYPES 1, 3, 4 & 5 CALL SBCF5H TO CALCULATE
C1B-----HORIZONTAL CONDUCTANCES.
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
c      CALL SBCF5H(HNEW,IBOUND,CR,CC,CV,HY,TRPY,DELR,DELC,BOT,TOP,
c      1   KK,KB,KT,KITER,KSTP,KPER,NCOL,NROW,NLAY,IOUT,WETDRY,IWDFLG,
c      2   CVWD,WETFCT,IWETIT,IHDWET,HDRY,BUFF)
      CALL SBCF5H(HNEW,IBOUND,CR,CC,CV,HY,TRPY,DELR,DELC,BOT,TOP,   AT 0193
     1  KK,KB,KT,KITER,KSTP,KPER,NCOL,NROW,NLAY,IOUT,WETDRY,IWDFLG,
     2  CVWD,WETFCT,IWETIT,IHDWET,HDRY,BUFF,itrpy,vkgrad,kg,hold,ihold,
     3  iss,vmid,vkmax)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
  100 CONTINUE                                                      AT 023B
C
C2------IF THE SIMULATION IS TRANSIENT ADD STORAGE TO HCOF AND RHS
      IF(ISS.NE.0) GO TO 201                                        AT 0240
      KT=0                                                          AT 024C
C-Environment Agency (UK)-2.Variable storage with depth (VSD)>>>>>>>>>>>
      kb=0                                                          AT 0256
      kg=0                                                          AT 0260
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      DO 200 K=1,NLAY                                               AT 026A
C
C3------SEE IF THIS LAYER IS CONVERTIBLE OR NON-CONVERTIBLE.
      IF(LAYCON(K).EQ.3 .OR. LAYCON(K).EQ.2) GO TO 150              AT 0297
c
C-Environment Agency (UK)-2.Variable storage with depth (VSD)>>>>>>>>>>>
c3a-----See if This Layer has Varying Storage with Depth (VSD)
      if(laycon(k).eq.4.or.laycon(k).eq.5)goto 400                  AT 02B3
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C4------NON-CONVERTIBLE LAYER, SO USE PRIMARY STORAGE
      DO 140 I=1,NROW                                               AT 02CF
      DO 140 J=1,NCOL                                               AT 02FC
      IF(IBOUND(J,I,K).LE.0) GO TO 140                              AT 0329
      RHO=SC1(J,I,K)*TLED                                           AT 0362
      HCOF(J,I,K)=HCOF(J,I,K)-RHO                                   AT 039B
      RHS(J,I,K)=RHS(J,I,K)-RHO*HOLD(J,I,K)                         AT 03CE
  140 CONTINUE                                                      AT 0411
```

```
      GO TO 200                                                       AT 041B
C
C5------A CONVERTIBLE LAYER, SO CHECK OLD AND NEW HEADS TO DETERMINE
C5------WHEN TO USE PRIMARY AND SECONDARY STORAGE
  150 KT=KT+1                                                         AT 0420
      DO 180 I=1,NROW                                                 AT 0426
      DO 180 J=1,NCOL                                                 AT 0453
C
C5A-----IF THE CELL IS EXTERNAL THEN SKIP IT.
      IF(IBOUND(J,I,K).LE.0) GO TO 180                                AT 0480
      TP=TOP(J,I,KT)                                                  AT 04B9
      RHO2=SC2(J,I,KT)*TLED                                           AT 04F5
      RHO1=SC1(J,I,K)*TLED                                            AT 052F
C
C5B-----FIND STORAGE FACTOR AT START OF TIME STEP.
      SOLD=RHO2                                                       AT 056C
      IF(HOLD(J,I,K).GT.TP) SOLD=RHO1                                 AT 0584
C
C5C-----FIND STORAGE FACTOR AT END OF TIME STEP.
      HTMP=HNEW(J,I,K)                                                AT 05E2
      SNEW=RHO2                                                       AT 0622
      IF(HTMP.GT.TP) SNEW=RHO1                                        AT 063A
C
C5D-----ADD STORAGE TERMS TO RHS AND HCOF.
      HCOF(J,I,K)=HCOF(J,I,K)-SNEW                                    AT 0668
      RHS(J,I,K)=RHS(J,I,K) - SOLD*(HOLD(J,I,K)-TP) - SNEW*TP         AT 06A3
C
  180 CONTINUE                                                        AT 0700
C-Environment Agency (UK)-2.Variable storage with depth (VSD)>>>>>>>>>>>
      goto 200                                                        AT 070A
c
c5.5----Varying Storage with Depth (VSD), so calculate storage change
  400 kg=kg+1                                                         AT 070F
      kb=kb+1                                                         AT 0715
      if(laycon(k).eq.5)kt=kt+1                                       AT 071B
      do 410 i=1,nrow                                                 AT 0735
      do 410 j=1,ncol                                                 AT 0762
c
c5.5a---If The Cell Is Extenal Then Skip It.
      if(ibound(j,i,k).le.0)goto 410                                  AT 078F
      vm=vsmid(j,i,kg)                                                AT 07C8
      vsm=vsmax(j,i,kg)                                               AT 0801
      vsg=vsgrad(j,i,kg)                                              AT 083E
      esm=vm+((vsm-1.0)/vsg)                                          AT 087B
      bt=bot(j,i,kb)                                                  AT 089A
      rho1=sc1(j,i,k)*tled                                            AT 08DA
      rho2=sc1(j,i,k)*tled                                            AT 0917
      tp=1.0E+10                                                      AT 094E
      if(laycon(k).eq.5)then                                          AT 0962
        rho2=sc2(j,i,kt)*tled                                         AT 0976
        tp=top(j,i,kt)                                                AT 09B0
      endif                                                           AT 09F0
c
c5.5b---Find Storage At Start Of Time Step
      htmp=hold(j,i,k)                                                AT 09F0
      itmp=1                                                          AT 0A30
 420  if(htmp.le.vm)then                                              AT 0A3A
C-Environment Agency (UK)-9.Bug fix>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
c        stmp = rho2*(htmp-bt)
        if((laycon(k).eq.5.and.htmp.le.tp).or.laycon(k).eq.4)then     AT 0A50
          stmp = rho2*(htmp-bt)                                       AT 0A8E
        else                                                          AT 0AA7
          stmp = rho2*(tp-bt) + rho1*(htmp-tp)                        AT 0AAC
        endif                                                         AT 0AD9
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      elseif((laycon(k).eq.5.and.htmp.le.tp).or.laycon(k).eq.4)then   AT 0AD9
        if(htmp.le.esm)then                                           AT 0B1C
          stmp = rho2*(htmp-bt) + (rho2*vsg*(htmp-vm)**2)/2           AT 0B32
        else                                                          AT 0B6F
          stmp = rho2*(esm-bt) + (rho2*vsg*(esm-vm)**2)/2 +           AT 0B74
     1           (htmp-esm)*vsm*rho2
        endif                                                         AT 0BCB
      else                                                            AT 0BCB
        if(tp.le.esm)then                                             AT 0BD0
          stmp = rho2*(tp-bt) + (rho2*vsg*(tp-vm)**2)/2 + rho1*(htmp-tp) AT 0BE6
        else                                                          AT 0C37
          stmp = rho2*(esm-bt) + (rho2*vsg*(esm-vm)**2)/2 +           AT 0C3C
     1           (tp-esm)*vsm*rho2 + rho1*(htmp-tp)
        endif                                                         AT 0CA7
      endif                                                           AT 0CA7
c
c5.5c---Find Storage At End Of Time Step
```

```
      if(itmp.eq.1)then                                               AT 0CA7
        sold=stmp                                                     AT 0CB4
        htmp=hnew(j,i,k)                                              AT 0CCC
        itmp=0                                                        AT 0D0C
        goto 420                                                      AT 0D16
      endif                                                           AT 0D1B
      snew=stmp                                                       AT 0D1B
c
c5.5d---Find Storage Factor At End Of Time Step
      if(htmp.le.vm)then                                              AT 0D33
C-Environment Agency (UK)-9.Bug fix>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
c         scof = rho2
        if((laycon(k).eq.5.and.htmp.le.tp).or.laycon(k).eq.4)then     AT 0D49
          scof = rho2                                                 AT 0D87
        else                                                          AT 0D9F
          scof = rho1                                                 AT 0DA4
        endif                                                         AT 0DBC
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      elseif((laycon(k).eq.5.and.htmp.le.tp).or.laycon(k).eq.4)then   AT 0DBC
        if(htmp.le.esm)then                                           AT 0DFF
          scof = rho2*vsg*(htmp-vm)                                   AT 0E15
        else                                                          AT 0E34
          scof = vsm*rho2                                             AT 0E39
        endif                                                         AT 0E4C
      else                                                            AT 0E4C
        scof = rho1                                                   AT 0E51
      endif                                                           AT 0E69
c
c5.5e---Add Storage Terms To RHS and HCOF.
      rhs(j,i,k)=rhs(j,i,k) + snew - sold - (scof*htmp)               AT 0E69
      hcof(j,i,k)=hcof(j,i,k) - scof                                  AT 0EB8
c
  410 continue                                                        AT 0EF3
c
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
  200 CONTINUE                                                        AT 0EFD
C
C6------FOR EACH LAYER DETERMINE IF CORRECTION TERMS ARE NEEDED FOR
C6------FLOW DOWN INTO PARTIALLY SATURATED LAYERS.
  201 KT=0                                                            AT 0F02
      DO 300 K=1,NLAY                                                 AT 0F0C
C
C7------SEE IF CORRECTION IS NEEDED FOR LEAKAGE FROM ABOVE.
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
c      IF(LAYCON(K).NE.3 .AND. LAYCON(K).NE.2) GO TO 250
      IF(LAYCON(K).NE.3.AND.LAYCON(K).NE.2.and.laycon(k).ne.5) GO TO 250   AT 0F39
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      KT=KT+1                                                         AT 0F63
      IF(K.EQ.1) GO TO 250                                            AT 0F69
C
C7A-----FOR EACH CELL MAKE THE CORRECTION IF NEEDED.
      DO 220 I=1,NROW                                                 AT 0F76
      DO 220 J=1,NCOL                                                 AT 0FA3
C
C7B-----IF THE CELL IS EXTERNAL(IBOUND<=0) THEN SKIP IT.
      IF(IBOUND(J,I,K).LE.0) GO TO 220                                AT 0FD0
      HTMP=HNEW(J,I,K)                                                AT 1009
C
C7C-----IF HEAD IS ABOVE TOP THEN CORRECTION NOT NEEDED
      IF(HTMP.GE.TOP(J,I,KT)) GO TO 220                               AT 1045
C
C7D-----WITH HEAD BELOW TOP ADD CORRECTION TERMS TO RHS.
      RHS(J,I,K)=RHS(J,I,K) + CV(J,I,K-1)*(TOP(J,I,KT)-HTMP)          AT 108B
  220 CONTINUE                                                        AT 1121
C
C8------SEE IF THIS LAYER MAY NEED CORRECTION FOR LEAKAGE TO BELOW.
  250 IF(K.EQ.NLAY) GO TO 300                                         AT 112B
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
c      IF(LAYCON(K+1).NE.3 .AND. LAYCON(K+1).NE.2) GO TO 300
      IF(LAYCON(K+1).NE.3 .AND. LAYCON(K+1).NE.2 .and. laycon(k+1).ne.5)   AT 113C
     1      GO TO 300
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      KTT=KT+1                                                        AT 1166
C
C8A-----FOR EACH CELL MAKE THE CORRECTION IF NEEDED.
      DO 280 I=1,NROW                                                 AT 1173
      DO 280 J=1,NCOL                                                 AT 11A0
C
C8B-----IF CELL IS EXTERNAL (IBOUND<=0) THEN SKIP IT.
      IF(IBOUND(J,I,K).LE.0) GO TO 280                                AT 11CD
C
```

```
C8C-----IF HEAD IN THE LOWER CELL IS LESS THAN TOP ADD CORRECTION
C8C-----TERM TO RHS.
      HTMP=HNEW(J,I,K+1)                                              AT 1206
      IF(HTMP.LT.TOP(J,I,KTT)) RHS(J,I,K)=RHS(J,I,K)                  AT 1243
     1                        - CV(J,I,K)*(TOP(J,I,KTT)-HTMP)
  280 CONTINUE                                                        AT 12FA
  300 CONTINUE                                                        AT 1304
C
C9------RETURN
      RETURN                                                          AT 1309
      END                                                             AT 130E
C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
c      SUBROUTINE SBCF5C(CR,CC,TRPY,DELR,DELC,K,NCOL,NROW,NLAY)


      SUBROUTINE SBCF5C(CR,CC,TRPY,DELR,DELC,K,NCOL,NROW,NLAY,itrpy)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C
C-----VERSION 1512 02JULY1993 SBCF5C
C     ******************************************************************
C     COMPUTE BRANCH CONDUCTANCE USING HARMONIC MEAN OF BLOCK
C     CONDUCTANCES -- BLOCK TRANSMISSIVITY IS IN CC UPON ENTRY
C     ******************************************************************
C
C      SPECIFICATIONS:
C     ------------------------------------------------------------------
C
C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
c      DIMENSION CR(NCOL,NROW,NLAY), CC(NCOL,NROW,NLAY)
c    2   , TRPY(NLAY), DELR(NCOL), DELC(NROW)
      DIMENSION CR(NCOL,NROW,NLAY), CC(NCOL,NROW,NLAY)
     2   , TRPY(ncol,nrow,nlay), DELR(NCOL), DELC(NROW)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C     ------------------------------------------------------------------
      ZERO=0.                                                         AT 0058
      TWO=2.                                                          AT 0066
C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
c      YX=TRPY(K)*TWO
      if(itrpy.eq.0)YX=TRPY(K,1,1)*TWO                                AT 0074
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C1------FOR EACH CELL CALCULATE BRANCH CONDUCTANCES FROM THAT CELL
C1------TO THE ONE ON THE RIGHT AND THE ONE IN FRONT.
      DO 40 I=1,NROW                                                  AT 009E
      DO 40 J=1,NCOL                                                  AT 00BC
      T1=CC(J,I,K)                                                    AT 00DA
C
C2------IF T=0 THEN SET CONDUCTANCE EQUAL TO 0. GO ON TO NEXT CELL.
C-Environment Agency (UK)-6.X & Y-dir transmissivities>>>>>>>>>>>>>>>>>>
c      IF(T1.NE.ZERO) GO TO 10
      IF(T1.NE.ZERO.or.itrpy.eq.2) GO TO 10                           AT 0103
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
      CR(J,I,K)=ZERO                                                  AT 0122
      GO TO 40                                                        AT 0146
C
C3------IF THIS IS NOT THE LAST COLUMN(RIGHTMOST) THEN CALCULATE
C3------BRANCH CONDUCTANCE IN THE ROW DIRECTION (CR) TO THE RIGHT.
   10 IF(J.EQ.NCOL) GO TO 30                                          AT 014B
      T2=CC(J+1,I,K)                                                  AT 0159
C-Environment Agency (UK)-6.X & Y-dir transmissivities>>>>>>>>>>>>>>>>>>
      if(itrpy.eq.2)then                                              AT 0185
        t1=cr(j,i,k)                                                  AT 0191
        t2=cr(j+1,i,k)                                                AT 01B8
      endif                                                           AT 01DF
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      CR(J,I,K)=TWO*T2*T1*DELC(I)/(T1*DELR(J+1)+T2*DELR(J))           AT 01DF
C
C4------IF THIS IS NOT THE LAST ROW(FRONTMOST) THEN CALCULATE
C4------BRANCH CONDUCTANCE IN THE COLUMN DIRECTION (CC) TO THE FRONT.
   30 IF(I.EQ.NROW) GO TO 40                                          AT 0234
      T2=CC(J,I+1,K)                                                  AT 0242
C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
c      CC(J,I,K)=YX*T2*T1*DELR(J)/(T1*DELC(I+1)+T2*DELC(I))
      if(itrpy.eq.0)then                                              AT 026F
       CC(J,I,K)=YX*T2*T1*DELR(J)/(T1*DELC(I+1)+T2*DELC(I))           AT 027B
      else                                                            AT 02CD
        t1=cc(j,i,k)                                                  AT 02D2
        t2=cc(j,i+1,k)                                                AT 02FC
```

```
         if(itrpy.ne.2)then                                          AT 0326
           t1=t1*trpy(j,i,k)                                         AT 0332
           t2=t2*trpy(j,i+1,k)                                       AT 0359
         endif                                                       AT 0380
         CC(J,I,K)=2*T2*T1*DELR(J)/(T1*DELC(I+1)+T2*DELC(I))         AT 0380
       endif                                                         AT 03D8
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
   40 CONTINUE                                                       AT 03D8
C
C5------RETURN
      RETURN                                                         AT 03E2
      END                                                            AT 03E7


      SUBROUTINE SBCF5B(HNEW,IBOUND,CR,CC,CV,TOP,NCOL,NROW,NLAY,KSTP,
     1      KPER,IBCFCB,BUFF,IOUT,ICBCFL,DELT,PERTIM,TOTIM,
     2      IDIR,IBDRET,ICHFLG,IC1,IC2,IR1,IR2,IL1,IL2)
C
C-----VERSION 1308 28JUNE1993 SBCF5B
C     ****************************************************************
C     COMPUTE FLOW BETWEEN ADJACENT CELLS IN A SUBREGION OF THE GRID
C     ****************************************************************
C
C     SPECIFICATIONS:
C     -----------------------------------------------------------------
      CHARACTER*16 TEXT(3)
      DOUBLE PRECISION HNEW,HD
C
      DIMENSION HNEW(NCOL,NROW,NLAY), IBOUND(NCOL,NROW,NLAY),
     1     CR(NCOL,NROW,NLAY), CC(NCOL,NROW,NLAY),
     2     CV(NCOL,NROW,NLAY), TOP(NCOL,NROW,NLAY),
     3     BUFF(NCOL,NROW,NLAY)
C
      COMMON /FLWCOM/LAYCON(200)
C
      DATA TEXT(1),TEXT(2),TEXT(3)                                   AT 0061
     1 /'FLOW RIGHT FACE ','FLOW FRONT FACE ','FLOW LOWER FACE '/
C     -----------------------------------------------------------------
C
C1------IF CELL-BY-CELL FLOWS WILL BE SAVED IN A FILE, SET FLAG IBD.
C1------RETURN IF FLOWS ARE NOT BEING SAVED OR RETURNED.
      ZERO=0.                                                        AT 0061
      IBD=0                                                          AT 0075
      IF(IBCFCB.GT.0) IBD=ICBCFL                                     AT 007F
      IF(IBD.EQ.0 .AND. IBDRET.EQ.0) RETURN                          AT 0096
C
C2------SET THE SUBREGION EQUAL TO THE ENTIRE GRID IF VALUES ARE BEING
C2------SAVED IN A FILE.
      IF(IBD.NE.0) THEN                                              AT 00AF
         K1=1                                                        AT 00BC
         K2=NLAY                                                     AT 00C6
         I1=1                                                        AT 00D1
         I2=NROW                                                     AT 00DB
         J1=1                                                        AT 00E6
         J2=NCOL                                                     AT 00F0
      END IF                                                         AT 00FB
C
C3------TEST FOR DIRECTION OF CALCULATION;  IF NOT ACROSS COLUMNS, GO TO
C3------STEP 4.  IF ONLY 1 COLUMN, RETURN.
      IF(IDIR.NE.1) GO TO 405                                        AT 00FB
      IF(NCOL.EQ.1) RETURN                                           AT 0107
C
C3A-----CALCULATE FLOW ACROSS COLUMNS (THROUGH RIGHT FACE).  IF NOT
C3A-----SAVING IN A FILE, SET THE SUBREGION.  CLEAR THE BUFFER.
      IF(IBD.EQ.0) THEN                                              AT 0113
         K1=IL1                                                      AT 0120
         K2=IL2                                                      AT 012B
         I1=IR1                                                      AT 0136
         I2=IR2                                                      AT 0141
         J1=IC1-1                                                    AT 014C
         IF(J1.LT.1) J1=1                                            AT 0158
         J2=IC2                                                      AT 016B
      END IF                                                         AT 0176
      DO 310 K=K1,K2                                                 AT 0176
      DO 310 I=I1,I2                                                 AT 01AB
      DO 310 J=J1,J2                                                 AT 01E0
      BUFF(J,I,K)=ZERO                                               AT 0215
  310 CONTINUE                                                       AT 024B
C
C3B-----FOR EACH CELL CALCULATE FLOW THRU RIGHT FACE & STORE IN BUFFER.
      IF(J2.EQ.NCOL) J2=J2-1                                         AT 0257
```

```
      DO 400 K=K1,K2                                                    AT 026E
      DO 400 I=I1,I2                                                    AT 02A3
      DO 400 J=J1,J2                                                    AT 02D8
      IF(ICHFLG.EQ.0) THEN                                              AT 030D
         IF((IBOUND(J,I,K).LE.0) .AND. (IBOUND(J+1,I,K).LE.0)) GO TO 400   AT 0319
      ELSE                                                              AT 0357
         IF((IBOUND(J,I,K).EQ.0) .OR. (IBOUND(J+1,I,K).EQ.0)) GO TO 400    AT 035C
      END IF                                                            AT 039E
      HDIFF=HNEW(J,I,K)-HNEW(J+1,I,K)                                   AT 039E
      BUFF(J,I,K)=HDIFF*CR(J,I,K)                                       AT 03D3
  400 CONTINUE                                                          AT 0407
C
C3C-----RECORD CONTENTS OF BUFFER AND RETURN.
      IF(IBD.EQ.1)                                                      AT 0416
     1   CALL UBUDSV(KSTP,KPER,TEXT(1),IBCFCB,BUFF,NCOL,NROW,NLAY,IOUT)
      IF(IBD.EQ.2) CALL UBDSV1(KSTP,KPER,TEXT(1),IBCFCB,BUFF,NCOL,NROW,     AT 0450
     1     NLAY,IOUT,DELT,PERTIM,TOTIM,IBOUND)
      RETURN                                                            AT 0496
C
C4------TEST FOR DIRECTION OF CALCULATION;  IF NOT ACROSS ROWS, GO TO
C4------STEP 5.  IF ONLY 1 ROW, RETURN.
  405 IF(IDIR.NE.2) GO TO 505                                           AT 049B
      IF(NROW.EQ.1) RETURN                                              AT 04A7
C
C4A-----CALCULATE FLOW ACROSS ROWS (THROUGH FRONT FACE).  IF NOT SAVING
C4A-----IN A FILE, SET THE SUBREGION.  CLEAR THE BUFFER.
      IF(IBD.EQ.0) THEN                                                 AT 04B3
         K1=IL1                                                         AT 04C0
         K2=IL2                                                         AT 04CB
         I1=IR1-1                                                       AT 04D6
         IF(I1.LT.1) I1=1                                               AT 04E2
         I2=IR2                                                         AT 04F5
         J1=IC1                                                         AT 0500
         J2=IC2                                                         AT 050B
      END IF                                                            AT 0516
      DO 410 K=K1,K2                                                    AT 0516
      DO 410 I=I1,I2                                                    AT 054B
      DO 410 J=J1,J2                                                    AT 0580
      BUFF(J,I,K)=ZERO                                                  AT 05B5
  410 CONTINUE                                                          AT 05EB
C
C4B-----FOR EACH CELL CALCULATE FLOW THRU FRONT FACE & STORE IN BUFFER.
      IF(I2.EQ.NROW) I2=I2-1                                            AT 05F7
      DO 500 K=K1,K2                                                    AT 060E
      DO 500 I=I1,I2                                                    AT 0643
      DO 500 J=J1,J2                                                    AT 0678
      IF(ICHFLG.EQ.0) THEN                                              AT 06AD
         IF((IBOUND(J,I,K).LE.0) .AND. (IBOUND(J,I+1,K).LE.0)) GO TO 500   AT 06B9
      ELSE                                                              AT 0722
         IF((IBOUND(J,I,K).EQ.0) .OR. (IBOUND(J,I+1,K).EQ.0)) GO TO 500    AT 0727
      END IF                                                            AT 0794
      HDIFF=HNEW(J,I,K)-HNEW(J,I+1,K)                                   AT 0794
      BUFF(J,I,K)=HDIFF*CC(J,I,K)                                       AT 07EE
  500 CONTINUE                                                          AT 0822
C
C4C-----RECORD CONTENTS OF BUFFER AND RETURN.
      IF(IBD.EQ.1)                                                      AT 0831
     1   CALL UBUDSV(KSTP,KPER,TEXT(2),IBCFCB,BUFF,NCOL,NROW,NLAY,IOUT)
      IF(IBD.EQ.2) CALL UBDSV1(KSTP,KPER,TEXT(2),IBCFCB,BUFF,NCOL,NROW,     AT 086B
     1     NLAY,IOUT,DELT,PERTIM,TOTIM,IBOUND)
      RETURN                                                            AT 08B1
C
C5------DIRECTION OF CALCULATION IS ACROSS LAYERS BY ELIMINATION.  IF
C5------ONLY 1 LAYER, RETURN.
  505 IF(NLAY.EQ.1) RETURN                                              AT 08B6
C
C5A-----CALCULATE FLOW ACROSS LAYERS (THROUGH LOWER FACE).  IF NOT
C5A-----SAVING IN A FILE, SET THE SUBREGION.  CLEAR THE BUFFER.
      IF(IBD.EQ.0) THEN                                                 AT 08C2
         K1=IL1-1                                                       AT 08CF
         IF(K1.LT.1) K1=1                                               AT 08DB
         K2=IL2                                                         AT 08EE
         I1=IR1                                                         AT 08F9
         I2=IR2                                                         AT 0904
         J1=IC1                                                         AT 090F
         J2=IC2                                                         AT 091A
      END IF                                                            AT 0925
      DO 510 K=K1,K2                                                    AT 0925
      DO 510 I=I1,I2                                                    AT 095A
      DO 510 J=J1,J2                                                    AT 098F
      BUFF(J,I,K)=ZERO                                                  AT 09C4
  510 CONTINUE                                                          AT 09FA
```

```
C
C5B-----FOR EACH CELL CALCULATE FLOW THRU LOWER FACE & STORE IN BUFFER.
      IF(K2.EQ.NLAY) K2=K2-1                                             AT 0A06
      KT=0                                                               AT 0A1D
      DO 600 K=1,K2                                                      AT 0A27
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
c      IF(LAYCON(K).EQ.3 .OR. LAYCON(K).EQ.2) KT=KT+1
      IF(LAYCON(K).EQ.3 .OR. LAYCON(K).EQ.2 .or. laycon(k).eq.5) KT=KT+1 AT 0A55
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      IF(K.LT.K1) GO TO 600                                              AT 0A85
      DO 590 I=I1,I2                                                     AT 0A97
      DO 590 J=J1,J2                                                     AT 0ACC
      IF(ICHFLG.EQ.0) THEN                                               AT 0B01
         IF((IBOUND(J,I,K).LE.0) .AND. (IBOUND(J,I,K+1).LE.0)) GO TO 590 AT 0B0D
      ELSE                                                               AT 0B71
         IF((IBOUND(J,I,K).EQ.0) .OR. (IBOUND(J,I,K+1).EQ.0)) GO TO 590  AT 0B76
      END IF                                                             AT 0BE2
      HD=HNEW(J,I,K+1)                                                   AT 0BE2
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
c      IF(LAYCON(K+1).NE.3 .AND. LAYCON(K+1).NE.2) GO TO 580
      IF(LAYCON(K+1).NE.3 .AND. LAYCON(K+1).NE.2 .and. laycon(k+1).ne.5) AT 0C1D
     1      GO TO 580
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      TMP=HD                                                             AT 0C4D
      IF(TMP.LT.TOP(J,I,KT+1)) HD=TOP(J,I,KT+1)                          AT 0C65
  580 HDIFF=HNEW(J,I,K)-HD                                               AT 0CDE
      BUFF(J,I,K)=HDIFF*CV(J,I,K)                                        AT 0D15
  590 CONTINUE                                                           AT 0D49
  600 CONTINUE                                                           AT 0D53
C
C5C-----RECORD CONTENTS OF BUFFER AND RETURN.
      IF(IBD.EQ.1)                                                       AT 0D58
     1   CALL UBUDSV(KSTP,KPER,TEXT(3),IBCFCB,BUFF,NCOL,NROW,NLAY,IOUT)
      IF(IBD.EQ.2) CALL UBDSV1(KSTP,KPER,TEXT(3),IBCFCB,BUFF,NCOL,NROW,  AT 0D92
     1     NLAY,IOUT,DELT,PERTIM,TOTIM,IBOUND)
      RETURN                                                             AT 0DD8
      END                                                                AT 0DDD
C-Environment Agency (UK)-2.Variable storage with depth (VSD)>>>>>>>>>>>
c      SUBROUTINE SBCF5S(VBNM,VBVL,MSUM,HNEW,IBOUND,HOLD,SC1,
c     1   TOP,SC2,DELT,ISS,NCOL,NROW,NLAY,KSTP,KPER,IBCFCB,
c     2   ICBCFL,BUFF,IOUT,PERTIM,TOTIM)


      SUBROUTINE SBCF5S(VBNM,VBVL,MSUM,HNEW,IBOUND,HOLD,SC1,
     1   TOP,SC2,DELT,ISS,NCOL,NROW,NLAY,KSTP,KPER,IBCFCB,
     2   ICBCFL,BUFF,IOUT,PERTIM,TOTIM,bot,vmid,vsgrad,vsmax,vsmid)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-----VERSION 1257 28JUNE1993 SBCF5S
C     ******************************************************************
C     COMPUTE STORAGE BUDGET FLOW TERM FOR BCF.
C     ******************************************************************
C
C     SPECIFICATIONS:
C     ------------------------------------------------------------------
      CHARACTER*16 VBNM(MSUM),TEXT
C-Environment Agency (UK)-2.Variable storage with depth (VSD)>>>>>>>>>>>
c      DOUBLE PRECISION HNEW,STOIN,STOUT,SSTRG
      DOUBLE PRECISION HNEW,STOIN,STOUT,SSTRG,htmp,vm,vsm,vsg,esm,bt,
     1                  stmp,sold,snew,scof,rho1,rho2,tp
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
      DIMENSION HNEW(NCOL,NROW,NLAY), IBOUND(NCOL,NROW,NLAY),
     1   HOLD(NCOL,NROW,NLAY),SC1(NCOL,NROW,NLAY),VBVL(4,MSUM),
     2   SC2(NCOL,NROW,NLAY),TOP(NCOL,NROW,NLAY),BUFF(NCOL,NROW,NLAY)
C-Environment Agency (UK)-2.Variable storage with depth (VSD)>>>>>>>>>>>
      dimension bot(ncol,nrow,nlay), vmid(ncol,nrow,nlay),
     1  vsgrad(ncol,nrow,nlay), vsmax(ncol,nrow,nlay),
     2  vsmid(ncol,nrow,nlay)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
      COMMON /FLWCOM/LAYCON(200)
C
      DATA TEXT /'         STORAGE'/                                     AT 008A
C     ------------------------------------------------------------------
C
C1------RETURN IF STEADY STATE.
      IF(ISS.NE.0) RETURN                                                AT 008A
C
C2------INITIALIZE BUDGET ACCUMULATORS AND 1/DELT.
      ZERO=0.                                                            AT 0096
      STOIN=ZERO                                                         AT 00AA
```

```
      STOUT=ZERO                                                          AT 00C2
      ONE=1.                                                              AT 00DA
      TLED=ONE/DELT                                                       AT 00EE
C
C3------IF CELL-BY-CELL FLOWS WILL BE SAVED, SET FLAG IBD.
      IBD=0                                                               AT 0100
      IF(IBCFCB.GT.0) IBD=ICBCFL                                          AT 010A
C
C4------CLEAR BUFFER.
      DO 210 K=1,NLAY                                                     AT 0121
      DO 210 I=1,NROW                                                     AT 014E
      DO 210 J=1,NCOL                                                     AT 017B
      BUFF(J,I,K)=ZERO                                                    AT 01A8
210   CONTINUE                                                            AT 01E1
C
C5------LOOP THROUGH EVERY CELL IN THE GRID.
      KT=0                                                                AT 01ED
C-Environment Agency (UK)-2.Variable storage with depth (VSD)>>>>>>>>>>>
      kb=0                                                                AT 01F7
      kg=0                                                                AT 0201
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      DO 300 K=1,NLAY                                                     AT 020B
      LC=LAYCON(K)                                                        AT 0238
C-Environment Agency (UK)-2.Variable storage with depth (VSD)>>>>>>>>>>>
c      IF(LC.EQ.3 .OR. LC.EQ.2) KT=KT+1
      IF(LC.EQ.3 .OR. LC.EQ.2 .or. lc.eq.5) KT=KT+1                       AT 0245
      if(lc.eq.1 .or. lc.eq.3 .or. lc.eq.4 .or. lc.eq.5) kb=kb+1          AT 0266
      if(lc.eq.4 .or. lc.eq.5) kg=kg+1                                    AT 02A0
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      DO 300 I=1,NROW                                                     AT 02C0
      DO 300 J=1,NCOL                                                     AT 02ED
C
C6------SKIP NO-FLOW AND CONSTANT-HEAD CELLS.
      IF(IBOUND(J,I,K).LE.0) GO TO 300                                    AT 031A
      HSING=HNEW(J,I,K)                                                   AT 0350
C
C7-----CHECK LAYER TYPE TO SEE IF ONE STORAGE CAPACITY OR TWO.
C-Environment Agency (UK)-2.Variable storage with depth (VSD)>>>>>>>>>>>
c      IF(LC.NE.3 .AND. LC.NE.2) GO TO 285
      IF(LC.eq.0 .or. LC.eq.1) GO TO 285                                  AT 0389
      IF(LC.eq.4 .or. LC.eq.5) GO TO 400                                  AT 03A3
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C7A----TWO STORAGE CAPACITIES.
      TP=TOP(J,I,KT)                                                      AT 03BD
      RHO2=SC2(J,I,KT)*TLED                                               AT 03FA
      RHO1=SC1(J,I,K)*TLED                                                AT 0434
      SOLD=RHO2                                                           AT 046E
      IF(HOLD(J,I,K).GT.TP) SOLD=RHO1                                     AT 0486
      SNEW=RHO2                                                           AT 04E1
      IF(HSING.GT.TP) SNEW=RHO1                                           AT 04F9
      STRG=SOLD*(HOLD(J,I,K)-TP) + SNEW*TP - SNEW*HSING                   AT 0527
      GO TO 288                                                           AT 0583
C
C7B----ONE STORAGE CAPACITY.
  285 RHO=SC1(J,I,K)*TLED                                                 AT 0588
      STRG=RHO*HOLD(J,I,K) - RHO*HSING                                    AT 05C2

C-Environment Agency (UK)-2.Variable storage with depth (VSD)>>>>>>>>>>>
      goto 288                                                            AT 0604
c
c7c1----Varying Storage with Depth (VSD), so calculate storage change
  400 vm=vsmid(j,i,kg)                                                    AT 0609
      vsm=vsmax(j,i,kg)                                                   AT 0646
      vsg=vsgrad(j,i,kg)                                                  AT 0683
      esm=vm+((vsm-1.0)/vsg)                                              AT 06C0
      bt=bot(j,i,kb)                                                      AT 06DF
      rho1=sc1(j,i,k)*tled                                                AT 071C
      rho2=sc1(j,i,k)*tled                                                AT 0756
      tp=1.0E+10                                                          AT 078D
      if(laycon(k).eq.5)then                                              AT 07A1
        rho2=sc2(j,i,kt)*tled                                             AT 07B5
        tp=top(j,i,kt)                                                    AT 07EF
      endif                                                               AT 082C
c
c7c2----Find Storage At Start Of Time Step
      htmp=hold(j,i,k)                                                    AT 082C
      itmp=1                                                              AT 0869
 420  if(htmp.le.vm)then                                                  AT 0873
C-Environment Agency (UK)-9.Bug fix>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
c        stmp = rho2*(htmp-bt)
        if((laycon(k).eq.5.and.htmp.le.tp).or.laycon(k).eq.4)then         AT 0889
```

```
            stmp = rho2*(htmp-bt)                                                    AT 08C7
          else                                                                       AT 08E0
            stmp = rho2*(tp-bt) + rho1*(htmp-tp)                                     AT 08E5
          endif                                                                      AT 0912
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
        elseif((laycon(k).eq.5.and.htmp.le.tp).or.laycon(k).eq.4)then                AT 0912
          if(htmp.le.esm)then                                                        AT 0955
            stmp = rho2*(htmp-bt) + (rho2*vsg*(htmp-vm)**2)/2                        AT 096B
          else                                                                       AT 09A8
            stmp = rho2*(esm-bt) + (rho2*vsg*(esm-vm)**2)/2 +                        AT 09AD
     1             (htmp-esm)*vsm*rho2
          endif                                                                      AT 0A04
        else                                                                         AT 0A04
          if(tp.le.esm)then                                                          AT 0A09
            stmp = rho2*(tp-bt) + (rho2*vsg*(tp-vm)**2)/2 + rho1*(htmp-tp)           AT 0A1F
          else                                                                       AT 0A70
            stmp = rho2*(esm-bt) + (rho2*vsg*(esm-vm)**2)/2 +                        AT 0A75
     1             (tp-esm)*vsm*rho2 + rho1*(htmp-tp)
          endif                                                                      AT 0AE0
        endif                                                                        AT 0AE0
c
c7c3----Find Storage At End Of Time Step
        if(itmp.eq.1)then                                                            AT 0AE0
          sold=stmp                                                                  AT 0AED
          htmp=hnew(j,i,k)                                                           AT 0B05
          itmp=0                                                                     AT 0B42
          goto 420                                                                   AT 0B4C
        endif                                                                        AT 0B51
        snew=stmp                                                                    AT 0B51
c
c7c4----Add Storage Terms To RHS and HCOF.
        strg = sold - snew                                                           AT 0B69
c
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C8-----STORE CELL-BY-CELL FLOW IN BUFFER AND ADD TO ACCUMULATORS.
  288 BUFF(J,I,K)=STRG                                                               AT 0B7C
      SSTRG=STRG                                                                     AT 0BB9
      IF(STRG) 292,300,294                                                           AT 0BD1
  292 STOUT=STOUT-SSTRG                                                              AT 0BF1
      GO TO 300                                                                      AT 0C04
  294 STOIN=STOIN+SSTRG                                                              AT 0C09
C
  300 CONTINUE                                                                       AT 0C1C
C
C9-----IF IBD FLAG IS SET RECORD THE CONTENTS OF THE BUFFER.
      IF(IBD.EQ.1) CALL UBUDSV(KSTP,KPER,TEXT,                                       AT 0C2B
     1                       IBCFCB,BUFF,NCOL,NROW,NLAY,IOUT)
      IF(IBD.EQ.2) CALL UBDSV1(KSTP,KPER,TEXT,IBCFCB,                                AT 0C63
     1            BUFF,NCOL,NROW,NLAY,IOUT,DELT,PERTIM,TOTIM,IBOUND)
C
C10-----ADD TOTAL RATES AND VOLUMES TO VBVL & PUT TITLE IN VBNM.
      SIN=STOIN                                                                      AT 0CA7
      SOUT=STOUT                                                                     AT 0CBF
      VBVL(1,MSUM)=VBVL(1,MSUM)+SIN*DELT                                             AT 0CD7
      VBVL(2,MSUM)=VBVL(2,MSUM)+SOUT*DELT                                            AT 0CF7
      VBVL(3,MSUM)=SIN                                                               AT 0D10
      VBVL(4,MSUM)=SOUT                                                              AT 0D2C
      VBNM(MSUM)=TEXT                                                                AT 0D4B
      MSUM=MSUM+1                                                                    AT 0D67
C
C11----RETURN.
      RETURN                                                                         AT 0D6C
      END                                                                            AT 0D71
COMMENT - SCOF has not been used


      SUBROUTINE SBCF5F(VBNM,VBVL,MSUM,HNEW,IBOUND,CR,CC,CV,TOP,DELT,
     1         NCOL,NROW,NLAY,KSTP,KPER,IBCFCB,BUFF,IOUT,ICBCFL,
     2         PERTIM,TOTIM,ICHFLG)
C-----VERSION 1315 18DEC1992 SBCF5F
C     ******************************************************************
C     COMPUTE FLOW FROM CONSTANT-HEAD CELLS
C     ******************************************************************
C
C     SPECIFICATIONS:
C     -------------------------------------------------------------------
      CHARACTER*16 VBNM(MSUM),TEXT
      DOUBLE PRECISION HNEW,HD,CHIN,CHOUT,XX1,XX2,XX3,XX4,XX5,XX6
C
      DIMENSION HNEW(NCOL,NROW,NLAY), IBOUND(NCOL,NROW,NLAY),
```

```
         1     CR(NCOL,NROW,NLAY), CC(NCOL,NROW,NLAY),
         2     CV(NCOL,NROW,NLAY), VBVL(4,MSUM),
         3     TOP(NCOL,NROW,NLAY),BUFF(NCOL,NROW,NLAY)
C
      COMMON /FLWCOM/LAYCON(200)
C
      DATA TEXT /'   CONSTANT HEAD'/                                      AT 006C
C     ------------------------------------------------------------------
C
C1------SET IBD TO INDICATE IF CELL-BY-CELL BUDGET VALUES WILL BE SAVED.
      IBD=0                                                               AT 006C
      IF(IBCFCB.LT.0 .AND. ICBCFL.NE.0) IBD=-1                            AT 0076
      IF(IBCFCB.GT.0) IBD=ICBCFL                                          AT 0098
C
C2------CLEAR BUDGET ACCUMULATORS.
      ZERO=0.                                                             AT 00AF
      CHIN=ZERO                                                           AT 00C3
      CHOUT=ZERO                                                          AT 00DB
      IBDLBL=0                                                            AT 00F3
C
C3------CLEAR BUFFER.
      DO 5 K=1,NLAY                                                       AT 00FD
      DO 5 I=1,NROW                                                       AT 012A
      DO 5 J=1,NCOL                                                       AT 0157
      BUFF(J,I,K)=ZERO                                                    AT 0184
5     CONTINUE                                                            AT 01B7
C
C3A-----IF SAVING CELL-BY-CELL FLOW IN A LIST, COUNT CONSTANT-HEAD
C3A-----CELLS AND WRITE HEADER RECORDS.
      IF(IBD.EQ.2) THEN                                                   AT 01C0
         NCH=0                                                            AT 01CD
         DO 7 K=1,NLAY                                                    AT 01D7
         DO 7 I=1,NROW                                                    AT 0204
         DO 7 J=1,NCOL                                                    AT 0231
         IF(IBOUND(J,I,K).LT.0) NCH=NCH+1                                 AT 025E
7        CONTINUE                                                         AT 0294
         CALL UBDSV2(KSTP,KPER,TEXT,IBCFCB,NCOL,NROW,NLAY,                AT 029D
     1          NCH,IOUT,DELT,PERTIM,TOTIM,IBOUND)
      END IF                                                              AT 02D8
C
C4------LOOP THROUGH EACH CELL AND CALCULATE FLOW INTO MODEL FROM EACH
C4------CONSTANT-HEAD CELL.
      KT=0                                                                AT 02D8
      DO 200 K=1,NLAY                                                     AT 02E2
      LC=LAYCON(K)                                                        AT 030F
C-Environment Agency (UK)-2.Variable storage with depth (VSD)>>>>>>>>>>>
c      IF(LC.EQ.3 .OR. LC.EQ.2) KT=KT+1
      IF(LC.EQ.3 .OR. LC.EQ.2 .or. lc.eq.5) KT=KT+1                       AT 031C
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      DO 200 I=1,NROW                                                     AT 033D
      DO 200 J=1,NCOL                                                     AT 036A
C
C5------IF CELL IS NOT CONSTANT HEAD SKIP IT & GO ON TO NEXT CELL.
      IF (IBOUND(J,I,K).GE.0)GO TO 200                                    AT 0397
C
C6------CLEAR VALUES FOR FLOW RATE THROUGH EACH FACE OF CELL.
      X1=ZERO                                                             AT 03C7
      X2=ZERO                                                             AT 03DF
      X3=ZERO                                                             AT 03F7
      X4=ZERO                                                             AT 040F
      X5=ZERO                                                             AT 0427
      X6=ZERO                                                             AT 043F
C
C7------CALCULATE FLOW THROUGH THE LEFT FACE.
C7------COMMENTS A-C APPEAR ONLY IN THE SECTION HEADED BY COMMENT 7,
C7------BUT THEY APPLY IN A SIMILAR MANNER TO SECTIONS 8-12.
C
C7A-----IF THERE IS NO FLOW TO CALCULATE THROUGH THIS FACE, THEN GO ON
C7A-----TO NEXT FACE.  NO FLOW OCCURS AT THE EDGE OF THE GRID, TO AN
C7A-----ADJACENT NO-FLOW CELL, OR TO AN ADJACENT CONSTANT-HEAD CELL
C7A-----WHEN ICHFLG IS 0.
      IF(J.EQ.1) GO TO 30                                                 AT 0457
      IF(IBOUND(J-1,I,K).EQ.0) GO TO 30                                   AT 0460
      IF(ICHFLG.EQ.0 .AND. IBOUND(J-1,I,K).LT.0) GO TO 30                 AT 048D
C
C7B-----CALCULATE FLOW THROUGH THIS FACE INTO THE ADJACENT CELL.
      HDIFF=HNEW(J,I,K)-HNEW(J-1,I,K)                                     AT 04C6
      X1=HDIFF*CR(J-1,I,K)                                                AT 04F8
      XX1=X1                                                              AT 0528
C
C7C-----ACCUMULATE POSITIVE AND NEGATIVE FLOW.
      IF (X1) 10,30,20                                                    AT 0540
```

```
   10 CHOUT=CHOUT-XX1                                                      AT 055A
      GO TO 30                                                             AT 056D
   20 CHIN=CHIN+XX1                                                        AT 0572
C
C8------CALCULATE FLOW THROUGH THE RIGHT FACE.
   30 IF(J.EQ.NCOL) GO TO 60                                               AT 0585
      IF(IBOUND(J+1,I,K).EQ.0) GO TO 60                                    AT 0596
      IF(ICHFLG.EQ.0 .AND. IBOUND(J+1,I,K).LT.0) GO TO 60                  AT 05C6
      HDIFF=HNEW(J,I,K)-HNEW(J+1,I,K)                                      AT 05FF
      X2=HDIFF*CR(J,I,K)                                                   AT 0631
      XX2=X2                                                               AT 0661
      IF(X2)40,60,50                                                       AT 0679
   40 CHOUT=CHOUT-XX2                                                      AT 0693
      GO TO 60                                                             AT 06A6
   50 CHIN=CHIN+XX2                                                        AT 06AB
C
C9------CALCULATE FLOW THROUGH THE BACK FACE.
   60 IF(I.EQ.1) GO TO 90                                                  AT 06BE
      IF (IBOUND(J,I-1,K).EQ.0) GO TO 90                                   AT 06CB
      IF(ICHFLG.EQ.0 .AND. IBOUND(J,I-1,K).LT.0) GO TO 90                  AT 0702
      HDIFF=HNEW(J,I,K)-HNEW(J,I-1,K)                                      AT 0742
      X3=HDIFF*CC(J,I-1,K)                                                 AT 0799
      XX3=X3                                                               AT 07CC
      IF(X3) 70,90,80                                                      AT 07E4
   70 CHOUT=CHOUT-XX3                                                      AT 07FE
      GO TO 90                                                             AT 0811
   80 CHIN=CHIN+XX3                                                        AT 0816
C
C10-----CALCULATE FLOW THROUGH THE FRONT FACE.
   90 IF(I.EQ.NROW) GO TO 120                                              AT 0829
      IF(IBOUND(J,I+1,K).EQ.0) GO TO 120                                   AT 083A
      IF(ICHFLG.EQ.0 .AND. IBOUND(J,I+1,K).LT.0) GO TO 120                 AT 086B
      HDIFF=HNEW(J,I,K)-HNEW(J,I+1,K)                                      AT 08AB
      X4=HDIFF*CC(J,I,K)                                                   AT 0902
      XX4=X4                                                               AT 0932
      IF (X4) 100,120,110                                                  AT 094A
  100 CHOUT=CHOUT-XX4                                                      AT 0964
      GO TO 120                                                            AT 0977
  110 CHIN=CHIN+XX4                                                        AT 097C
C
C11-----CALCULATE FLOW THROUGH THE UPPER FACE.
  120 IF(K.EQ.1) GO TO 150                                                 AT 098F
      IF (IBOUND(J,I,K-1).EQ.0) GO TO 150                                  AT 099C
      IF(ICHFLG.EQ.0 .AND. IBOUND(J,I,K-1).LT.0) GO TO 150                 AT 09D1
      HD=HNEW(J,I,K)                                                       AT 0A0F
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
c      IF(LC.NE.3 .AND. LC.NE.2) GO TO 122
      IF(LC.NE.3 .AND. LC.NE.2 .and. lc.ne.5) GO TO 122                    AT 0A46
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      TMP=HD                                                               AT 0A6D
      IF(TMP.LT.TOP(J,I,KT)) HD=TOP(J,I,KT)                                AT 0A85
  122 HDIFF=HD-HNEW(J,I,K-1)                                               AT 0AF6
      X5=HDIFF*CV(J,I,K-1)                                                 AT 0B2B
      XX5=X5                                                               AT 0B5C
      IF(X5) 130,150,140                                                   AT 0B74
  130 CHOUT=CHOUT-XX5                                                      AT 0B8E
      GO TO 150                                                            AT 0BA1
  140 CHIN=CHIN+XX5                                                        AT 0BA6
C
C12-----CALCULATE FLOW THROUGH THE LOWER FACE.
  150 IF(K.EQ.NLAY) GO TO 180                                              AT 0BB9
      IF(IBOUND(J,I,K+1).EQ.0) GO TO 180                                   AT 0BCA
      IF(ICHFLG.EQ.0 .AND. IBOUND(J,I,K+1).LT.0) GO TO 180                 AT 0BF9
      HD=HNEW(J,I,K+1)                                                     AT 0C37
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
c      IF(LAYCON(K+1).NE.3 .AND. LAYCON(K+1).NE.2) GO TO 152
      IF(LAYCON(K+1).NE.3 .AND. LAYCON(K+1).NE.2 .and. laycon(k+1).ne.5)   AT 0C6F
     1      GO TO 152
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      TMP=HD                                                               AT 0C9F
      IF(TMP.LT.TOP(J,I,KT+1)) HD=TOP(J,I,KT+1)                            AT 0CB7
  152 HDIFF=HNEW(J,I,K)-HD                                                 AT 0D2A
      X6=HDIFF*CV(J,I,K)                                                   AT 0D5E
      XX6=X6                                                               AT 0D8E
      IF(X6) 160,180,170                                                   AT 0DA6
  160 CHOUT=CHOUT-XX6                                                      AT 0DC0
      GO TO 180                                                            AT 0DD3
  170 CHIN=CHIN+XX6                                                        AT 0DD8
C
C13-----SUM THE FLOWS THROUGH SIX FACES OF CONSTANT HEAD CELL, AND
C13-----STORE SUM IN BUFFER.
 180  RATE=X1+X2+X3+X4+X5+X6                                               AT 0DEB
```

```
      BUFF(J,I,K)=RATE                                                     AT 0E16
C
C14-----PRINT THE FLOW FOR THE CELL IF REQUESTED.
      IF(IBD.LT.0) THEN                                                    AT 0E4D
         IF(IBDLBL.EQ.0) WRITE(IOUT,899) TEXT,KPER,KSTP                    AT 0E5A
  899    FORMAT(1X,/1X,A,'   PERIOD',I3,'   STEP',I3)                      AT 0EC5
         WRITE(IOUT,900) K,I,J,RATE                                        AT 0F0C
  900    FORMAT(1X,'LAYER',I3,'   ROW',I4,'   COL',I4,                     AT 0F82
     1       '   RATE',1PG15.6)
         IBDLBL=1                                                          AT 0FDD
      END IF                                                               AT 0FE7
C
C15-----IF SAVING CELL-BY-CELL FLOW IN LIST, WRITE FLOW FOR CELL.
      IF(IBD.EQ.2) CALL UBDSVA(IBCFCB,NCOL,NROW,J,I,K,RATE,IBOUND,NLAY)    AT 0FE7
  200 CONTINUE                                                             AT 1027
C
C16-----IF SAVING CELL-BY-CELL FLOW IN 3-D ARRAY, WRITE THE ARRAY.
      IF(IBD.EQ.1) CALL UBUDSV(KSTP,KPER,TEXT,                             AT 1036
     1                   IBCFCB,BUFF,NCOL,NROW,NLAY,IOUT)
C
C17-----SAVE TOTAL CONSTANT HEAD FLOWS AND VOLUMES IN VBVL TABLE
C17-----FOR INCLUSION IN BUDGET. PUT LABELS IN VBNM TABLE.
      CIN=CHIN                                                             AT 106E
      COUT=CHOUT                                                           AT 1086
      VBVL(1,MSUM)=VBVL(1,MSUM)+CIN*DELT                                   AT 109E
      VBVL(2,MSUM)=VBVL(2,MSUM)+COUT*DELT                                  AT 10BE
      VBVL(3,MSUM)=CIN                                                     AT 10D7
      VBVL(4,MSUM)=COUT                                                    AT 10F3
      VBNM(MSUM)=TEXT                                                      AT 1112
      MSUM=MSUM+1                                                          AT 112E
C
C18-----RETURN.
      RETURN                                                               AT 1133
      END                                                                  AT 1138
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
c      SUBROUTINE SBCF5H(HNEW,IBOUND,CR,CC,CV,HY,TRPY,DELR,DELC
c     1,BOT,TOP,K,KB,KT,KITER,KSTP,KPER,NCOL,NROW,NLAY,IOUT
c     2,WETDRY,IWDFLG,CVWD,WETFCT,IWETIT,IHDWET,HDRY,BUFF)


      SUBROUTINE SBCF5H(HNEW,IBOUND,CR,CC,CV,HY,TRPY,DELR,DELC
     1,BOT,TOP,K,KB,KT,KITER,KSTP,KPER,NCOL,NROW,NLAY,IOUT
     2,WETDRY,IWDFLG,CVWD,WETFCT,IWETIT,IHDWET,HDRY,BUFF,itrpy
     3,vkgrad,kg,hold,ihold,iss,vmid,vkmax)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-----VERSION 1501 29JUNE1993 SBCF5H
C     ******************************************************************
C     COMPUTE CONDUCTANCE FOR ONE LAYER FROM SATURATED THICKNESS AND
C     HYDRAULIC CONDUCTIVITY
C     ******************************************************************
C
C      SPECIFICATIONS:
C     ------------------------------------------------------------------
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
c      DOUBLE PRECISION HNEW,HD,BBOT,TTOP
      DOUBLE PRECISION HNEW,HD,BBOT,TTOP,hda
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
c      DIMENSION HNEW(NCOL,NROW,NLAY),IBOUND(NCOL,NROW,NLAY)
c     1,CR(NCOL,NROW,NLAY), CC(NCOL,NROW,NLAY), CV(NCOL,NROW,NLAY)
c     2,HY(NCOL,NROW,NLAY), TRPY(NLAY), DELR(NCOL), DELC(NROW)
c     3,BOT(NCOL,NROW,NLAY),TOP(NCOL,NROW,NLAY),WETDRY(NCOL,NROW,NLAY)
c     4,CVWD(NCOL,NROW,NLAY),BUFF(NCOL,NROW,NLAY)
      DIMENSION HNEW(NCOL,NROW,NLAY),IBOUND(NCOL,NROW,NLAY)
     1,CR(NCOL,NROW,NLAY), CC(NCOL,NROW,NLAY), CV(NCOL,NROW,NLAY)
     2,HY(NCOL,NROW,NLAY), TRPY(ncol,nrow,NLAY), DELR(NCOL), DELC(NROW)
     3,BOT(NCOL,NROW,NLAY),TOP(NCOL,NROW,NLAY),WETDRY(NCOL,NROW,NLAY)
     4,CVWD(NCOL,NROW,NLAY),BUFF(NCOL,NROW,NLAY),vkgrad(ncol,nrow,nlay)
     5,hold(NCOL,NROW,NLAY),vmid(ncol,nrow,nlay),vkmax(ncol,nrow,nlay)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      CHARACTER*3 ACNVRT
      DIMENSION ICNVRT(5),JCNVRT(5),ACNVRT(5)
C
      COMMON /FLWCOM/LAYCON(200)
      COMMON /FLWAVG/LAYAVG(200)
C     ------------------------------------------------------------------
C
C1------LOOP THROUGH EACH CELL IN LAYER AND CALCULATE TRANSMISSIVITY AT
C1------EACH ACTIVE CELL.
      ZERO=0.                                                              AT 0085
```

```
      NCNVRT=0                                                          AT 0099
      IHDCNV=0                                                          AT 00A3
      ITFLG=1                                                           AT 00AD
      IF(IWDFLG.NE.0) ITFLG=MOD(KITER,IWETIT)                           AT 00B7
      DO 200 I=1,NROW                                                   AT 00D4
      DO 200 J=1,NCOL                                                   AT 0101
C
C2------IF CELL IS ACTIVE, THEN SKIP TO CODE THAT CALCULATES SATURATED
C2------THICKNESS.
      IF(IBOUND(J,I,K).NE.0) GO TO 20                                   AT 012E
C
C3------DETERMINE IF THE CELL CAN CONVERT BETWEEN wet and dry.
C3------IF NOT, SKIP TO CODE THAT SETS TRANSMISSIVITY TO 0.
      IF(ITFLG.NE.0) GO TO 6                                            AT 0166
      IF(WETDRY(J,I,KB).EQ.ZERO)GO TO 6                                 AT 0173
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
      WD=WETDRY(J,I,KB)                                                 AT 01B1
      IF(WD.LT.ZERO) WD=-WD                                             AT 01EA
      TURNON=BOT(J,I,KB)+WD                                             AT 020F
C
C3A-----CHECK HEAD IN CELL BELOW TO SEE IF WETTING THRESHOLD HAS BEEN
C3A-----REACHED.
      IF(K.EQ.NLAY)GO TO 2                                              AT 0248
      HTMP=HNEW(J,I,K+1)                                                AT 0258
      IF(IBOUND(J,I,K+1).GT.0.AND.HTMP.GE.TURNON)GO TO 9                AT 0293
C
C3B-----CHECK HEAD IN ADJACENT HORIZONTAL CELLS TO SEE IF WETTING
C3B-----THRESHOLD HAS BEEN REACHED.
    2 IF(WETDRY(J,I,KB).LT.ZERO) GO TO 6                                AT 02E6
      IF(J.EQ.1)GO TO 3                                                 AT 0328
      HTMP=HNEW(J-1,I,K)                                                AT 0335
      IF(IBOUND(J-1,I,K).GT.0.AND.IBOUND(J-1,I,K).NE.30000.AND.         AT 0374
     1                          HTMP.GE.TURNON)GO TO 9
    3 IF(J.EQ.NCOL)GO TO 4                                              AT 03D1
      HTMP=HNEW(J+1,I,K)                                                AT 03E2
      IF(IBOUND(J+1,I,K).GT.0.AND.HTMP.GE.TURNON)GO TO 9                AT 041D
    4 IF(I.EQ.1)GO TO 5                                                 AT 046C
      HTMP=HNEW(J,I-1,K)                                                AT 0479
      IF(IBOUND(J,I-1,K).GT.0.AND.IBOUND(J,I-1,K).NE.30000.AND.         AT 04BB
     1                           HTMP.GE.TURNON)GO TO 9
    5 IF(I.EQ.NROW)GO TO 6                                              AT 051B
      HTMP=HNEW(J,I+1,K)                                                AT 052C
      IF(IBOUND(J,I+1,K).GT.0.AND.HTMP.GE.TURNON)GO TO 9                AT 0568
C
C3C-----CELL IS DRY AND STAYS DRY.  SET TRANSMISSIVITY TO 0, SET
C3C-----SATURATED THICKNESS (BUFF) TO 0, AND SKIP TO THE NEXT CELL.
    6 CC(J,I,K)=ZERO                                                    AT 05BA
C-Environment Agency (UK)-6.X & Y-dir transmissivities>>>>>>>>>>>>>>>>>>
      cr(j,i,k)=zero                                                    AT 05F9
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      IF(LAYAVG(K).EQ.30) BUFF(J,I,K)=ZERO                              AT 0635
      GO TO 200                                                         AT 067F
C
C4------CELL BECOMES WET.  SET INITIAL HEAD AND VERTICAL CONDUCTANCE.
    9 IF(IHDWET.NE.0) HNEW(J,I,K)=BOT(J,I,KB)+WETFCT*WD                 AT 0684
      IF(IHDWET.EQ.0) HNEW(J,I,K)=BOT(J,I,KB)+WETFCT*(HTMP-BOT(J,I,KB)) AT 06F7
      IF(K.EQ.NLAY) GO TO 12                                            AT 0771
      IF(IBOUND(J,I,K+1).NE.0) CV(J,I,K)= CVWD(J,I,K)                   AT 0781
   12 IF(K.EQ.1) GO TO 14                                               AT 07F4
      IF(IBOUND(J,I,K-1).NE.0) CV(J,I,K-1)= CVWD(J,I,K-1)               AT 0800
   14 IBOUND(J,I,K)=30000                                               AT 0876
C
C4A-----PRINT MESSAGE SAYING CELL HAS BEEN CONVERTED TO WET.
      NCNVRT=NCNVRT+1                                                   AT 08A9
      ICNVRT(NCNVRT)=I                                                  AT 08AF
      JCNVRT(NCNVRT)=J                                                  AT 08C2
      ACNVRT(NCNVRT)='WET'                                              AT 08CF
      IF(NCNVRT.LT.5) GO TO 20                                          AT 08E9
         IF(IHDCNV.EQ.0) WRITE(IOUT,17) KITER,K,KSTP,KPER               AT 08F6
   17    FORMAT(1X,/1X,'CELL CONVERSIONS FOR ITER.=',I3,'  LAYER=',     AT 0969
     1    I3,'  STEP=',I3,'  PERIOD=',I3,'   (ROW,COL)')
         IHDCNV=1                                                       AT 09F2
         WRITE(IOUT,18) (ACNVRT(L),ICNVRT(L),JCNVRT(L),L=1,NCNVRT)      AT 09FC
   18    FORMAT(1X,3X,5(A,'(',I3,',',I3,')   '))                        AT 0AAD
         NCNVRT=0                                                       AT 0AF4
C
C5------CALCULATE SATURATED THICKNESS.
   20 HD=HNEW(J,I,K)                                                    AT 0AFE
      BBOT=BOT(J,I,KB)                                                  AT 0B3D
C-Environment Agency (UK)-7.Explicit transmissivity calculation>>>>>>>>>
      if(ihold.eq.1.and.iss.eq.0)hd=hold(j,i,k)                         AT 0B79
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
```

```
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
c      IF(LAYCON(K).EQ.1) GO TO 50
      IF(LAYCON(K).EQ.1.or.laycon(k).eq.4) GO TO 50                           AT 0BCD
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      TTOP=TOP(J,I,KT)                                                        AT 0BEE
      IF(HD.GT.TTOP) HD=TTOP                                                  AT 0C2A
   50 THCK=HD-BBOT                                                            AT 0C58
C
C6------CHECK TO SEE IF SATURATED THICKNESS IS GREATER THAN ZERO.
      IF(THCK.LE.ZERO) GO TO 100                                              AT 0C6B
C
C6A-----IF SATURATED THICKNESS>0 THEN EITHER CALCULATE TRANSMISSIVITY
C6A-----AS HYDRAULIC CONDUCTIVITY TIMES SATURATED THICKNESS OR STORE
C6A-----K IN CC AND SATURATED THICKNESS IN BUFF.
C-Environment Agency (UK)-3.Auto-conversion for steady state VKD>>>>>>>>
c      IF(LAYAVG(K).EQ.30) THEN
      IF(LAYAVG(K).EQ.30.or.((laycon(k).eq.4.or.laycon(k).eq.5).and.          AT 0C7B
     1      iss.eq.2)) THEN
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
         CC(J,I,K)=HY(J,I,KB)                                                 AT 0CBB
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
c         BUFF(J,I,K)=THCK
c      ELSE
c         CC(J,I,K)=THCK*HY(J,I,KB)
c      END IF
c
c6a1----Store K in CC & CR and saturated thickness in BUFF
         if(itrpy.eq.2)then                                                   AT 0D21
           cr(j,i,k)=hy(j,i,kb)                                               AT 0D2D
           cc(j,i,k)=trpy(j,i,kb)                                             AT 0D93
         endif                                                                AT 0DF9
         if(layavg(k).eq.30)BUFF(J,I,K)=THCK                                  AT 0DF9
c
c6a2----Calculate transmissivity as hydraulic conductivity times
c6a2----saturated thickness.
      ELSEif(layavg(k).lt.30) then                                            AT 0E43
         if(itrpy.ne.2)then                                                   AT 0E5B
           CC(J,I,K)=THCK*HY(J,I,KB)                                          AT 0E67
         else                                                                 AT 0EC7
           cr(j,i,k)=thck*hy(j,i,kb)                                          AT 0ECC
           cc(j,i,k)=thck*trpy(j,i,kb)                                        AT 0F31
         endif                                                                AT 0F93
          if((laycon(k).eq.4.or.laycon(k).eq.5).and.hd.gt.vmid(j,i,kg)        AT 0F93
     1            .and.iss.ne.3)then
c
c6a3----Calculate transmissivity using VKD relationships.
           ekmax=vmid(j,i,kg)+((vkmax(j,i,kg)-1.0)/                           AT 1002
     1                 vkgrad(j,i,kg))
           if(hd.gt.ekmax)then                                                AT 1049
            if(itrpy.ne.2)then                                                AT 105B
              cc(j,i,k)=cc(j,i,k)+(((ekmax-vmid(j,i,kg))**2)                  AT 1067
     1            *vkgrad(j,i,kg)*hy(j,i,kb)/2)
     2            +hy(j,i,kb)*(vkmax(j,i,kg)-1.0)*(hd-ekmax)
            else                                                              AT 1190
              cr(j,i,k)=cr(j,i,k)+(((ekmax-vmid(j,i,kg))**2)                  AT 1195
     1            *vkgrad(j,i,kg)*hy(j,i,kb)/2)
     2            +hy(j,i,kb)*(vkmax(j,i,kg)-1.0)*(hd-ekmax)
              cc(j,i,k)=cc(j,i,k)+(((ekmax-vmid(j,i,kg))**2)                  AT 12BB
     1            *vkgrad(j,i,kg)*trpy(j,i,kb)/2)
     2            +trpy(j,i,kb)*(vkmax(j,i,kg)-1.0)*(hd-ekmax)
            endif                                                             AT 13E1
           else                                                               AT 13E1
            if(itrpy.ne.2)then                                                AT 13E6
              cc(j,i,k)=cc(j,i,k)+(((hd-vmid(j,i,kg))**2)*vkgrad(j,i,kg)      AT 13F2
     1            *hy(j,i,kb)/2)
            else                                                              AT 1498
              cr(j,i,k)=cr(j,i,k)+(((hd-vmid(j,i,kg))**2)*vkgrad(j,i,kg)      AT 149D
     1            *hy(j,i,kb)/2)
              cc(j,i,k)=cc(j,i,k)+(((hd-vmid(j,i,kg))**2)*vkgrad(j,i,kg)      AT 1543
     1            *trpy(j,i,kb)/2)
            endif                                                             AT 15E6
           endif                                                              AT 15E6
          endif                                                               AT 15E6
      else                                                                    AT 15E6
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-Environment Agency (UK)-5.Inter-nodal transmissivities>>>>>>>>>>>>>>>>
c
c6a4----if internodal transmissivity then calculate transmissivity for
c6a4----the right hand side and front cell faces.
c
c6a5----if not last column calculate transmissivity at right face.
        if(j.ne.ncol)then                                                     AT 15EB
```

```
            if(ibound(j+1,i,k).ne.0)then                                           AT 15FC
              jp=j+1                                                               AT 1634
            else                                                                   AT 163B
              go to 21                                                             AT 1640
            end if                                                                 AT 1645
         else                                                                      AT 1645
            go to 21                                                               AT 164A
         end if                                                                    AT 164F
           HD=(HNEW(J,I,K)+hnew(jp,i,k))/2                                         AT 164F
          if(ihold.eq.1.and.iss.eq.0)hd=(hold(j,i,k)+hold(jp,i,k))/2               AT 16A7
           BBOT=(BOT(J,I,KB)+bot(jp,i,kb))/2                                       AT 1711
           IF(LAYCON(K).EQ.1.or.laycon(k).eq.4) GO TO 501                          AT 1766
           TTOP=(TOP(J,I,KT)+top(jp,i,kt))/2                                       AT 1787
           IF(HD.GT.TTOP) HD=TTOP                                                  AT 17DC
 501       cr(j,i,k)=(HD-BBOT)*hy(j,i,kb)                                          AT 1806
           if(laycon(k).eq.4 .or. laycon(k).eq.5) then                             AT 1871
c
c6a6----Calculate transmissivity using VKD relationships.
             vvmid=(vmid(j,i,kg)+vmid(jp,i,kg))/2                                  AT 1892
             if(hd.gt.vvmid)then                                                   AT 18E7
               vvkgrad=(vkgrad(J,I,Kg)+vkgrad(jp,i,kg))/2                          AT 18F9
               vvkmax=(vkmax(J,I,Kg)+vkmax(jp,i,kg))/2                             AT 194B
               ekmax=vvmid+((vvkmax-1.0)/vvkgrad)                                  AT 199D
               if(hd.gt.ekmax)then                                                 AT 19B2
                 cr(j,i,k)=cr(j,i,k)+(((ekmax-vvmid)**2)*vvkgrad                   AT 19C4
     1             *hy(j,i,kb)/2)+hy(j,i,kb)*(vvkmax-1.0)*(hd-ekmax)
               else                                                                AT 1A63
                 cr(j,i,k)=cr(j,i,k)+(((hd-vvmid)**2)*vvkgrad                      AT 1A68
     1             *hy(j,i,kb)/2)
               endif                                                               AT 1AE7
             endif                                                                 AT 1AE7
           endif                                                                   AT 1AE7
c
c6a7----if not last row calculate transmissivity at front face.
 21      if(i.ne.nrow)then                                                         AT 1AE7
            if(ibound(j,i+1,k).ne.0)then                                           AT 1AF8
              ip=i+1                                                               AT 1B31
            else                                                                   AT 1B3E
              go to 200                                                            AT 1B43
            end if                                                                 AT 1B48
         else                                                                      AT 1B48
            go to 200                                                              AT 1B4D
         end if                                                                    AT 1B52
           HD=(HNEW(J,I,K)+hnew(j,ip,k))/2                                         AT 1B52
          if(ihold.eq.1.and.iss.eq.0)hd=(hold(j,i,k)+hold(j,ip,k))/2               AT 1BB7
           BBOT=(BOT(J,I,KB)+bot(j,ip,kb))/2                                       AT 1C2E
           IF(LAYCON(K).EQ.1.or.laycon(k).eq.4) GO TO 502                          AT 1C90
           TTOP=(TOP(J,I,KT)+top(j,ip,kt))/2                                       AT 1CB1
           IF(HD.GT.TTOP) HD=TTOP                                                  AT 1D13
 502       cc(j,i,k)=(HD-BBOT)*hy(j,i,kb)                                          AT 1D3D
           if(itrpy.eq.2) cc(j,i,k)=(HD-BBOT)*trpy(j,i,kb)                         AT 1DA8
           if(laycon(k).eq.4 .or. laycon(k).eq.5) then                             AT 1E1C
c
c6a8----Calculate transmissivity using VKD relationships.
             vvmid=(vmid(j,i,kg)+vmid(j,ip,kg))/2                                  AT 1E3D
             if(hd.gt.vvmid)then                                                   AT 1E9F
               vvkgrad=(vkgrad(j,i,kg)+vkgrad(j,ip,kg))/2                          AT 1EB1
               vvkmax=(vkmax(j,i,kg)+vkmax(j,ip,kg))/2                             AT 1F10
               ekmax=vvmid+((vvkmax-1.0)/vvkgrad)                                  AT 1F6F
               if(hd.gt.ekmax)then                                                 AT 1F84
                if(itrpy.ne.2)then                                                 AT 1F96
                  cc(j,i,k)=cc(j,i,k)+(((ekmax-vvmid)**2)*vvkgrad*                 AT 1FA2
     1                          hy(j,i,kb)/2)+hy(j,i,kb)*(vvkmax-1.0)
     2                          *(hd-ekmax)
                else                                                               AT 2041
                  cc(j,i,k)=cc(j,i,k)+(((ekmax-vvmid)**2)*vvkgrad*                 AT 2046
     1                          trpy(j,i,kb)/2)+trpy(j,i,kb)*(vvkmax-1.0)
     2                          *(hd-ekmax)
                endif                                                              AT 20E5
               else                                                                AT 20E5
                if(itrpy.ne.2)then                                                 AT 20EA
                  cc(j,i,k)=cc(j,i,k)+(((hd-vvmid)**2)*vvkgrad*                    AT 20F6
     1                          hy(j,i,kb)/2)
                else                                                               AT 2175
                  cc(j,i,k)=cc(j,i,k)+(((hd-vvmid)**2)*vvkgrad*                    AT 217A
     1                          trpy(j,i,kb)/2)
                endif                                                              AT 21F9
               endif                                                               AT 21F9
             endif                                                                 AT 21F9
           endif                                                                   AT 21F9
      end if                                                                       AT 21F9
```

```
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      GO TO 200                                                    AT 21F9
C
C6B-----WHEN SATURATED THICKNESS < 0, PRINT A MESSAGE AND SET
C6B-----TRANSMISSIVITY, IBOUND, AND VERTICAL CONDUCTANCE =0
  100 NCNVRT=NCNVRT+1                                              AT 21FE
      ICNVRT(NCNVRT)=I                                             AT 2204
      JCNVRT(NCNVRT)=J                                             AT 2217
      ACNVRT(NCNVRT)='DRY'                                         AT 2224
      IF(NCNVRT.LT.5) GO TO 150                                    AT 2238
         IF(IHDCNV.EQ.0) WRITE(IOUT,17) KITER,K,KSTP,KPER          AT 2245
         IHDCNV=1                                                  AT 22B8
         WRITE(IOUT,18) (ACNVRT(L),ICNVRT(L),JCNVRT(L),L=1,NCNVRT) AT 22C2
         NCNVRT=0                                                  AT 2373
  150 HNEW(J,I,K)=HDRY                                             AT 237D
      CC(J,I,K)=ZERO                                               AT 23B8
C-Environment Agency (UK)-5.Inter-nodal transmissivities>>>>>>>>>>>>>>>>
      cr(j,i,k)=zero                                               AT 23F7
c
c6c-----If inter-nodal transmissivities or Ks, then set all connecting
c6c-----transmissivities to zero
      if(layavg(k).eq.40)then                                      AT 2436
         jm=j                                                      AT 2449
         im=i                                                      AT 2455
         if(j.ne.1)jm=j-1                                          AT 2461
         if(i.ne.1)im=i-1                                          AT 2471
         cc(j,im,k)=zero                                           AT 248B
         cr(jm,i,k)=zero                                           AT 24CA
      end if                                                       AT 2506
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      IF(IBOUND(J,I,K).GE.0) GO TO 160                             AT 2506
         WRITE(IOUT,151)                                           AT 2542
  151    FORMAT(1X,/1X,'CONSTANT-HEAD CELL WENT DRY',              AT 2568
     1          ' -- SIMULATION ABORTED')
         WRITE(IOUT,152) K,I,J,KITER,KSTP,KPER                     AT 25BD
  152    FORMAT(1X,'LAYER=',I2,'   ROW=',I3,'   COLUMN=',I3,       AT 264B
     1    '   ITERATION=',I3,'   TIME STEP=',I3,'   STRESS PERIOD=',I3)
         STOP                                                      AT 26E0
  160 IBOUND(J,I,K)=0                                              AT 26F3
      IF(K.LT.NLAY) CV(J,I,K)=ZERO                                 AT 2726
      IF(K.GT.1) CV(J,I,K-1)=ZERO                                  AT 276D
  200 CONTINUE                                                     AT 27B6
C
C7------PRINT ANY REMAINING CELL CONVERSIONS NOT YET PRINTED
      IF(NCNVRT.EQ.0) GO TO 203                                    AT 27C0
         IF(IHDCNV.EQ.0) WRITE(IOUT,17) KITER,K,KSTP,KPER          AT 27CD
         IHDCNV=1                                                  AT 2840
         WRITE(IOUT,18) (ACNVRT(L),ICNVRT(L),JCNVRT(L),L=1,NCNVRT) AT 284A
         NCNVRT=0                                                  AT 28FB
C
C8------CHANGE IBOUND VALUE FOR CELLS THAT CONVERTED TO WET THIS
C8------ITERATION FROM 30000 to 1.
  203 IF(IWDFLG.EQ.0) GO TO 210                                    AT 2905
      DO 205 I=1,NROW                                              AT 2911
      DO 205 J=1,NCOL                                              AT 293E
      IF(IBOUND(J,I,K).EQ.30000) IBOUND(J,I,K)=1                   AT 296B
  205 CONTINUE                                                     AT 29CC
C
C9------COMPUTE HORIZONTAL BRANCH CONDUCTANCES FROM TRANSMISSIVITY.
C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
c  210 IF(LAYAVG(K).EQ.0) THEN
c         CALL SBCF5C(CR,CC,TRPY,DELR,DELC,K,NCOL,NROW,NLAY)
c      ELSE IF(LAYAVG(K).EQ.10) THEN
c         CALL SBCF5A(CR,CC,TRPY,DELR,DELC,K,NCOL,NROW,NLAY)
c      ELSE IF(LAYAVG(K).EQ.20) THEN
c         CALL SBCF5L(CR,CC,TRPY,DELR,DELC,K,NCOL,NROW,NLAY)
c      ELSE
c         CALL SBCF5U(CR,CC,TRPY,DELR,DELC,BUFF,K,NCOL,NROW,NLAY)
c      END IF
  210 IF(LAYAVG(K).EQ.0) THEN                                      AT 29D3
         CALL SBCF5C(CR,CC,TRPY,DELR,DELC,K,NCOL,NROW,NLAY,itrpy)  AT 29E6
      ELSE IF(LAYAVG(K).EQ.10) THEN                                AT 2A1B
         CALL SBCF5A(CR,CC,TRPY,DELR,DELC,K,NCOL,NROW,NLAY,itrpy)  AT 2A33
      ELSE IF(LAYAVG(K).EQ.20) THEN                                AT 2A68
         CALL SBCF5L(CR,CC,TRPY,DELR,DELC,K,NCOL,NROW,NLAY,itrpy)  AT 2A80
      ELSE if(layavg(k).eq.30) then                                AT 2AB5
         CALL SBCF5U(CR,CC,TRPY,DELR,DELC,BUFF,K,NCOL,NROW,NLAY,itrpy) AT 2ACD
      else                                                         AT 2B05
         call sbcf5i(CR,CC,TRPY,DELR,DELC,K,NCOL,NROW,NLAY,itrpy)  AT 2B0A
      END IF                                                       AT 2B3F
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
```

```
C10-----RETURN.
        RETURN                                                              AT 2B3F
        END                                                                 AT 2B44
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
c        SUBROUTINE SBCF5N(HNEW,IBOUND,SC1,SC2,CR,CC,CV,HY,TRPY,DELR,DELC,
c      1     ISS,NCOL,NROW,NLAY,IOUT,WETDRY,IWDFLG,CVWD)
COMMENT - HDA has not been used

        SUBROUTINE SBCF5N(HNEW,IBOUND,SC1,SC2,CR,CC,CV,HY,TRPY,DELR,DELC,
       1     ISS,NCOL,NROW,NLAY,IOUT,WETDRY,IWDFLG,CVWD,itrpy,vkgrad,vkmax,
       2     vsgrad,vsmax)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C-----VERSION 1456 29JUNE1993 SBCF5N
C     ****************************************************************
C     INITIALIZE AND CHECK BCF DATA
C     ****************************************************************
C
C        SPECIFICATIONS:
C     -----------------------------------------------------------------
C
        DOUBLE PRECISION HNEW,HCNV
C
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
c        DIMENSION HNEW(NCOL,NROW,NLAY),IBOUND(NCOL,NROW,NLAY)
c      1     ,SC1(NCOL,NROW,NLAY),CR(NCOL,NROW,NLAY)
c      2     ,CC(NCOL,NROW,NLAY),CV(NCOL,NROW,NLAY)
c      3     ,HY(NCOL,NROW,NLAY),TRPY(NLAY),DELR(NCOL),DELC(NROW)
c      4     ,SC2(NCOL,NROW,NLAY),WETDRY(NCOL,NROW,NLAY)
c      5     ,CVWD(NCOL,NROW,NLAY)
        DIMENSION HNEW(NCOL,NROW,NLAY),IBOUND(NCOL,NROW,NLAY)
       1     ,SC1(NCOL,NROW,NLAY),CR(NCOL,NROW,NLAY)
       2     ,CC(NCOL,NROW,NLAY),CV(NCOL,NROW,NLAY)
       3     ,HY(NCOL,NROW,NLAY),TRPY(ncol,nrow,NLAY),DELR(NCOL),DELC(NROW)
       4     ,SC2(NCOL,NROW,NLAY),WETDRY(NCOL,NROW,NLAY)
       5     ,CVWD(NCOL,NROW,NLAY),vkgrad(ncol,nrow,nlay)
       6     ,vkmax(ncol,nrow,nlay),vsgrad(ncol,nrow,nlay)
       7     ,vsmax(ncol,nrow,nlay)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
        COMMON /FLWCOM/LAYCON(200)
        COMMON /FLWAVG/LAYAVG(200)
C     -----------------------------------------------------------------
C
C1------MULTIPLY VERTICAL LEAKANCE BY AREA TO MAKE CONDUCTANCE.
        ZERO=0.                                                             AT 0061
        IF(NLAY.EQ.1) GO TO 20                                              AT 0075
        K1=NLAY-1                                                           AT 007E
        DO 10 K=1,K1                                                        AT 0085
        DO 10 I=1,NROW                                                      AT 00AD
        DO 10 J=1,NCOL                                                      AT 00DA
        CV(J,I,K)=CV(J,I,K)*DELR(J)*DELC(I)                                 AT 0107
     10 CONTINUE                                                            AT 0143
C
C2------IF WETTING CAPABILITY IS ACTIVATED, SAVE CV IN CVWD FOR USE WHEN
C2------WETTING CELLS.
        IF(IWDFLG.EQ.0) GO TO 20                                            AT 014F
        DO 15 K=1,K1                                                        AT 015B
        DO 15 I=1,NROW                                                      AT 0189
        DO 15 J=1,NCOL                                                      AT 01B6
        CVWD(J,I,K)=CV(J,I,K)                                               AT 01E3
     15 CONTINUE                                                            AT 0215
C
C3------IF IBOUND=0, SET CV=0 AND CC=0.
     20 DO 30 K=1,NLAY                                                      AT 021E
        DO 30 I=1,NROW                                                      AT 024B
        DO 30 J=1,NCOL                                                      AT 0278
        IF(IBOUND(J,I,K).NE.0) GO TO 30                                     AT 02A5
        IF(K.NE.NLAY) CV(J,I,K)=ZERO                                        AT 02D5
        IF(K.NE.1) CV(J,I,K-1)=ZERO                                         AT 0315
        CC(J,I,K)=ZERO                                                      AT 035A
     30 CONTINUE                                                            AT 0391
C
C4------INSURE THAT EACH ACTIVE CELL HAS AT LEAST ONE NON-ZERO
C4------TRANSMISSIVE PARAMETER.
        HCNV=888.88                                                         AT 03A0
        KB=0                                                                AT 03B4
        DO 60 K=1,NLAY                                                      AT 03BE
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
c        IF(LAYCON(K).EQ.1 .OR. LAYCON(K).EQ.3) GO TO 50
```

```
   2010.01        IF(LAYCON(K).EQ.1 .OR. LAYCON(K).EQ.3 .or. laycon(k).eq.4 .or.              AT 03EB
   2011.01       1      laycon(k).eq.5) GO TO 50
   2012.01  C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
   2013.01  C
   2014.01  C4A-----WHEN LAYER TYPE IS 0 OR 2, TRANSMISSIVITY OR CV MUST BE NONZERO.
   2015.01        DO 45 I=1,NROW                                                              AT 0423
   2016.02        DO 45 J=1,NCOL                                                              AT 0450
   2017.03        IF(IBOUND(J,I,K).EQ.0) GO TO 45                                             AT 047D
   2018.03        IF(CC(J,I,K).NE.ZERO) GO TO 45                                              AT 04AD
   2019.03  C-Environment Agency (UK)-5.Inter-nodal transmissivities>>>>>>>>>>>>>>>>
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
   2020.03        if(itrpy.eq.2.and.cr(j,i,k).ne.zero)goto 45                                 AT 04E6
   2021.03  c
   2022.03  c4a2----If using inter-nodal Ts, check all connecting Ts.
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
   2023.03        if(layavg(K).eq.40)then                                                     AT 052F
   2024.04          if(itrpy.eq.2)then                                                        AT 0543
   2025.05            if(j.ne.1)then                                                          AT 054F
   2026.06              if(cr(j-1,i,k).ne.zero)goto 45                                        AT 055C
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
   2027.06            endif                                                                   AT 0595
   2028.05            if(i.ne.1)then                                                          AT 0595
   2029.06              if(cc(j,i-1,k).ne.zero)goto 45                                        AT 05A2
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
   2030.06            endif                                                                   AT 05E2
   2031.05          elseif(itrpy.eq.0)then                                                    AT 05E2
   2032.05            if(j.ne.1)then                                                          AT 05F3
   2033.06              if(cc(j-1,i,k).ne.zero)goto 45                                        AT 0600
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
   2034.06            endif                                                                   AT 063D
   2035.05            if(i.ne.1)then                                                          AT 063D
   2036.06              if(cc(j,i-1,k).ne.zero.and.trpy(k,1,1).ne.zero)goto 45                AT 064A
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
   2037.06            endif                                                                   AT 06BB
   2038.05          else                                                                      AT 06BB
   2039.05            if(j.ne.1)then                                                          AT 06C0
   2040.06              if(cc(j-1,i,k).ne.zero)goto 45                                        AT 06CD
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
   2041.06            endif                                                                   AT 070A
   2042.05            if(i.ne.1)then                                                          AT 070A
   2043.06              if(cc(j,i-1,k).ne.zero.and.trpy(j,i-1,k).ne.zero)goto 45              AT 0717
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
   2044.06            endif                                                                   AT 076E
   2045.05          endif                                                                     AT 076E
   2046.04        endif                                                                       AT 076E
   2047.04  C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
   2048.03        IF(K.EQ.NLAY) GO TO 41                                                      AT 076E
   2049.03        IF(CV(J,I,K).NE.ZERO) GO TO 45                                              AT 077F
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
   2050.03     41 IF(K.EQ.1) GO TO 42                                                         AT 07B8
   2051.03        IF(CV(J,I,K-1).NE.ZERO) GO TO 45                                            AT 07C5
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
   2052.03     42 IBOUND(J,I,K)=0                                                             AT 0803
   2053.03        HNEW(J,I,K)=HCNV                                                            AT 082E
   2054.03        WRITE(IOUT,43) K,I,J                                                        AT 0865
   2055.03     43 FORMAT(1X,'NODE (LAYER,ROW,COL)',3I4,                                       AT 08C7
   2056.03       1      ' ELIMINATED BECAUSE ALL CONDUCTANCES TO NODE ARE 0')
   2057.03     45 CONTINUE                                                                    AT 092E
   2058.01        GO TO 60                                                                    AT 0938
   2059.01  C
   2060.01  C4B-----WHEN LAYER TYPE IS 1, 3, 4 or 5, HY OR CV MUST BE NONZERO.
   2061.01     50 KB=KB+1                                                                     AT 093D
   2062.01        DO 59 I=1,NROW                                                              AT 0943
   2063.02        DO 59 J=1,NCOL                                                              AT 0970
   2064.02  C
   2065.02  C4B1----IF WETTING CAPABILITY IS ACTIVE, CHECK CVWD.
   2066.03        IF(IWDFLG.EQ.0) GO TO 55                                                    AT 099D
   2067.03        IF(WETDRY(J,I,KB).EQ.ZERO) GO TO 55                                         AT 09A9
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
   2068.03        IF(K.EQ.NLAY) GO TO 51                                                      AT 09E2
   2069.03        IF(CVWD(J,I,K).NE.ZERO) GO TO 59                                            AT 09F3
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
   2070.03     51 IF(K.EQ.1) GO TO 57                                                         AT 0A2C
   2071.03        IF(CVWD(J,I,K-1).NE.ZERO) GO TO 59                                          AT 0A39
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
   2072.03        GO TO 57                                                                    AT 0A77
   2073.03  C
   2074.03  C4B2----WETTING CAPABILITY IS INACTIVE, SO CHECK CV AT ACTIVE CELLS.
   2075.03     55 IF(IBOUND(J,I,K).EQ.0) GO TO 59                                             AT 0A7C
   2076.03        IF(K.EQ.NLAY) GO TO 56                                                      AT 0AB0
```

```
      IF(CV(J,I,K).NE.ZERO) GO TO 59                                        AT 0AC1
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
   56 IF(K.EQ.1) GO TO 57                                                   AT 0AFA
      IF(CV(J,I,K-1).NE.ZERO) GO TO 59                                      AT 0B07
C
C4B3----CHECK HYDRAULIC CONDUCTIVITY.
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
   57 IF(HY(J,I,KB).NE.ZERO) GO TO 59                                       AT 0B45
C-Environment Agency (UK)-5.Inter-nodal transmissivities>>>>>>>>>>>>>>>>
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
      if(itrpy.eq.2.and.trpy(j,i,kb).ne.zero)goto 59                        AT 0B82
c
c4b3i---If using inter-nodal Ks, check all connecting Ks.
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
      if(layavg(K).eq.40)then                                               AT 0BCB
        if(j.ne.1)then                                                      AT 0BDF
          if(hy(j-1,i,k).ne.zero)goto 59                                    AT 0BEC
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
        endif                                                               AT 0C25
        if(itrpy.eq.2)then                                                  AT 0C25
          if(i.ne.1)then                                                    AT 0C31
            if(trpy(j,i-1,kb).ne.zero)goto 59                               AT 0C3E
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
          endif                                                             AT 0C7E
        elseif(itrpy.eq.0)then                                              AT 0C7E
          if(i.ne.1)then                                                    AT 0C8F
            if(hy(j,i-1,k).ne.zero.and.trpy(k,1,1).ne.zero)goto 59          AT 0C9C
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
          endif                                                             AT 0D0D
        else                                                                AT 0D0D
          if(i.ne.1)then                                                    AT 0D12
            if(hy(j,i-1,k).ne.zero.and.trpy(j,i-1,k).ne.zero)goto 59        AT 0D1F
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
          endif                                                             AT 0D76
        endif                                                               AT 0D76
      endif                                                                 AT 0D76
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C4B4----HY AND CV ARE ALL 0, SO CONVERT CELL TO NO FLOW.
      IBOUND(J,I,K)=0                                                       AT 0D76
      HNEW(J,I,K)=HCNV                                                      AT 0DA1
      IF(IWDFLG.NE.0) WETDRY(J,I,KB)=ZERO                                   AT 0DD8
      WRITE(IOUT,43) K,I,J                                                  AT 0E1B
   59 CONTINUE                                                              AT 0E7D
   60 CONTINUE                                                              AT 0E87
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
C
C4C-----CHECK FOR VKGRAD = 0, & REPLACE WITH VKMAX = 1.
      kg=0                                                                  AT 0E8C
      do 110 k=1,nlay                                                       AT 0E96
        if(laycon(k).eq.4 .or. laycon(k).eq.5)then                          AT 0EC3
          kg=kg+1                                                           AT 0EDF
          do 120 i=1,nrow                                                   AT 0EE5
          do 120 j=1,ncol                                                   AT 0F12
            if(vkgrad(j,i,kg).le.0.0.and.ibound(j,i,k).ne.0)then            AT 0F3F
              vkgrad(j,i,kg)=1.0                                            AT 0FA8
              vkmax(j,i,kg)=1.0                                             AT 0FD8
              write(iout,143)K,I,J                                          AT 100B
            endif                                                           AT 106D
 120      continue                                                          AT 106D
        endif                                                               AT 1077
 110  continue                                                              AT 1077
 143  format(' K GRAD FACTOR & MAX FACTOR IN NODE (LAYER,ROW,COL)',3I4,     AT 107C
     1      ' SET TO 1.0 BECAUSE GRAD FACTOR WAS <= 0.0')
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-Environment Agency (UK)-2.Variable storage with depth (VSD)>>>>>>>>>>>
C
C4C-----CHECK FOR VSGRAD = 0, & REPLACE WITH VSMAX = 1.
      if(iss.eq.0)then                                                      AT 10F7
        kg=0                                                                AT 1103
        do 111 k=1,nlay                                                     AT 110D
          if(laycon(k).eq.4 .or. laycon(k).eq.5)then                        AT 113A
            kg=kg+1                                                         AT 1156
            do 121 i=1,nrow                                                 AT 115C
            do 121 j=1,ncol                                                 AT 1189
              if(vsgrad(j,i,kg).le.0.0.and.ibound(j,i,k).ne.0)then          AT 11B6
                vsgrad(j,i,kg)=1.0                                          AT 121F
                vsmax(j,i,kg)=1.0                                           AT 124F
                write(iout,144)K,I,J                                        AT 1282
              endif                                                         AT 12E4
```

```
 121       continue                                                 AT 12E4
         endif                                                      AT 12EE
 111    continue                                                    AT 12EE
       endif                                                        AT 12F3
 144  format(' S GRAD FACTOR & MAX FACTOR IN NODE (LAYER,ROW,COL)',3I4, AT 12F3
     1       ' SET TO 1.0 BECAUSE GRAD FACTOR WAS <= 0.0')
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C5------CALCULATE HOR. CONDUCTANCE(CR AND CC) FOR CONSTANT T LAYERS.
      DO 70 K=1,NLAY                                                AT 136E
      KK=K                                                          AT 139B
C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
c      IF(LAYCON(K).EQ.3 .OR. LAYCON(K).EQ.1) GO TO 70
c      IF(LAYAVG(K).EQ.0) THEN
c         CALL SBCF5C(CR,CC,TRPY,DELR,DELC,KK,NCOL,NROW,NLAY)
c      ELSE IF(LAYAVG(K).EQ.10) THEN
c         CALL SBCF5A(CR,CC,TRPY,DELR,DELC,KK,NCOL,NROW,NLAY)
c      ELSE
c         CALL SBCF5L(CR,CC,TRPY,DELR,DELC,KK,NCOL,NROW,NLAY)
c      END IF
      IF(LAYCON(K).EQ.3 .OR. LAYCON(K).EQ.1 .or. laycon(k).eq.4 .or.  AT 13A1
     1       laycon(k).eq.5) GO TO 70
      IF(LAYAVG(K).EQ.0) THEN                                       AT 13D9
         CALL SBCF5C(CR,CC,TRPY,DELR,DELC,KK,NCOL,NROW,NLAY,itrpy)  AT 13E7
      ELSE IF(LAYAVG(K).EQ.10) THEN                                 AT 1411
         CALL SBCF5A(CR,CC,TRPY,DELR,DELC,KK,NCOL,NROW,NLAY,itrpy)  AT 142A
      ELSE if(layavg(k).eq.20) then                                 AT 1454
         CALL SBCF5L(CR,CC,TRPY,DELR,DELC,KK,NCOL,NROW,NLAY,itrpy)  AT 146D
      else                                                          AT 1497
        if(itrpy.eq.2)goto 66                                       AT 149C
         do 65 i=1,nrow                                             AT 14A8
         do 65 j=1,ncol                                             AT 14D5
            cr(j,i,k)=cc(j,i,k)                                     AT 1502
   65 continue                                                      AT 1534
   66 continue                                                      AT 1538
         call sbcf5i(CR,CC,TRPY,DELR,DELC,K,NCOL,NROW,NLAY,itrpy)   AT 1538
      END IF                                                        AT 1562
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
   70 CONTINUE                                                      AT 1562
C
C6------IF TRANSIENT, LOOP THROUGH LAYERS AND CALCULATE STORAGE
C6------CAPACITY.
      IF(ISS.NE.0) GO TO 100                                        AT 1567
      KT=0                                                          AT 1573
      DO 90 K=1,NLAY                                                AT 157D
C
C6A-----MULTIPLY PRIMARY STORAGE COEFFICIENT BY DELR & DELC TO GET
C6A-----PRIMARY STORAGE CAPACITY.
      DO 80 I=1,NROW                                                AT 15AA
      DO 80 J=1,NCOL                                                AT 15D7
      SC1(J,I,K)=SC1(J,I,K)*DELR(J)*DELC(I)                         AT 1604
   80 CONTINUE                                                      AT 1640
C
C6B-----IF LAYER IS CONF/UNCONF MULTIPLY SECONDARY STORAGE COEFFICIENT
C6B-----BY DELR AND DELC TO GET SECONDARY STORAGE CAPACITY(SC2).
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
c      IF(LAYCON(K).NE.3 .AND. LAYCON(K).NE.2) GO TO 90
      IF(LAYCON(K).NE.3.AND.LAYCON(K).NE.2.and.laycon(k).ne.5) GO TO 90  AT 1647
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      KT=KT+1                                                       AT 1677
      DO 85 I=1,NROW                                                AT 167D
      DO 85 J=1,NCOL                                                AT 16AA
      SC2(J,I,KT)=SC2(J,I,KT)*DELR(J)*DELC(I)                       AT 16D7
   85 CONTINUE                                                      AT 1713
   90 CONTINUE                                                      AT 171A
C
C7------RETURN.
  100 RETURN                                                        AT 171F
      END                                                           AT 1724
C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
c      SUBROUTINE SBCF5A(CR,CC,TRPY,DELR,DELC,K,NCOL,NROW,NLAY)


      SUBROUTINE SBCF5A(CR,CC,TRPY,DELR,DELC,K,NCOL,NROW,NLAY,itrpy)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C-----VERSION 02JULY1993 SBCF5A
C     ******************************************************************
C-------COMPUTE CONDUCTANCE USING ARITHMETIC MEAN TRANSMISSIVITY
C-------ACTIVATED BY LAYAVG=10
C     ******************************************************************
```

```
C
C      SPECIFICATIONS:
C     ------------------------------------------------------------------
C
C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
c      DIMENSION CR(NCOL,NROW,NLAY), CC(NCOL,NROW,NLAY)
c     2   , TRPY(NLAY), DELR(NCOL), DELC(NROW)
      DIMENSION CR(NCOL,NROW,NLAY), CC(NCOL,NROW,NLAY)
     2   , TRPY(ncol,nrow,NLAY), DELR(NCOL), DELC(NROW)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C     ------------------------------------------------------------------
      ZERO=0.                                                          AT 0058
C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
c      YX=TRPY(K)
      if(itrpy.eq.0)YX=TRPY(K,1,1)                                     AT 0066
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C1------FOR EACH CELL CALCULATE BRANCH CONDUCTANCES FROM THAT CELL
C1------TO THE ONE ON THE RIGHT AND THE ONE IN FRONT.
      DO 40 I=1,NROW                                                   AT 0093
      DO 40 J=1,NCOL                                                   AT 00B1
      T1=CC(J,I,K)                                                     AT 00CF
C
C2------IF T=0 THEN SET CONDUCTANCE EQUAL TO 0. GO ON TO NEXT CELL.
C-Environment Agency (UK)-6.X & Y-dir transmissivities>>>>>>>>>>>>>>>>>>
c      IF(T1.NE.ZERO) GO TO 10
      IF(T1.NE.ZERO.or.itrpy.eq.2) GO TO 10                            AT 00F5
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
      CR(J,I,K)=ZERO                                                   AT 0111
      GO TO 40                                                         AT 0135
C
C3------IF THIS IS NOT THE LAST COLUMN(RIGHTMOST) THEN CALCULATE
C3------BRANCH CONDUCTANCE IN THE ROW DIRECTION (CR) TO THE RIGHT.
   10 IF(J.EQ.NCOL) GO TO 30                                           AT 013A
      T2=CC(J+1,I,K)                                                   AT 0148
C-Environment Agency (UK)-6.X & Y-dir transmissivities>>>>>>>>>>>>>>>>>>
c      IF(T2.EQ.ZERO) THEN
      if(itrpy.eq.2)then                                               AT 0171
        t1=cr(j,i,k)                                                   AT 017D
        t2=cr(j+1,i,k)                                                 AT 01A1
      endif                                                            AT 01C5
C3A-----ARITHMETIC MEAN INTERBLOCK TRANSMISSIVITY
      IF(T2.EQ.ZERO.or.t1.eq.zero) THEN                                AT 01C5
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
         CR(J,I,K)=ZERO                                                AT 01E8
      ELSE                                                             AT 020F
         CR(J,I,K)=DELC(I)*(T1+T2)/(DELR(J+1)+DELR(J))                 AT 0214
      END IF                                                           AT 0255
C
C4------IF THIS IS NOT THE LAST ROW(FRONTMOST) THEN CALCULATE
C4------BRANCH CONDUCTANCE IN THE COLUMN DIRECTION (CC) TO THE FRONT.
   30 IF(I.EQ.NROW) GO TO 40                                           AT 0255
      T2=CC(J,I+1,K)                                                   AT 0263
      IF(T2.EQ.ZERO) THEN                                              AT 028D
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
         CC(J,I,K)=ZERO                                                AT 02A0
      ELSE                                                             AT 02C1
C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
c         CC(J,I,K)=YX*DELR(J)*(T1+T2)/(DELC(I+1)+DELC(I))
         if(itrpy.eq.0)then                                            AT 02C6
           CC(J,I,K)=YX*DELR(J)*(T1+T2)/(DELC(I+1)+DELC(I))            AT 02D2
         else                                                          AT 0316
           t1=cc(j,i,k)                                                AT 031B
           t2=cc(j,i+1,k)                                              AT 0342
           if(itrpy.ne.2)then                                          AT 0369
             t1=t1*trpy(j,i,k)                                         AT 0375
             t2=t2*trpy(j,i+1,k)                                       AT 0396
           endif                                                       AT 03BD
           if(t1.eq.zero.or.t2.eq.zero)then                            AT 03BD
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
             cc(j,i,k)=zero                                            AT 03E0
             goto 40                                                   AT 0407
           endif                                                       AT 040C
           CC(J,I,K)=DELR(J)*(T1+T2)/(DELC(I+1)+DELC(I))               AT 040C
         endif                                                         AT 044D
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      END IF                                                           AT 044D
```

```
   40 CONTINUE                                                            AT 044D
C
C5------RETURN
      RETURN                                                              AT 0457
      END                                                                 AT 045C
C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
c      SUBROUTINE SBCF5L(CR,CC,TRPY,DELR,DELC,K,NCOL,NROW,NLAY)


      SUBROUTINE SBCF5L(CR,CC,TRPY,DELR,DELC,K,NCOL,NROW,NLAY,itrpy)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C-----VERSION 02JULY1993 SBCF5L
C     ******************************************************************
C-------COMPUTE CONDUCTANCE USING LOGARITHMIC MEAN TRANSMISSIVITY
C-------ACTIVATED BY LAYAVG=20
C     ******************************************************************
C
C      SPECIFICATIONS:
C     ------------------------------------------------------------------
C
C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
c      DIMENSION CR(NCOL,NROW,NLAY), CC(NCOL,NROW,NLAY)
c     2   , TRPY(NLAY), DELR(NCOL), DELC(NROW)
      DIMENSION CR(NCOL,NROW,NLAY), CC(NCOL,NROW,NLAY)
     2   , TRPY(ncol,nrow,NLAY), DELR(NCOL), DELC(NROW)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C     ------------------------------------------------------------------
      ZERO=0.                                                             AT 0058
      TWO=2.                                                              AT 0066
      HALF=0.5                                                            AT 0074
      FRAC1=1.005                                                         AT 0082
      FRAC2=0.995                                                         AT 0090
C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
c      YX=TRPY(K)*TWO
      if(itrpy.eq.0)YX=TRPY(K,1,1)*TWO                                    AT 009E
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C1------FOR EACH CELL CALCULATE BRANCH CONDUCTANCES FROM THAT CELL
C1------TO THE ONE ON THE RIGHT AND THE ONE IN FRONT.
      DO 40 I=1,NROW                                                      AT 00CB
      DO 40 J=1,NCOL                                                      AT 00F8
      T1=CC(J,I,K)                                                        AT 0125
C
C2------IF T=0 THEN SET CONDUCTANCE EQUAL TO 0. GO ON TO NEXT CELL.
C-Environment Agency (UK)-6.X & Y-dir transmissivities>>>>>>>>>>>>>>>>>>
c      IF(T1.NE.ZERO) GO TO 10
      IF(T1.NE.ZERO.or.itrpy.eq.2) GO TO 10                               AT 0154
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
      CR(J,I,K)=ZERO                                                      AT 0173
      GO TO 40                                                            AT 019D
C
C3------IF THIS IS NOT THE LAST COLUMN(RIGHTMOST) THEN CALCULATE
C3------BRANCH CONDUCTANCE IN THE ROW DIRECTION (CR) TO THE RIGHT.
   10 IF(J.EQ.NCOL) GO TO 30                                              AT 01A2
      T2=CC(J+1,I,K)                                                      AT 01B3
C-Environment Agency (UK)-6.X & Y-dir transmissivities>>>>>>>>>>>>>>>>>>
c      IF(T2.EQ.ZERO) THEN
      if(itrpy.eq.2)then                                                  AT 01E2
        t1=cr(j,i,k)                                                      AT 01EE
        t2=cr(j+1,i,k)                                                    AT 021E
      endif                                                               AT 024B
      IF(T2.EQ.ZERO.or.t1.eq.zero) THEN                                   AT 024B
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C3A-----SET TO ZERO AND EXIT IF T2 IS ZERO
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
         CR(J,I,K)=ZERO                                                   AT 0271
         GO TO 30                                                         AT 029E
      END IF                                                              AT 02A3
C3B-----LOGARITHMIC MEAN INTERBLOCK TRANSMISSIVITY
      RATIO=T2/T1                                                         AT 02A3
      IF(RATIO.GT.FRAC1.OR.RATIO.LT.FRAC2) THEN                           AT 02B6
         T=(T2-T1)/LOG(RATIO)                                             AT 02D6
      ELSE                                                                AT 02F6
         T=HALF*(T1+T2)                                                   AT 02FB
      END IF                                                              AT 0311
      CR(J,I,K)=TWO*DELC(I)*T/(DELR(J+1)+DELR(J))                         AT 0311
C
```

```
C4------IF THIS IS NOT THE LAST ROW(FRONTMOST) THEN CALCULATE
C4------BRANCH CONDUCTANCE IN THE COLUMN DIRECTION (CC) TO THE FRONT.
   30 IF(I.EQ.NROW) GO TO 40                                              AT 035E
      T2=CC(J,I+1,K)                                                      AT 036F
      IF(T2.EQ.ZERO) THEN                                                 AT 039F
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
         CC(J,I,K)=ZERO                                                   AT 03B2
         GO TO 40                                                         AT 03D9
      END IF                                                              AT 03DE
C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
c      RATIO=T2/T1
c      IF(RATIO.GT.FRAC1.OR.RATIO.LT.FRAC2) THEN
c         T=(T2-T1)/LOG(RATIO)
c      ELSE
c         T=HALF*(T1+T2)
c      END IF
c      CC(J,I,K)=YX*DELR(J)*T/(DELC(I+1)+DELC(I))
       if(itrpy.eq.0)then                                                 AT 03DE
         RATIO=T2/T1                                                      AT 03EA
         IF(RATIO.GT.FRAC1.OR.RATIO.LT.FRAC2) THEN                        AT 03FD
            T=(T2-T1)/LOG(RATIO)                                          AT 041D
         ELSE                                                             AT 043D
            T=HALF*(T1+T2)                                                AT 0442
         END IF                                                           AT 0458
         CC(J,I,K)=YX*DELR(J)*T/(DELC(I+1)+DELC(I))                       AT 0458
       else                                                               AT 04A8
         t1=cc(j,i,k)                                                     AT 04AD
         t2=cc(j,i+1,k)                                                   AT 04E0
         if(itrpy.ne.2)then                                               AT 0510
           t1=t1*trpy(j,i,k)                                              AT 051C
           t2=t2*trpy(j,i+1,k)                                            AT 0549
         endif                                                            AT 0576
         if(t1.eq.zero.or.t2.eq.zero)then                                 AT 0576
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
           cc(j,i,k)=zero                                                 AT 059C
           goto 40                                                        AT 05C9
         endif                                                            AT 05CE
         ratio=t2/t1                                                      AT 05CE
         if(ratio.gt.frac1.or.ratio.lt.frac2) then                        AT 05E1
            t=(t2-t1)/log(ratio)                                          AT 0601
         else                                                             AT 0621
            t=half*(t1+t2)                                                AT 0626
         end if                                                           AT 063C
         cc(j,i,k)=2*delr(j)*t/(delc(i+1)+delc(i))                        AT 063C
       endif                                                              AT 068C
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
   40 CONTINUE                                                            AT 068C
C
C5------RETURN
      RETURN                                                              AT 0696
      END                                                                 AT 069B
C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
c      SUBROUTINE SBCF5U(CR,CC,TRPY,DELR,DELC,BUFF,K,NCOL,NROW,NLAY)



      SUBROUTINE SBCF5U(CR,CC,TRPY,DELR,DELC,BUFF,K,NCOL,NROW,NLAY,
     1 itrpy)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C-----VERSION 02JULY1993 SBCF5U
C     ******************************************************************
C-------COMPUTE CONDUCTANCE USING ARITHMETIC MEAN SATURATED THICKNESS
C-------AND LOGARITHMIC MEAN HYDRAULIC CONDUCTIVITY
C-------NODE HYDRAULIC CONDUCTIVITY IS IN CC,
C-------NODE SATURATED THICKNESS IS IN BUFF
C-------ACTIVATED BY LAYAVG=30
C     ******************************************************************
C
C      SPECIFICATIONS:
C     ------------------------------------------------------------------
C
C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
c      DIMENSION CR(NCOL,NROW,NLAY), CC(NCOL,NROW,NLAY)
c     2   , TRPY(NLAY), DELR(NCOL), DELC(NROW)
c     3   , BUFF(NCOL,NROW,NLAY)
      DIMENSION CR(NCOL,NROW,NLAY), CC(NCOL,NROW,NLAY)
     2   , TRPY(ncol,nrow,NLAY), DELR(NCOL), DELC(NROW)
     3   , BUFF(NCOL,NROW,NLAY)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
```

```
C     ------------------------------------------------------------------
         ZERO=0.                                                          AT 0058
         HALF=0.5                                                         AT 0066
         FRAC1=1.005                                                      AT 0074
         FRAC2=0.995                                                      AT 0082
C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
c         YX=TRPY(K)
         if(itrpy.eq.0)YX=TRPY(K,1,1)                                     AT 0090
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C1------FOR EACH CELL CALCULATE BRANCH CONDUCTANCES FROM THAT CELL
C1------TO THE ONE ON THE RIGHT AND THE ONE IN FRONT.
         DO 40 I=1,NROW                                                   AT 00C0
         DO 40 J=1,NCOL                                                   AT 00ED
         T1=CC(J,I,K)                                                     AT 011A
C
C2------IF T=0 THEN SET CONDUCTANCE EQUAL TO 0. GO ON TO NEXT CELL.
C-Environment Agency (UK)-6.X & Y-dir transmissivities>>>>>>>>>>>>>>>>>>
c         IF(T1.NE.ZERO) GO TO 10
         IF(T1.NE.ZERO.or.itrpy.eq.2) GO TO 10                            AT 0149
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
         CR(J,I,K)=ZERO                                                   AT 0168
         GO TO 40                                                         AT 0192
C
C3------IF THIS IS NOT THE LAST COLUMN(RIGHTMOST) THEN CALCULATE
C3------BRANCH CONDUCTANCE IN THE ROW DIRECTION (CR) TO THE RIGHT.
      10 IF(J.EQ.NCOL) GO TO 30                                           AT 0197
         T2=CC(J+1,I,K)                                                   AT 01A8
C-Environment Agency (UK)-6.X & Y-dir transmissivities>>>>>>>>>>>>>>>>>>
c         IF(T2.EQ.ZERO) THEN
         if(itrpy.eq.2)then                                               AT 01D7
           t1=cr(j,i,k)                                                   AT 01E3
           t2=cr(j+1,i,k)                                                 AT 0213
         endif                                                            AT 0240
         IF(T2.EQ.ZERO.or.t1.eq.zero) THEN                                AT 0240
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C3A-----SET TO ZERO AND EXIT IF T2 IS ZERO
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
            CR(J,I,K)=ZERO                                                AT 0266
            GO TO 30                                                      AT 0293
         END IF                                                           AT 0298
C3B-----LOGARITHMIC MEAN HYDRAULIC CONDUCTIVITY
         RATIO=T2/T1                                                      AT 0298
         IF(RATIO.GT.FRAC1.OR.RATIO.LT.FRAC2) THEN                        AT 02AB
            T=(T2-T1)/LOG(RATIO)                                          AT 02CB
         ELSE                                                             AT 02EB
            T=HALF*(T1+T2)                                                AT 02F0
         END IF                                                           AT 0306
C3C-----MULTIPLY LOGARITHMIC K BY ARITHMETIC SAT THICK
         CR(J,I,K)=DELC(I)*T*(BUFF(J,I,K)+BUFF(J+1,I,K))                  AT 0306
        *               /(DELR(J+1)+DELR(J))
C
C4------IF THIS IS NOT THE LAST ROW(FRONTMOST) THEN CALCULATE
C4------BRANCH CONDUCTANCE IN THE COLUMN DIRECTION (CC) TO THE FRONT.
      30 IF(I.EQ.NROW) GO TO 40                                           AT 035B
         T2=CC(J,I+1,K)                                                   AT 036C
         IF(T2.EQ.ZERO) THEN                                              AT 039C
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
            CC(J,I,K)=ZERO                                                AT 03AF
            GO TO 40                                                      AT 03D6
         END IF                                                           AT 03DB
C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
c         RATIO=T2/T1
c         IF(RATIO.GT.FRAC1.OR.RATIO.LT.FRAC2) THEN
c            T=(T2-T1)/LOG(RATIO)
c         ELSE
c            T=HALF*(T1+T2)
c         END IF
c         CC(J,I,K)=YX*DELR(J)*T*(BUFF(J,I,K)+BUFF(J,I+1,K))
c        *             /(DELC(I+1)+DELC(I))
         if(itrpy.eq.0)then                                               AT 03DB
           RATIO=T2/T1                                                    AT 03E7
           IF(RATIO.GT.FRAC1.OR.RATIO.LT.FRAC2) THEN                      AT 03FA
              T=(T2-T1)/LOG(RATIO)                                        AT 041A
           ELSE                                                           AT 043A
              T=HALF*(T1+T2)                                              AT 043F
           END IF                                                         AT 0455
           CC(J,I,K)=YX*DELR(J)*T*(BUFF(J,I,K)+BUFF(J,I+1,K))             AT 0455
        *             /(DELC(I+1)+DELC(I))
         else                                                             AT 04CC
```

```
          t1=cc(j,i,k)                                                   AT 04D1
          t2=cc(j,i+1,k)                                                 AT 0504
          if(itrpy.ne.2)then                                             AT 0534
            t1=t1*trpy(j,i,k)                                            AT 0540
            t2=t2*trpy(j,i+1,k)                                          AT 056D
          endif                                                          AT 059A
          if(t1.eq.zero.or.t2.eq.zero)then                               AT 059A
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
            cc(j,i,k)=zero                                               AT 05C0
            goto 40                                                      AT 05ED
          endif                                                          AT 05F2
          ratio=t2/t1                                                    AT 05F2
          if(ratio.gt.frac1.or.ratio.lt.frac2) then                      AT 0605
             t=(t2-t1)/log(ratio)                                        AT 0625
          else                                                           AT 0645
             t=half*(t1+t2)                                              AT 064A
          end if                                                         AT 0660
          cc(j,i,k)=delr(j)*t*(buff(j,i,k)+buff(j,i+1,k))                AT 0660
     *              /(delc(i+1)+delc(i))
        endif                                                            AT 06D1
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
   40 CONTINUE                                                           AT 06D1
C
C5------RETURN
      RETURN                                                             AT 06DB
      END                                                                AT 06E0



C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
C-----17/02/99-----------(based on subroutine SBCF5A(from BCF5.f01)-----
C


      SUBROUTINE SBCF5I(CR,CC,TRPY,DELR,DELC,K,NCOL,NROW,NLAY,itrpy)
C
C-----VERSION 02JULY1993 SBCF5I
C     ******************************************************************
C-------COMPUTE CONDUCTANCE USING INTERNODAL TRANSMISSIVITY
C-------ACTIVATED BY LAYAVG=40
c-------TRANSMISSIVITIES ARE IN CC & CR UPON ENTRY
C     ******************************************************************
C
C      SPECIFICATIONS:
C     ------------------------------------------------------------------
C
      DIMENSION CR(NCOL,NROW,NLAY), CC(NCOL,NROW,NLAY)
     2   , TRPY(ncol,nrow,NLAY), DELR(NCOL), DELC(NROW)
C
C     ------------------------------------------------------------------
      ZERO=0.                                                            AT 0058
      if(itrpy.eq.0)YX=TRPY(K,1,1)                                       AT 0066
C
C1------FOR EACH CELL CALCULATE BRANCH CONDUCTANCES FROM THAT CELL
C1------TO THE ONE ON THE RIGHT AND THE ONE IN FRONT.
      DO 40 I=1,NROW                                                     AT 0093
      DO 40 J=1,NCOL                                                     AT 00B1
      T1=Cr(J,I,K)                                                       AT 00CF
      t2=cc(j,i,k)                                                       AT 00F5

C3------IF THIS IS NOT THE LAST COLUMN(RIGHTMOST) THEN CALCULATE
C3------BRANCH CONDUCTANCE IN THE ROW DIRECTION (CR) TO THE RIGHT.
      IF(J.EQ.NCOL) GO TO 30                                             AT 011C
C3A-----INTERBLOCK TRANSMISSIVITY
      cr(j,i,k)=delc(i)*t1*2/(delr(j+1)+delr(j))                         AT 012A
C
C4------IF THIS IS NOT THE LAST ROW(FRONTMOST) THEN CALCULATE
C4------BRANCH CONDUCTANCE IN THE COLUMN DIRECTION (CC) TO THE FRONT.
   30 IF(I.EQ.NROW) GO TO 40                                             AT 0164
         if(itrpy.eq.0)then                                              AT 0172
           CC(J,I,K)=YX*DELR(J)*T2*2/(DELC(I+1)+DELC(I))                 AT 017E
         else                                                            AT 01C1
           CC(J,I,K)=DELR(J)*T2*2/(DELC(I+1)+DELC(I))                    AT 01C6
           if(itrpy.ne.2)cc(j,i,k)=cc(j,i,k)*trpy(j,i,k)                 AT 020A
         endif                                                           AT 0240
   40 CONTINUE                                                           AT 0240
C
C5------RETURN
      RETURN                                                             AT 024A
      END                                                                AT 024F
```

```
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<

C-Environment Agency (UK)-3.Auto-conversion for steady state VKD>>>>>>>>
c-------based on SBAS5O,bas5df,bas5al,bas5rp,bcf5al & bcf5rp


      subroutine bcf5vk (hnew,inunit,inbcf,inbas,ibcfout,
     1        inout,ibasout,ncol,nrow,nlay,ifrefm,fname,cunit,IXSEC,
     2        NPER,iout,IBOUND,HOLD,TRPY,DELR,DELC,SC1,CC,HY,BOT,CV,
     3        TOP,SC2,vkgrad,WETDRY,buff,cr,vmid,vkmax)
c
c-----version 11march1999 bcf5vk
c     ****************************************************************
c-------Writes new BCF, BAS and NAME files for 2nd initial conditions
c-------for VKD run. Initial heads from previous (specified T) run,
c-------top and bottom elevations calculated from heads and upper and
c-------lower zone thicknesses, ISS set to 1, return to start of sim.
c     ****************************************************************
c
c      specifications:
c     -----------------------------------------------------------------
c
      CHARACTER*24 ANAME(20)
      CHARACTER*4 CUNIT
      CHARACTER*80 LINE,fname,fnam2
      CHARACTER*11 FMTARG
      CHARACTER*80 HEADNG(2)
      CHARACTER*80 LINE1,LINE2
      character*20 fmtin(15,200)
      character*10 atmp,ahdry,awetfct
      character*30 PERLENNSTPTSMULT
c
      dimension buff(ncol,nrow,nlay),HOLD(NCOL,NROW,NLAY),
     1    vkgrad(ncol,nrow,nlay),WETDRY(NCOL,NROW,NLAY),
     2    SC2(NCOL,NROW,NLAY),TOP(NCOL,NROW,NLAY),BOT(NCOL,NROW,NLAY),
     3    SC1(NCOL,NROW,NLAY),HY(NCOL,NROW,NLAY),CC(NCOL,NROW,NLAY),
     4    CV(NCOL,NROW,NLAY),TRPY(ncol,nrow,nlay),cr(ncol,nrow,nlay),
     5    vmid(ncol,nrow,nlay),vkmax(ncol,nrow,nlay)
c
      integer IBOUND(NCOL,NROW,NLAY)
c
      DOUBLE PRECISION HNEW,hhnew
c
      DIMENSION HNEW(NCOL,NROW,NLAY),DELR(NCOL),DELC(NROW)
      common /flwcom/laycon(200)
      common /flwavg/layavg(200)
c
C     -----------------------------------------------------------------
c
      DATA ANAME(1) /'    PRIMARY STORAGE COEF'/                         AT 0088
      DATA ANAME(2) /'    TRANSMIS. ALONG ROWS'/                         AT 0088
      DATA ANAME(3) /'   HYD. COND. ALONG ROWS'/                         AT 0088
      DATA ANAME(4) /'VERT HYD COND /THICKNESS'/                         AT 0088
      DATA ANAME(5) /'                  BOTTOM'/                         AT 0088
      DATA ANAME(6) /'                     TOP'/                         AT 0088
      DATA ANAME(7) /'  SECONDARY STORAGE COEF'/                         AT 0088
      DATA ANAME(8) /'COLUMN TO ROW ANISOTROPY'/                         AT 0088
      DATA ANAME(9) /'                    DELR'/                         AT 0088
      DATA ANAME(10)/'                    DELC'/                         AT 0088
      DATA ANAME(11)/'        WETDRY PARAMETER'/                         AT 0088
      data aname(12)/'HYD COND GRADIENT FACTOR'/                         AT 0088
      data aname(13)/'          BOUNDARY ARRAY'/                         AT 0088
      data aname(14)/'          STARTING HEADS'/                         AT 0088
      data aname(15)/' TRANSMIS. ALONG COLUMNS'/                         AT 0088
      data aname(16)/'HYD. COND. ALONG COLUMNS'/                         AT 0088
      data aname(17)/'ELEVATION OF CHANGE IN K'/                         AT 0088
      data aname(18)/' MAXIMUM HYD COND FACTOR'/                         AT 0088
      data aname(19)/' THICKNESS OF UPPER ZONE'/                         AT 0088
      DATA ANAME(20)/' THICKNESS OF LOWER ZONE'/                         AT 0088
c
C     -----------------------------------------------------------------
C
C-----------------------------------------------------------------------
C1------OPEN OLD AND NEW NAME FILES-------------------------------------
C1------(CODE BASED ON SUBROUTINE 'SBAS5O' IN BAS5.FOR)-----------------
C-----------------------------------------------------------------------
      OPEN(UNIT=inunit,FILE=FNAME,STATUS='OLD')                          AT 0088
      lloc=1                                                             AT 00BE
      call urword(fname,lloc,istart,istop,0,n,r,iout,inunit)             AT 00C8
      fnam2=fname                                                        AT 010A
      fnam2(istop:istop)='2'                                             AT 011C
```

```
            OPEN(UNIT=inout,FILE=FNAM2)                                              AT 012A
            write(iout,33)fnam2                                                      AT 0154
      33    format(1x,/1x,'CREATING NEW NAME FILE: ',a)                              AT 0194
            ierr=0                                                                   AT 01D1
      C
      C1a-----READ A LINE; IGNORE BLANK LINES
      10    READ(INUNIT,'(A)',END=1000) LINE                                         AT 01DB
            IF(LINE.EQ.' '.or.LINE(1:1).EQ.'#') then                                 AT 023F
               write(inout,'(A)')line                                                AT 0261
               GO TO 10                                                              AT 02B9
            END IF                                                                   AT 02BE
      C
      C1b-----DECODE THE FILE TYPE.
            LLOC=1                                                                   AT 02BE
            CALL URWORD(LINE,LLOC,ITYP1,ITYP2,1,N,R,IOUT,INUNIT)                     AT 02C8
            CALL URWORD(LINE,LLOC,ISTART,ISTOP,2,IU,R,IOUT,INUNIT)                   AT 030E
            CALL URWORD(LINE,LLOC,INAM1,INAM2,0,N,R,IOUT,INUNIT)                     AT 0354
      c
      C1c-----IF TYPE IS LIST, DATA OR BINARY, rename output files
            IF(LINE(ITYP1:ITYP2).EQ.'LIST'.or.line(ityp1:ityp2).eq.'DATA'.or.        AT 039A
           1 line(ityp1:ityp2).eq.'DATA(BINARY)') then
       111    call urword(line,lloc,istart,istop,0,n,r,iout,inunit)                  AT 0424
              if(line(istop:istop).eq.'>')then                                       AT 046A
                call urword(line,lloc,istart,istop,0,n,r,iout,inunit)                AT 047E
                line1=line(istart:80)                                                AT 04C4
                write(line(inam1:80),'(a)')line1                                     AT 04EE
              elseif(line(istart:istop).eq.'DIRECT')then                             AT 0560
                call urword(line,lloc,istart,istop,2,irecl,r,iout,inunit)            AT 0597
                goto 111                                                             AT 05DD
              elseif(line(ityp1:ityp2).ne.'DATA')then                                AT 05E2
                line(inam2:inam2)='2'                                                AT 0614
              endif                                                                  AT 0622
            endif                                                                    AT 0622
      c
      C1d-----IF TYPE IS BCF OR BAS, CREATE NEW BCF & BAS FILES (BC2 & BA2)
            if(line(ityp1:ityp2).eq.cunit.or.line(ityp1:ityp2).eq.'BAS')then         AT 0622
               CALL URWORD(LINE,LLOC,ISTART,ISTOP,0,N,R,IOUT,INUNIT)                 AT 0682
      c
      C1e-----CHANGE FILENAMES OF NEW BAS & BCF FILES
               IF(LINE(ityp1:ityp2).eq.cunit)then                                    AT 06C8
                  iu1=inbcf                                                          AT 06F9
                  iu2=ibcfout                                                        AT 0707
               else                                                                  AT 0715
                  iu1=inbas                                                          AT 071A
                  iu2=ibasout                                                        AT 0728
               end if                                                                AT 0736
               IF(LINE(ISTop:ISTOP).EQ.'>') THEN                                     AT 0736
                  OPEN(UNIT=IU1,FILE=LINE(INAM1:INAM2),FORM='FORMATTED',             AT 074A
           1            ACCESS='SEQUENTIAL')
                  CALL URWORD(LINE,LLOC,ISTART,ISTOP,0,n,R,IOUT,INUNIT)              AT 07AA
                  WRITE(IOUT,36) LINE(istart:istop),                                 AT 07F0
           1           LINE(ITYP1:ITYP2),IU2
                  OPEN(UNIT=IU2,FILE=LINE(Istart:Istop),FORM='FORMATTED',            AT 0888
           1            ACCESS='SEQUENTIAL')
                  line1=line(istart:istop)                                           AT 08E8
                  write(line(inam1:80),'(a)')line1                                   AT 0914
               ELSE                                                                  AT 0986
                  OPEN(UNIT=IU1,FILE=LINE(INAM1:INAM2),FORM='FORMATTED',             AT 098B
           1            ACCESS='SEQUENTIAL')
                  line(inam2:inam2)='2'                                              AT 09EB
                  WRITE(IOUT,36) LINE(INAM1:INAM2),                                  AT 09F9
           1           LINE(ITYP1:ITYP2),IU2
      36          FORMAT(1X,/1X,'CREATING ',A,/                                      AT 0A91
           1        1X,'FILE TYPE:',A,'   UNIT',I4)
                  OPEN(UNIT=IU2,FILE=LINE(INAM1:INAM2),FORM='FORMATTED',             AT 0AEE
           1            ACCESS='SEQUENTIAL')
               END IF                                                                AT 0B4E
            end if                                                                   AT 0B4E
      c
      C1f-----WRITE FILENAMES IN NEW NAME FILE
            write(inout,'(A)')line                                                   AT 0B4E
            GO TO 10                                                                 AT 0BA6
      1000  CLOSE(UNIT=INUNIT)                                                       AT 0BAB
            CLOSE(UNIT=INOUT)                                                        AT 0BC3
            fname=fnam2                                                              AT 0BDB
      C     -------------------------------------------------------------
      c
      C-------------------------------------------------------------------
      C2------READ OLD BAS FILE, WRITE NEW (BA2) FILE-------------------------
      C2------(CODE BASED ON BAS5DF, BAS5AL & BAS5RP)
      C-------------------------------------------------------------------
      c
```

```
C2a-----READ AND PRINT A HEADING.
      READ(INBAS,'(A)') HEADNG(1)                                          AT 0BED
      READ(INBAS,'(A)') HEADNG(2)                                          AT 0C45
      WRITE(IBASOUT,'(A)') HEADNG(1)                                       AT 0C9D
      WRITE(IBASOUT,'(A)') HEADNG(2)                                       AT 0CF5
C
C2b-----READ LINE SPECIFYING NUMBER OF LAYERS,ROWS,COLUMNS,STRESS
C2b-----PERIODS AND UNITS OF TIME CODE, BUT DON'T DECODE.
      READ(INBAS,'(A)') LINE1                                              AT 0D4D
      READ(INBAS,'(A)') LINE2                                              AT 0DA5
      WRITE(IBASOUT,'(A)') LINE1                                           AT 0DFD
      WRITE(IBASOUT,'(A)') LINE2                                           AT 0E55
c
C2c-----READ & PRINT FLAG IAPART (RHS & BUFFER SHARE SPACE?) AND
C2c-----FLAG ISTRT (SHOULD STARTING HEADS BE KEPT FOR DRAWDOWN?).
      IF(IFREFM.EQ.0) THEN                                                 AT 0EAD
         READ(INBAS,'(2I10)') IAPART,ISTRT                                 AT 0EB9
         WRITE(IBASOUT,'(2I10)') IAPART,ISTRT                              AT 0F1F
      ELSE                                                                 AT 0F85
         READ(INBAS,*) IAPART,ISTRT                                        AT 0F8A
         WRITE(IBASOUT,*) IAPART,ISTRT                                     AT 0FCF
      END IF                                                               AT 1014
C
C2d-----READ BOUNDARY ARRAY(IBOUND) ONE LAYER AT A TIME.
      IF(IXSEC.EQ.0) THEN                                                  AT 1014
         DO 101 K=1,NLAY                                                   AT 1020
         KK=K                                                              AT 104D
         write(ibasout,64) inbas,k                                         AT 1053
         CALL U2DINT(Ibound(1,1,kk),aname(13),nrow,ncol,0,INbas,iout)      AT 10A3
         DO 102 I=1,nrow                                                   AT 10F9
  102       WRITE(IbasOUT,'(25i3)') (ibound(j,i,k),j=1,ncol)               AT 1126
  101    CONTINUE                                                          AT 11D8
      ELSE                                                                 AT 11DD
         CALL U2DINT(IBOUND(1,1,1),ANAME(13),NLAY,NCOL,-1,INBAS,IOUT)      AT 11E2
         write(ibasout,63) inbas                                           AT 1230
         DO 103 I=1,nlay                                                   AT 126C
  103       WRITE(IbasOUT,'(25i3)') (ibound(j,i,k),j=1,ncol)               AT 1299
      END IF                                                               AT 134B
   64    format(i10,9x,'1(25I3)',22x,' 4     IBOUND Layer',i4)             AT 134B
   63    format(i10,9x,'1(25I3)',22x,' 4     IBOUND')                      AT 1392
C
C2e-----READ AND PRINT HEAD VALUE TO BE PRINTED FOR NO-FLOW CELLS.
      IF(IFREFM.EQ.0) THEN                                                 AT 13CD
         READ(INBAS,'(a10)') aTMP                                          AT 13D9
         WRITE(IBASOUT,'(a10)') aTMP                                       AT 1431
      ELSE                                                                 AT 1489
         READ(INBAS,*) TMP                                                 AT 148E
         WRITE(IBASOUT,*) TMP                                              AT 14BF
      END IF                                                               AT 14F0
C
C2f-----READ INITIAL HEADS.
      IF(IXSEC.EQ.0) THEN                                                  AT 14F0
         DO 301 K=1,NLAY                                                   AT 14FC
         KK=K                                                              AT 1529
         CALL b12dri(fmtin(1,kk),HOLD(1,1,KK),NROW,NCOL,INBAS,IBASOUT,0,   AT 152F
     1      0)
  301    CONTINUE                                                          AT 15A4
      ELSE                                                                 AT 15A9
         CALL b12dri(fmtin(1,1),HOLD(1,1,1),NLAY,NCOL,INBAS,IBASOUT,0,0)   AT 15AE
      END IF                                                               AT 1604
c
c2g-----Put heads calculated from specified transmissivity model into
c2g-----new *.bas file as initial heads.
      IF(IXSEC.EQ.0) THEN                                                  AT 1604
         DO 302 K=1,NLAY                                                   AT 1610
           fmtin(1,k)='(10E23.16)'                                         AT 163D
           write(ibasout,56)inbas,fmtin(1,k),aname(14),K                   AT 1661
           do 302 i=1,nrow                                                 AT 1702
             write(ibasout,fmtin(1,k)) (hnew(j,i,k),j=1,ncol)              AT 172F
  302    CONTINUE                                                          AT 17ED
      ELSE                                                                 AT 17F7
         write(ibasout,43)inbas,fmtin(1,1)                                 AT 17FC
         do 303 i=1,nlay                                                   AT 1852
           write(ibasout,fmtin(1,1)) (hnew(j,i,k),j=1,ncol)                AT 187F
  303    CONTINUE                                                          AT 1920
   56 format(i10,5x,'1.000',a20,8x,' 0',a24,' Layer',i4)                   AT 1925
   43 format(i10,5x,'1.000',a20,8x,'-1     STARTING HEADS')                AT 196E
      END IF                                                               AT 19B5
C
C2h-----READ AND WRITE LENGTH OF STRESS PERIOD, NUMBER OF TIME STEPS AND
C2h-----TIME STEP MULTIPLIER.
      DO 777 III=1,NPER                                                    AT 19B5
```

```
          IF(IFREFM.EQ.0) THEN                                                AT 19E2
             READ(INBAS,'(a30)') PERLENNSTPTSMULT                             AT 19EE
             WRITE(IBASOUT,'(a30)') PERLENNSTPTSMULT                          AT 1A46
          ELSE                                                                AT 1A9E
             READ(INBAS,*) PERLEN,NSTP,TSMULT                                 AT 1AA3
             WRITE(IBASOUT,*) PERLEN,NSTP,TSMULT                              AT 1AFC
          END IF                                                              AT 1B55
  777  CONTINUE                                                               AT 1B55
c
       CLOSE (UNIT=IBASOUT)                                                   AT 1B5A
C     ------------------------------------------------------------------
c
C-----------------------------------------------------------------------
C3------READ OLD BCF FILE
c3------(CODE BASED ON BCF5AL & BCF5RP)
C-----------------------------------------------------------------------
c
C3a-----READ ISS (STEADY-STATE FLAG), IBCFCB (FLAG FOR
C3a-----PRINTING OR UNIT# FOR RECORDING CELL-BY-CELL FLOW TERMS), HDRY
C3a-----(HEAD AT CELLS THAT CONVERT TO DRY), WETTING PARAMETERS,
c3a-----anisotropy option, explicit transmissivity calculation option,
c3a-----transmissivity output option, and layer type codes.
       IF(IFREFM.EQ.0) THEN                                                   AT 1B72
          READ(INbcf,'(2I10,a10,I10,a10,5I10)')                               AT 1B7E
     1              ISS,IBCFCB,aHDRY,IWDFLG,aWETFCT,IWETIT,IHDWET,itrpy
     2              ,ihold,itrans
          read(aHDRY,*)HDRY                                                   AT 1CA8
WARNING - The use of list-directed formatting with an internal file is a Fortran 77 extension
          READ(INbcf,'(40I2)') (LAYCon(I),I=1,NLAY)                           AT 1CE0
       ELSE                                                                   AT 1D62
          READ(INbcf,*) ISS,IBCFCB,HDRY,IWDFLG,WETFCT,IWETIT,IHDWET,itrpy     AT 1D67
     1                 ,ihold,itrans
          READ(INbcf,*) (LAYCon(I),I=1,NLAY)                                  AT 1E4C
       END IF                                                                 AT 1EAD
c
C3b-----LOOP THROUGH LAYERS CALCULATING LAYAVG.
       do k=1,nlay                                                            AT 1EAD
         layavg(k)=int(laycon(k)/10)*10                                       AT 1EDA
         laycon(k)=laycon(k)-layavg(k)                                        AT 1EFB
       enddo                                                                  AT 1F09
c
c3c-----READ ANISOTROPY RATIOS.
       if(itrpy.eq.0)then                                                     AT 1F0B
         call b12dri(fmtin(2,1),trpy(1,1,1),1,nlay,inbcf,iBCFout,0,0)         AT 1F18
         do k=1,nlay                                                          AT 1F6D
           if(layavg(k).eq.40)then                                            AT 1F9A
             itrpy=1                                                          AT 1FA8
             do kk=nlay,1,-1                                                  AT 1FB2
               do i=nrow,1,-1                                                 AT 1FD5
                 do j=ncol,1,-1                                               AT 1FF8
                   trpy(j,i,kk)=trpy(kk,1,1)                                  AT 201E
                 enddo                                                        AT 2071
               enddo                                                          AT 2073
             enddo                                                            AT 2078
             goto 24                                                          AT 207D
           endif                                                              AT 2082
         enddo                                                                AT 2082
       elseif(itrpy.ne.2)then                                                 AT 2087
         do 23 k=1,nlay                                                       AT 2099
           call b12dri(fmtin(2,k),trpy(1,1,k),nrow,ncol,inbcf,iBCFout,0,      AT 20C6
     1       0)
   23    continue                                                             AT 213D
       endif                                                                  AT 2142
   24  continue                                                               AT 2142
c
c3d-----READ ROW AND COLUMN SPACINGS.
       CALL b12dri(fmtin(3,1),DELR,1,NCOL,INbcf,IOUT,0,0)                     AT 2142
       CALL b12dri(fmtin(4,1),DELC,1,NROW,INbcf,IOUT,0,0)                     AT 2178
C
C3e-----READ ALL PARAMETERS FOR EACH LAYER.
       KT=0                                                                   AT 21AB
       KB=0                                                                   AT 21B5
       kg=0                                                                   AT 21BF
       DO 200 K=1,NLAY                                                        AT 21C9
       KK=K                                                                   AT 21F6
C
C3e1----FIND ADDRESS OF EACH LAYER IN THREE DIMENSION ARRAYS.
       IF(LAYCON(K).EQ.1 .OR. LAYCON(K).EQ.3 .or. laycon(k).eq.4 .or.         AT 21FC
     1     laycon(k).eq.5) KB=KB+1
       IF(LAYCON(K).EQ.2 .OR. LAYCON(K).EQ.3 .or. laycon(k).eq.5) KT=KT+1     AT 223A
       if(laycon(k).eq.4 .or. laycon(k).eq.5) kg=kg+1                         AT 2270
C
```

```
C3e2----READ TRANSMISSIVITY INTO ARRAY CC IF LAYER TYPE IS 0 OR 2.
      IF(LAYCON(K).EQ.3 .OR. LAYCON(K).EQ.1 .or.laycon(k).eq.4 .or.        AT 2298
     1      laycon(k).eq.5) GO TO 100
      CALL b12dri(fmtin(6,k),CC(1,1,K),NROW,NCOL,INbcf,IOUT,0,0)          AT 22D6
      if(itrpy.ne.2)goto 110                                              AT 234A
      do 14 i=1,nrow                                                      AT 2357
      do 14 j=1,ncol                                                      AT 2384
         cr(j,i,k)=cc(j,i,k)                                              AT 23B4
 14   continue                                                            AT 23EC
c
c3e2i---READ Y-DIR TRANS IF TRPY=2.
      call b12dri(fmtin(2,k),cc(1,1,k),nrow,ncol,inbcf,iout,0,0)          AT 23F3
      GO TO 110                                                           AT 246D
C
C3e3----READ HYDRAULIC CONDUCTIVITY(HY) AND BOTTOM ELEVATION(BOT)
C3e3----IF LAYER TYPE IS 1, 3, 4 or 5.
  100 CALL b12dri(fmtin(7,k),HY(1,1,KB),NROW,NCOL,INbcf,IOUT,0,0)         AT 2472
      if(itrpy.ne.2)goto 105                                              AT 24EC
c
c3e3i---READ Y-DIR HYDRAULIC CONDUCTIVITY IF TRPY=2.
      call b12dri(fmtin(2,k),trpy(1,1,kb),nrow,ncol,inbcf,iout,0,0)       AT 24F9
 105  CALL b12dri(fmtin(8,k),BOT(1,1,KB),NROW,NCOL,INbcf,IOUT,0,0)        AT 2573
C
C3e4----READ VERTICAL HYCOND/THICK INTO ARRAY CV IF NOT BOTTOM LAYER;
C3e4----MULTIPLIED BY CELL AREA TO CONVERT TO CONDUCTANCE LATER.
  110 IF(K.EQ.NLAY) GO TO 120                                             AT 25ED
      CALL b12dri(fmtin(9,k),CV(1,1,K),NROW,NCOL,INbcf,IOUT,0,0)          AT 25FE
C
  120 IF(LAYCON(K).NE.3.AND.LAYCON(K).NE.2.and.laycon(k).ne.5) GO TO 130  AT 2672
C
C3e5----READ TOP ELEVATION(TOP) IF LAYER TYPE IS 2, 3 or 5.
      CALL b12dri(fmtin(11,k),TOP(1,1,KT),NROW,NCOL,INbcf,IOUT,0,0)       AT 26A2
C
C3e6----READ WETDRY CODES IF LAYER TYPE IS 1, 3, 4 or 5 AND WETTING
C3e6----CAPABILITY HAS BEEN INVOKED (IWDFLG NOT 0).
  130 IF(LAYCON(K).NE.3 .AND. LAYCON(K).NE.1 .and. laycon(k).ne.4 .and.   AT 271C
     1      laycon(k).ne.5) GO TO 200
      IF(IWDFLG.EQ.0)GO TO 140                                            AT 275A
      CALL b12dri(fmtin(13,k),WETDRY(1,1,KB),NROW,NCOL,INbcf,IOUT,0,0)    AT 2767
c
c3e7----read middle elevation into array VMID, hydraulic conductivity
c3e7----gradient into array VKGRAD, and maximum hydraulic conductivity
c3e7----into array VKMAX if layer type is 4 or 5.
 140  if(laycon(k).ne.4 .and. laycon(k).ne.5) go to 200                   AT 27E1
      call b12dri(fmtin(14,k),vmid(1,1,kg),nrow,ncol,inbcf,iout,0,0)      AT 2803
      fmtin(14,k)='(10E23.16)'                                            AT 287D
      fmtin(7,k)='(10E23.16)'                                             AT 28A9
      if(itrpy.eq.2.or.layavg(k).eq.40)fmtin(2,k)='(10E23.16)'            AT 28D5
      call b12dri(fmtin(12,k),vkgrad(1,1,kg),nrow,ncol,inbcf,iout,0,0)    AT 2922
      call b12dri(fmtin(15,k),vkmax(1,1,kg),nrow,ncol,inbcf,iout,0,0)     AT 299C
      do 141 i=1,nrow                                                     AT 2A16
      do 141 j=1,ncol                                                     AT 2A43
        if(vkgrad(j,i,kg).le.0.0)then                                     AT 2A73
          vkgrad(j,i,kg)=1.0                                              AT 2AB2
          vkmax(j,i,kg)=1.0                                               AT 2AE8
        endif                                                             AT 2B21
 141  continue                                                            AT 2B21
c
C-----------------------------------------------------------------------
C4------CALCULATE NEW HYD COND, ANISOTROPY, AND MIDDLE ELEVATIONS IF
C4------LAYER TYPE IS 4 OR 5.
C-----------------------------------------------------------------------
c
c4a-----CORRECT UPPER THICKNESSES IF THEY OVERLAP WITH THE BOTTOM
 150  if(laycon(k).eq.4.or.laycon(k).eq.5)then                            AT 2B2B
        do 42 j=1,ncol                                                    AT 2B4D
        do 42 i=1,nrow                                                    AT 2B7D
          if(laycon(k).eq.5.and.hnew(j,i,k).gt.top(j,i,kt))then           AT 2BAA
            if(vmid(j,i,kg).gt.top(j,i,kt)-bot(j,i,kb).and.ibound(j,i,k)  AT 2C29
     1            .ne.0)then
              if(ierr.eq.0)then                                           AT 2CFF
                ierr=1                                                    AT 2D0C
                write(iout,*)                                             AT 2D16
                write(iout,*)'TOP THICKNESSES REDUCED AT THE FOLLOWING L  AT 2D30
     1OCATIONS:'
                write(iout,*)'Reduced FROM    TO  at  LAYER,ROW,COL'      AT 2D62
                write(iout,*)'------------------------------------'       AT 2D94
              endif                                                       AT 2DC6
              write(iout,45)vmid(j,i,kg),top(j,i,kt)-bot(j,i,kb),k,i,j    AT 2DC6
  45          format(2g12.4,' (',i3,',',i3,',',i3,')')                    AT 2EEA
              vmid(j,i,kg)=top(j,i,kt)-bot(j,i,kb)                        AT 2F2B
            endif                                                         AT 2FB3
```

```
           else                                                          AT 2FB3
             if(vmid(j,i,kg).gt.hnew(j,i,k)-bot(j,i,kb).and.ibound(j,i,k)   AT 2FB8
     1            .ne.0)then
               if(ierr.eq.0)then                                         AT 307E
                 ierr=1                                                  AT 308B
                 write(iout,*)                                           AT 3095
                 write(iout,*)'TOP THICKNESSES REDUCED AT THE FOLLOWING L  AT 30AF
     1OCATIONS:'
                 write(iout,*)'Reduced FROM    TO  at  LAYER,ROW,COL'     AT 30E1
                 write(iout,*)'------------------------------------'     AT 3113
               endif                                                     AT 3145
               write(iout,45)vmid(j,i,kg),hnew(j,i,k)-bot(j,i,kb),k,i,j   AT 3145
               vmid(j,i,kg)=hnew(j,i,k)-bot(j,i,kb)                      AT 326C
             endif                                                       AT 32F7
           endif                                                         AT 32F7
   42    continue                                                        AT 32F7
       endif                                                             AT 3301
C
C4B-----LOOP THRU CELLS IN LAYER CALCULATING VKD PARAMETERS.
       do 44 j=1,ncol                                                    AT 3301
       do 44 i=1,nrow                                                    AT 3331
          if(ibound(j,i,k).eq.0)goto 44                                  AT 335E
C
C4C-----INTERNODAL TRANSMISSIVITY OPTION.
          if(layavg(k).eq.40)then                                        AT 3394
C
C4C1----SET VARIABLES FOR X-DIR CALCULATIONS.
             jp=j+1                                                      AT 33A8
             ip=i+1                                                      AT 33B5
             if(j.eq.ncol)jp=j                                           AT 33BC
             if(i.eq.nrow)ip=i                                           AT 33D6
             bbot=(bot(j,i,kb)+bot(jp,i,kb))/2                           AT 33ED
             hhnew=(hnew(j,i,k)+hnew(jp,i,k))/2                          AT 3443
             if(laycon(k).eq.3.or.laycon(k).eq.5)then                    AT 349C
               ttop=(top(j,i,kt)+top(jp,i,kt))/2                         AT 34BE
               if(hhnew.gt.ttop)hhnew=ttop                               AT 3514
             endif                                                       AT 353E
             if(laycon(k).eq.4.or.laycon(k).eq.5)then                    AT 353E
               vvmid=(vmid(j,i,kg)+vmid(jp,i,kg))/2                      AT 3560
               vvkgrad=(vkgrad(j,i,kg)+vkgrad(jp,i,kg))/2                AT 35B6
               vvkmax=(vkmax(j,i,kg)+vkmax(jp,i,kg))/2                   AT 360C
               if(iss.eq.3)then                                          AT 3662
                 T=hy(j,i,kb)*(hhnew-bbot)                               AT 366F
               else                                                      AT 36AF
                 T=hy(j,i,kb)                                            AT 36B4
               endif                                                     AT 36F1
C
C4C2----CALCULATE X-DIR BASE HYDRAULIC CONDUCTIVITIES.
               if(vvmid.gt.(vvkmax - 1.0)/vvkgrad)then                   AT 36F1
                 buff(j,i,k) =T / (hhnew-bbot-(1.0/(2.0*vvkgrad)         AT 3713
     1             * (vvkmax - 1.0) ** 2.0 ) + vvmid * (vvkmax - 1.0) )
               else                                                      AT 37A5
                 buff(j,i,k) = T / ((hhnew-vvmid-bbot) + vvmid           AT 37AA
     1             * (1.0 + 0.5 * vvkgrad * vvmid) )
               endif                                                     AT 3812
C
C4C3----CALCULATE X-DIR HYDRAULIC CONDUCTIVITIES (NOT USED)
             elseif(laycon(k).eq.1.or.laycon(k).eq.3)then                AT 3812
               buff(j,i,k) = hy(j,i,kb) / (hhnew - bbot)                 AT 3839
             endif                                                       AT 38A2
C
C4C4----SET VARIABLES FOR Y-DIR CALCULATIONS.
             bbot=(bot(j,i,kb)+bot(j,ip,kb))/2                           AT 38A2
             hhnew=(hnew(j,i,k)+hnew(j,ip,k))/2                          AT 3905
             if(laycon(k).eq.3.or.laycon(k).eq.5)then                    AT 396B
               ttop=(top(j,i,kt)+top(j,ip,kt))/2                         AT 398D
               if(hhnew.gt.ttop)hhnew=ttop                               AT 39F0
             endif                                                       AT 3A1A
             if(laycon(k).eq.4.or.laycon(k).eq.5)then                    AT 3A1A
               vvmid=(vmid(j,i,kg)+vmid(j,ip,kg))/2                      AT 3A3C
               vvkgrad=(vkgrad(j,i,kg)+vkgrad(j,ip,kg))/2                AT 3A9F
               vvkmax=(vkmax(j,i,kg)+vkmax(j,ip,kg))/2                   AT 3B02
               if(itrpy.eq.2)then                                        AT 3B65
                 T=trpy(j,i,kb)                                          AT 3B72
               else                                                      AT 3BAF
                 T=trpy(j,i,k)*hy(j,i,kb)/buff(j,i,k)                    AT 3BB4
               endif                                                     AT 3C1C
               if(iss.eq.3)T=T*(hhnew-bbot)                              AT 3C1C
C
C4C5----CALCULATE Y-DIR BASE HYDRAULIC CONDUCTIVITIES.
               if(vvmid.gt.(vvkmax - 1.0)/vvkgrad)then                   AT 3C42
                 trpy(j,i,kb)= T / (hhnew - bbot - (1.0 / (2.0 * vvkgrad)   AT 3C64
```

```
     1           * (vvkmax - 1.0) ** 2.0 ) + vvmid * (vvkmax - 1.0) )
               else                                                      AT 3CF6
                 trpy(j,i,kb)= T / ((hhnew - vvmid - bbot) + vvmid *     AT 3CFB
     1                     (1.0 + 0.5 * vvkgrad * vvmid))
               endif                                                     AT 3D63
C
C4C6----CALCULATE Y-DIR HYDRAULIC CONDUCTIVITIES (NOT USED)
             elseif(laycon(k).eq.1.or.laycon(k).eq.3)then                AT 3D63
               if(itrpy.eq.2)then                                        AT 3D8A
                 trpy(j,i,kb)=trpy(j,i,kb)/(hhnew - bbot)                AT 3D97
               else                                                      AT 3DD7
                 trpy(j,i,k)=(trpy(j,i,k)*hy(j,i,kb)/(hhnew - bbot))/    AT 3DDC
     1                            buff(j,i,k)
               end if                                                    AT 3E50
             endif                                                       AT 3E50
             hy(j,i,kb)=buff(j,i,k)                                      AT 3E50
C
C4D-----BLOCK-CENTRED TRANSMISSIVITY OPTION.
          else                                                           AT 3EB2
C
C4D1----CALCULATE BASE HYDRAULIC CONDUCTIVITIES.
             hhnew=hnew(j,i,k)                                           AT 3EB7
             if((laycon(k).eq.3.or.laycon(k).eq.5).and.                  AT 3EF7
     1             hhnew.gt.top(j,i,kt))hhnew=top(j,i,kt)
             if(iss.eq.3)then                                            AT 3F99
               T=hy(j,i,kb)*(hhnew-bot(j,i,kb))                          AT 3FA6
             else                                                        AT 3FE7
               T=hy(j,i,kb)                                              AT 3FEC
             endif                                                       AT 4029
             if(laycon(k).eq.4.or.laycon(k).eq.5)then                    AT 4029
               if(vmid(j,i,kg).gt.(vkmax(j,i,kg)-1.0)/vkgrad(j,i,kg))then AT 404B
                 hy(j,i,kb) = T / (hhnew-bot(j,i,kb)-(1.0                AT 409C
     1             /(2.0*vkgrad(j,i,kg))*(vkmax(j,i,kg)-1.0)**2.0)
     2             +vmid(j,i,kg)*(vkmax(j,i,kg)-1.0))
               else                                                      AT 4161
                 hy(j,i,kb)= T /((hhnew-vmid(j,i,kg)-bot(j,i,kb))+       AT 4166
     1             vmid(j,i,kg)*(1.0+0.5*vkgrad(j,i,kg)*vmid(j,i,kg)))
               endif                                                     AT 41F5
C
C4D2----CALCULATE HYDRAULIC CONDUCTIVITIES (NOT USED).
             elseif(laycon(k).eq.1.or.laycon(k).eq.3)then                AT 41F5
               hy(j,i,kb)=hy(j,i,kb)/(hhnew-bot(j,i,kb))                 AT 421C
             endif                                                       AT 425D
C
C4D3----CALCULATE Y-DIR BASE HYDRAULIC CONDUCTIVITIES.
             if(itrpy.eq.2)then                                          AT 425D
               if(iss.eq.3)then                                          AT 426A
                 T=trpy(j,i,kb)*(hhnew-bot(j,i,kb))                      AT 4277
               else                                                      AT 42B8
                 T=trpy(j,i,kb)                                          AT 42BD
               endif                                                     AT 42FA
               if(laycon(k).eq.4.or.laycon(k).eq.5)then                  AT 42FA
                if(vmid(j,i,kg).gt.(vkmax(j,i,kg)-1.0)/vkgrad(j,i,kg))   AT 431C
     1             then
                 trpy(j,i,kb)= T / (hhnew-bot(j,i,kb)-(1.0               AT 436D
     1             /(2.0*vkgrad(j,i,kg))*(vkmax(j,i,kg)-1.0)**2.0)
     2             +vmid(j,i,kg)*(vkmax(j,i,kg)-1.0))
                else                                                     AT 4432
                 trpy(j,i,kb)= T /((hhnew-vmid(j,i,kg)-                  AT 4437
     1             bot(j,i,kb))+vmid(j,i,kg)*(1.0+0.5*vkgrad(j,i,kg)*
     2             vmid(j,i,kg)))
                endif                                                    AT 44C6
C
C4D4----CALCULATE Y-DIR HYDRAULIC CONDUCTIVITIES (NOT USED)
               elseif(laycon(k).eq.1.or.laycon(k).eq.3)then              AT 44C6
                 trpy(j,i,kb)=trpy(j,i,kb)/(hhnew-bot(j,i,kb))           AT 44ED
               endif                                                     AT 452E
             endif                                                       AT 452E
          end if                                                         AT 452E
C
C4E-----CALCULATE ELEVATION OF CHANGE IN K (MIDDLE ELEVATION, VMID).
          if(laycon(k).eq.4.or.laycon(k).eq.5)then                       AT 452E
            if(laycon(k).eq.5.and.hnew(j,i,k).gt.top(j,i,kt))then        AT 4550
              vmid(j,i,kg)=top(j,i,kt)-vmid(j,i,kg)                      AT 45D7
            else                                                         AT 4633
              vmid(j,i,kg)=hnew(j,i,k)-vmid(j,i,kg)                      AT 4638
            endif                                                        AT 469A
          endif                                                          AT 469A
   44 continue                                                           AT 469A
C
  200 CONTINUE                                                           AT 46A4
C
```

```
C-----------------------------------------------------------------------
C5------WRITE NEW BCF FILE (BC2)
c5------(CODE BASED ON BCF5AL & BCF5RP)
C-----------------------------------------------------------------------
c
C5A-----CHANGE STEADY STATE FLAG BACK TO NORMAL VALUE (FROM 2 TO 1)
C5A-----AND WRITE TOP TWO LINES OF NEW BCF FILE.
      ISS=1                                                             AT 46A9
      IF(IFREFM.EQ.0) THEN                                              AT 46B3
         write(Ibcfout,'(2I10,a10,I10,a10,5I10)')                       AT 46BF
     1              ISS,IBCFCB,aHDRY,IWDFLG,aWETFCT,IWETIT,IHDWET,itrpy
     2              ,ihold,itrans
         write(Ibcfout,'(80I1)') (layavg(i)/10,LAYCon(I),I=1,NLAY)      AT 47E9
      else                                                              AT 489F
         write(Ibcfout,*) ISS,IBCFCB,HDRY,IWDFLG,WETFCT,IWETIT,IHDWET,  AT 48A4
     1                    itrpy,ihold,itrans
         write(Ibcfout,'(40(2i1,a1))') (layavg(i)/10,LAYCOn(I),' ',     AT 4989
     1       I=1,NLAY)
      end if                                                            AT 4A67
C
C5B-----WRITE ANISOTROPY RATIO
      if(itrpy.eq.0)then                                                AT 4A67
         call SBCF5W(trpy(1,1,1),IBOUND,HNEW,NLAY,1,NLAY,0,INBCF,       AT 4A74
     1           FMTIN(2,1),ANAME(8),IBCFOUT,HDRY)
      elseif(itrpy.ne.2)then                                            AT 4AE4
         do 510 k=1,nlay                                                AT 4AF6
  510       call SBCF5W(trpy(1,1,K),IBOUND,HNEW,NCOL,NROW,NLAY,K,INBCF, AT 4B23
     1           FMTIN(2,k),ANAME(8),IBCFOUT,HDRY)
      endif                                                             AT 4BC2
C
c5C-----WRITE ROW SPACINGS (COLUMN WIDTHS)
       call SBCF5W(delr(1),IBOUND,HNEW,NCOL,1,NLAY,0,INBCF,             AT 4BC2
     1           FMTIN(3,1),ANAME(9),IBCFOUT,HDRY)
C
C5D-----WRITE COLUMN SPACINGS (ROW HEIGHTS)
       call SBCF5W(delc(1),IBOUND,HNEW,Nrow,1,NLAY,0,INBCF,             AT 4C19
     1           FMTIN(4,1),ANAME(10),IBCFOUT,HDRY)
C
C5E-----WRITE ALL PARAMETERS FOR EACH LAYER.
      KT=0                                                              AT 4C6D
      KB=0                                                              AT 4C77
      kg=0                                                              AT 4C81
      DO 400 K=1,NLAY                                                   AT 4C8B
      KK=K                                                              AT 4CB8
C
C5F-----FIND ADDRESS OF EACH LAYER IN THREE DIMENSION ARRAYS.
      IF(LAYCON(K).EQ.1 .OR. LAYCON(K).EQ.3 .or. laycon(k).eq.4 .or.    AT 4CBE
     1     laycon(k).eq.5) KB=KB+1
      IF(LAYCON(K).EQ.2 .OR. LAYCON(K).EQ.3 .or. laycon(k).eq.5) KT=KT+1  AT 4CFC
      if(laycon(k).eq.4 .or. laycon(k).eq.5) kg=kg+1                    AT 4D32
C
C5G-----write TRANSMISSIVITY INTO ARRAY CC IF LAYER TYPE IS 0 OR 2.
      IF(LAYCON(K).EQ.3 .OR. LAYCON(K).EQ.1 .or. laycon(k).eq.4 .or.    AT 4D5A
     1     laycon(k).eq.5) GO TO 300
      if(itrpy.ne.2)then                                                AT 4D98
         call SBCF5W(cc(1,1,K),IBOUND,HNEW,NCOL,NROW,NLAY,K,INBCF,      AT 4DA5
     1           FMTIN(6,k),ANAME(2),IBCFOUT,HDRY)
       else                                                             AT 4E35
         call SBCF5W(cr(1,1,K),IBOUND,HNEW,NCOL,NROW,NLAY,K,INBCF,      AT 4E3A
     1           FMTIN(6,k),ANAME(2),IBCFOUT,HDRY)
         call SBCF5W(cc(1,1,K),IBOUND,HNEW,NCOL,NROW,NLAY,K,INBCF,      AT 4ED4
     1           FMTIN(2,k),ANAME(15),IBCFOUT,HDRY)
      endif                                                             AT 4F6E
      GO TO 310                                                         AT 4F6E
C
C5H-----write HYDRAULIC CONDUCTIVITY(HY) AND BOTTOM ELEVATION(BOT)
C5H-----IF LAYER TYPE IS 1, 3, 4 or 5.
 300  call SBCF5W(HY(1,1,KB),IBOUND,HNEW,NCOL,NROW,NLAY,K,INBCF,        AT 4F73
     1           FMTIN(7,k),ANAME(3),IBCFOUT,HDRY)
      if(itrpy.ne.2)goto 547                                            AT 500D
       call SBCF5W(trpy(1,1,KB),IBOUND,HNEW,NCOL,NROW,NLAY,K,INBCF,     AT 501A
     1           FMTIN(2,k),ANAME(16),IBCFOUT,HDRY)
 547  call SBCF5W(bot(1,1,KB),IBOUND,HNEW,NCOL,NROW,NLAY,K,INBCF,       AT 50B4
     1           FMTIN(8,k),ANAME(5),IBCFOUT,HDRY)
C
C5I-----write VERTICAL HYCOND/THICK INTO ARRAY CV IF NOT BOTTOM LAYER;
C5I-----MULTIPLIED BY CELL AREA TO CONVERT TO CONDUCTANCE LATER.
  310 IF(K.EQ.NLAY) GO TO 320                                           AT 514E
      call SBCF5W(CV(1,1,K),IBOUND,HNEW,NCOL,NROW,NLAY,K,INBCF,         AT 515F
     1           FMTIN(9,k),ANAME(4),IBCFOUT,HDRY)
C
  320 IF(LAYCON(K).NE.3.AND.LAYCON(K).NE.2.and.laycon(k).ne.5) GO TO 330  AT 51EF
```

```
C
C5J-----write TOP ELEVATION(TOP) IF LAYER TYPE IS 2, 3 or 5.
      call SBCF5W(TOP(1,1,KT),IBOUND,HNEW,NCOL,NROW,NLAY,K,INBCF,             AT 521F
     1            FMTIN(11,k),ANAME(6),IBCFOUT,HDRY)
C
C5K-----write WETDRY CODES IF LAYER TYPE IS 1, 3, 4 or 5 AND WETTING
C5K-----CAPABILITY HAS BEEN INVOKED (IWDFLG NOT 0).
  330 IF(LAYCON(K).NE.3 .AND. LAYCON(K).NE.1 .and. laycon(k).ne.4 .and.      AT 52B9
     1            laycon(k).ne.5)GO TO 400
      IF(IWDFLG.EQ.0)GO TO 340                                                AT 52F7
      call SBCF5W(WETDRY(1,1,KB),IBOUND,HNEW,NCOL,NROW,NLAY,K,INBCF,          AT 5304
     1            FMTIN(13,k),ANAME(11),IBCFOUT,HDRY)
c
C5L-----WRITE HYDRAULIC CONDUCTIVITY GRADIENT FROM ARRAY VKGRAD,
C5L-----MIDDLE ELEVATION FROM ARRAY VMID and maximum hyd cond IF LAYER
C5L-----TYPE IS 4 OR 5.
 340   if(laycon(k).ne.4 .and. laycon(k).ne.5) go to 400                      AT 539E
      call SBCF5W(VMID(1,1,KG),IBOUND,HNEW,NCOL,NROW,NLAY,K,INBCF,            AT 53C0
     1            FMTIN(14,k),ANAME(17),IBCFOUT,HDRY)
      call SBCF5W(VKGRAD(1,1,KG),IBOUND,HNEW,NCOL,NROW,NLAY,K,INBCF,          AT 545A
     1            FMTIN(12,k),ANAME(12),IBCFOUT,HDRY)
      call SBCF5W(VKMAX(1,1,KG),IBOUND,HNEW,NCOL,NROW,NLAY,K,INBCF,           AT 54F4
     1            FMTIN(15,k),ANAME(18),IBCFOUT,HDRY)
C
C5M-----WRITE NEW UPPER THICKNESSES TO 1ST OUTPUT FILE IF THEY HAVE
C5M-----BEEN CHANGED.
      if(ierr.eq.1)then                                                       AT 558E
        do i=1,nrow                                                           AT 559B
          do j=1,ncol                                                         AT 55C8
            if(laycon(k).eq.5.and.hnew(j,i,k).gt.top(j,i,kt))then             AT 55F8
              buff(j,i,1)=top(j,i,kt)-vmid(j,i,kg)                            AT 5677
            else                                                              AT 56F5
              buff(j,i,1)=hnew(j,i,k)-vmid(j,i,kg)                            AT 56FA
            endif                                                             AT 577E
          enddo                                                               AT 577E
        enddo                                                                 AT 5783
        WRITE(IOUT,94) ANAME(19),K                                            AT 5788
  94    FORMAT(1X,///11X,A,' FOR LAYER',I4)                                   AT 57DC
        call ulaprw(buff,aname(19),0,0,ncol,nrow,0,1,iout)                    AT 5811
      endif                                                                   AT 5849
C
C5N-----WRITE LOWER THICKNESSES TO 1ST OUTPUT FILE.
      do i=1,nrow                                                             AT 5849
        do j=1,ncol                                                           AT 5876
          buff(j,i,1)=vmid(j,i,kg)-bot(j,i,kb)                                AT 58A6
        enddo                                                                 AT 591B
      enddo                                                                   AT 5920
      WRITE(IOUT,94) ANAME(20),K                                              AT 5925
      call ulaprw(buff,aname(20),0,0,ncol,nrow,0,1,iout)                      AT 5979
  400 continue                                                                AT 59B1
C
      close(unit=ibcfout)                                                     AT 59B6
C
C99------RETURN
      RETURN                                                                  AT 59CE
      END                                                                     AT 59D3
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<


C-Environment Agency (UK)-3.Auto-conversion for steady state VKD>>>>>>>>
C-----BASED ON U2DINT, u1dint & u2drel FROM UTL5.FOR
WARNING - Label 150 has not been referenced
COMMENT - FMTARG has not been used


      SUBROUTINE B12DRI(fmtin,A,II,JJ,IN,IOUT,iint,ip)
C
C-----VERSION 0001 11MAR1999 B12DRI
C     ******************************************************************
C     ROUTINE TO INPUT 1 & 2-D real DATA MATRICES
C       A IS ARRAY TO INPUT
C       II IS NO. OF ROWS
C       JJ IS NO. OF COLS
C       IN IS INPUT UNIT
C       IOUT IS OUTPUT UNIT
c       iint is 1 for integer array(option removed),otherwise:real array
c       ip is set to 0 to dissable printing to file
C     ******************************************************************
C
C        SPECIFICATIONS:
C     ------------------------------------------------------------------
```

```
      DIMENSION A(JJ,II)
      CHARACTER*20 FMTIN
      CHARACTER*80 CNTRL
      character*80 fname
      data nunopn/99/                                                     AT 003B
C     -----------------------------------------------------------------
C
C1------READ ARRAY CONTROL RECORD AS CHARACTER DATA.
      READ(IN,'(A)') CNTRL                                                AT 003B
      if(ip.ne.0)WRITE(IOUT,'(A)') CNTRL                                  AT 0090

C
C2------LOOK FOR ALPHABETIC WORD THAT INDICATES THAT THE RECORD IS FREE
C2------FORMAT.  SET A FLAG SPECIFYING IF FREE FORMAT OR FIXED FORMAT.
      ICLOSE=0                                                            AT 00F1
      IFREE=1                                                             AT 00FB
      ICOL=1                                                              AT 0105
      CALL URWORD(CNTRL,ICOL,ISTART,ISTOP,1,N,R,IOUT,IN)                  AT 010F
      IF (CNTRL(ISTART:ISTOP).EQ.'CONSTANT') THEN                         AT 0152
         locat=0                                                          AT 0184
      ELSE IF(CNTRL(ISTART:ISTOP).EQ.'INTERNAL') THEN                     AT 018E
         LOCAT=IN                                                         AT 01C5
      ELSE IF(CNTRL(ISTART:ISTOP).EQ.'EXTERNAL') THEN                     AT 01D0
         CALL URWORD(CNTRL,ICOL,ISTART,ISTOP,2,LOCAT,R,IOUT,IN)           AT 0207
      ELSE IF(CNTRL(ISTART:ISTOP).EQ.'OPEN/CLOSE') THEN                   AT 024A
         CALL URWORD(CNTRL,ICOL,ISTART,ISTOP,0,N,R,IOUT,IN)               AT 0281
         FNAME=CNTRL(ISTART:ISTOP)                                        AT 02C4
         LOCAT=NUNOPN                                                     AT 02F0
         OPEN(UNIT=LOCAT,FILE=FNAME)                                      AT 02FC
         ICLOSE=1                                                         AT 0327
      ELSE                                                                AT 0331
C
C2A-----DID NOT FIND A RECOGNIZED WORD, SO NOT USING FREE FORMAT.
C2A-----READ THE CONTROL RECORD THE ORIGINAL WAY.
         IFREE=0                                                          AT 0336
         READ(CNTRL,1) LOCAT,CONST,FMTIN,IPRN                             AT 0340
    1    FORMAT(I10,f10.0,A20,I10)                                        AT 03C2
      END IF                                                              AT 03EB
C
C3------FOR FREE FORMAT CONTROL RECORD, READ REMAINING FIELDS.
      IF(IFREE.NE.0) THEN                                                 AT 03EB
         CALL URWORD(CNTRL,ICOL,ISTART,ISTOP,3,N,CONST,IOUT,IN)           AT 03F8
         IF(LOCAT.NE.0) THEN                                              AT 043B
            CALL URWORD(CNTRL,ICOL,ISTART,ISTOP,1,N,R,IOUT,IN)            AT 0448
            FMTIN=CNTRL(ISTART:ISTOP)                                     AT 048B
            IF(ICLOSE.NE.0) THEN                                          AT 04B6
               IF(FMTIN.EQ.'(BINARY)') THEN                               AT 04C3
                  OPEN(UNIT=LOCAT,FILE=FNAME,FORM='UNFORMATTED')          AT 04DB
               ELSE                                                       AT 0516
                  OPEN(UNIT=LOCAT,FILE=FNAME)                             AT 051B
               END IF                                                     AT 0546
            END IF                                                        AT 0546
            IF(LOCAT.GT.0 .AND. FMTIN.EQ.'BINARY') LOCAT=-LOCAT           AT 0546
            CALL URWORD(CNTRL,ICOL,ISTART,ISTOP,2,IPRN,R,IOUT,IN)         AT 0579
         END IF                                                           AT 05BC
      END IF                                                              AT 05BC
C
C4------TEST LOCAT TO SEE HOW TO DEFINE ARRAY VALUES.
      IF(LOCAT) 200,50,90                                                 AT 05BC
C
C4A-----LOCAT=0; SET ALL ARRAY VALUES EQUAL TO CONST. RETURN.
   50 DO 80 I=1,II                                                        AT 05D6
      DO 80 J=1,JJ                                                        AT 0603
   80    a(j,i)=const                                                     AT 0630
      RETURN                                                              AT 065E
C
C4B-----LOCAT>0; READ & WRITE FORMATTED RECORDS USING FORMAT FMTIN.
   90 DO 110 I=1,II                                                       AT 0663
      IF(FMTIN.EQ.'(FREE)') THEN                                          AT 0690
         READ(LOCAT,*) (A(J,I),J=1,JJ)                                    AT 06A8
         if(ip.ne.0)WRITE(IOUT,*) (A(J,I),J=1,JJ)                         AT 071F
      ELSE                                                                AT 079E
         READ(LOCAT,FMTIN) (A(J,I),J=1,JJ)                                AT 07A3
         if(ip.ne.0)WRITE(IOUT,FMTIN) (A(J,I),J=1,JJ)                     AT 0828
      END IF                                                              AT 08B5
  110 CONTINUE                                                            AT 08B5
C
      go to 300                                                           AT 08BA
C
C4C-----LOCAT<0; READ UNFORMATTED RECORD CONTAINING ARRAY VALUES.
  200 LOCAT=-LOCAT                                                        AT 08BF
      READ(LOCAT)                                                         AT 08CD
```

```
           READ(LOCAT) A                                                        AT 08E6
      C
      C5------IF CONST NOT ZERO THEN MULTIPLY ARRAY VALUES BY CONST.
        300 IF(ICLOSE.NE.0) CLOSE(UNIT=LOCAT)                                   AT 0913
            IF(CONST.EQ.0) GO TO 320                                            AT 0939
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
            DO 310 I=1,II                                                       AT 094F
            DO 310 J=1,JJ                                                       AT 097C
            A(J,I)=A(J,I)*CONST                                                 AT 09A9
        310 CONTINUE                                                            AT 09CA
      C
        320 RETURN                                                              AT 09CE
            END                                                                 AT 09D3
      C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<

      C-Environment Agency (UK)-8.Output calculated transmissivities>>>>>>>>>>
      c---------------based on subroutine sbas5h in bas package----
      C-Environment Agency (UK)-11.Output calc trans to binary file>>>>>>>>>>>
      c      subroutine bcf5ot(cc,cr,BUFF,KSTP,KPER,NCOL,NROW,NLAY,IOUT,
      c     1   IXSEC,delc,delr,itrpy,trpy,ihedfm,cv)



            subroutine bcf5ot(cc,cr,BUFF,KSTP,KPER,NCOL,NROW,NLAY,IOUT,
           1   IXSEC,delc,delr,itrpy,trpy,ihedfm,cv,itrans,ioflg,CHEDFM,
           2   LBHDSV,IBOUND,PERTIM,TOTIM)
      C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      C
      C-----VERSION 1647 18OCT1993 bcf5ot
      C     ****************************************************************
      C     PRINT AND RECORD internodal transmissivities
      C     ****************************************************************
      C
      C        SPECIFICATIONS:
      C     ----------------------------------------------------------------
            CHARACTER*16 TEXT
      C-Environment Agency (UK)-11.Output calc trans to binary file>>>>>>>>>>>
      c      DIMENSION cc(NCOL,NROW,NLAY),BUFF(NCOL,NROW,NLAY),
      c     1          cr(ncol,nrow,nlay),trpy(ncol,nrow,nlay),
      c     2          delc(nrow),delr(ncol),cv(ncol,nrow,nlay)
            DIMENSION cc(NCOL,NROW,NLAY),BUFF(NCOL,NROW,NLAY),
           1          cr(ncol,nrow,nlay),trpy(ncol,nrow,nlay),
           2          delc(nrow),delr(ncol),cv(ncol,nrow,nlay),
           3          ioflg(nlay,4),IBOUND(NCOL,NROW,NLAY)
            CHARACTER*20 CHEDFM
      C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      C     ----------------------------------------------------------------
      C
            DO 100 ITRAN=1,3                                                    AT 0079
            if(itran.eq.1)text='X-DIR INTR-TRANS'                               AT 0096
            if(itran.eq.2)text='Y-DIR INTR-TRANS'                               AT 00B6
            if(itran.eq.3)text='Z-DIR INTR-COND.'                               AT 00D6

      C1------FOR EACH LAYER put transmissivity in BUFF
            DO 59 K=1,NLAY                                                      AT 00F6
      C
      C2------Set temporary variables
            KL=K                                                                AT 0123
            IF(IXSEC.NE.0) KL=1                                                 AT 0129
            if(itrpy.eq.0)yx=trpy(k,1,1)                                        AT 013F
      C
      C3------put transmissivity inTO BUFF FOR THE LAYER.
            DO 58 I=1,NROW                                                      AT 0176
            DO 58 J=1,NCOL                                                      AT 01A3
            if (itran.eq.1) then                                                AT 01D0
              if (j.ne.ncol) then                                               AT 01DD
                  BUFF(J,I,K)=cr(J,I,K)*(delr(j+1)+delr(j))/(2*delc(i))         AT 01E8
              else                                                              AT 0233
                buff(j,i,k)=0                                                   AT 0238
              end if                                                            AT 026B
            elseif(itran.eq.2)then                                              AT 026B
              if (i.ne.nrow) then                                               AT 027D
                  BUFF(J,I,K)=cc(J,I,K)*(delc(i+1)+delc(i))/(2*delr(j))         AT 028E
              else                                                              AT 02D9
                buff(j,i,k)=0                                                   AT 02DE
              end if                                                            AT 0311
            else                                                                AT 0311
              if (k.ne.nlay) then                                               AT 0316
                  BUFF(J,I,K)=cv(J,I,K)/(delc(i)*delr(j))                       AT 0327
              else                                                              AT 036E
                goto 100                                                        AT 0373
              end if                                                            AT 0378
```

```
      end if                                                                   AT 0378
   58 CONTINUE                                                                 AT 0378
   59 CONTINUE                                                                 AT 0382
C
C4------FOR EACH LAYER PRINT x- and y-dir internodal transmissivities.
C-Environment Agency (UK)-11.Output calc trans to binary file>>>>>>>>>>>
      if(itrans.le.0) goto 70                                                  AT 0387
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      IF(IXSEC.EQ.0) THEN                                                      AT 0393
        DO 69 K=1,NLAY                                                         AT 039F
        KK=K                                                                   AT 03CC
        if(ihedfm.lt.0)CALL ULAPRS(BUFF(1,1,K),TEXT,KSTP,KPER,NCOL,NROW,       AT 03D2
     1                            KK,-ihedfm,IOUT)
        if(ihedfm.ge.0)CALL ULAPRW(BUFF(1,1,K),TEXT,KSTP,KPER,NCOL,NROW,       AT 043A
     1                            KK,ihedfm,IOUT)
   69   CONTINUE                                                               AT 0498
C
C4A-----PRINT transmissivities FOR CROSS SECTION (?).
      ELSE                                                                     AT 049D
        if(ihedfm.lt.0)CALL ULAPRS(BUFF,TEXT,KSTP,KPER,NCOL,NLAY,-1,           AT 04A2
     1                             -ihedfm,IOUT)
        if(ihedfm.ge.0)CALL ULAPRW(BUFF,TEXT,KSTP,KPER,NCOL,NLAY,-1,           AT 04EB
     1                            ihedfm,IOUT)
      END IF                                                                   AT 0526
C-Environment Agency (UK)-11.Output calc trans to binary file>>>>>>>>>>>
C
C5------FOR EACH LAYER: DETERMINE IF HEAD SHOULD BE SAVED ON DISK.
C5------IF SO THEN CALL ULASAV OR ULASV2 TO SAVE HEAD.
 70   IFIRST=1                                                                 AT 0526
      IF(itrans.ge.0) GO TO 100                                                AT 0530
      IF(IXSEC.EQ.0) THEN                                                      AT 053C
        DO 79 K=1,NLAY                                                         AT 0548
        KK=K                                                                   AT 0575
        IF(IOFLG(K,3).EQ.0) GO TO 79                                           AT 057B
        IF(IFIRST.EQ.1) WRITE(IOUT,74) text,-itrans,KSTP,KPER                  AT 059E
   74   FORMAT(1X,/1X,a16,' WILL BE SAVED ON UNIT',I4,                         AT 062C
     1      ' AT END OF TIME STEP',I3,', STRESS PERIOD',I3)
        IFIRST=0                                                               AT 06A5
        IF(CHEDFM.EQ.' ') THEN                                                 AT 06AF
           CALL ULASAV(BUFF(1,1,K),TEXT,KSTP,KPER,PERTIM,TOTIM,NCOL,           AT 06C3
     1                NROW,KK,-itrans)
        ELSE                                                                   AT 0729
           CALL ULASV2(BUFF(1,1,K),TEXT,KSTP,KPER,PERTIM,TOTIM,NCOL,           AT 072E
     1                NROW,KK,-itrans,CHEDFM,LBHDSV,IBOUND(1,1,K))
        END IF                                                                 AT 07BA
   79   CONTINUE                                                               AT 07BA
C
C5A-----SAVE HEAD FOR CROSS SECTION.
      ELSE                                                                     AT 07BF
        IF(IOFLG(1,3).NE.0) THEN                                               AT 07C4
          WRITE(IOUT,74) text,-itrans,KSTP,KPER                                AT 07E5
          IF(CHEDFM.EQ.' ') THEN                                               AT 0866
             CALL ULASAV(BUFF,TEXT,KSTP,KPER,PERTIM,TOTIM,NCOL,                AT 087A
     1                NLAY,-1,-itrans)
          ELSE                                                                 AT 08BD
             CALL ULASV2(BUFF,TEXT,KSTP,KPER,PERTIM,TOTIM,NCOL,                AT 08C2
     1                 NLAY,-1,-itrans,CHEDFM,LBHDSV,IBOUND)
          END IF                                                               AT 0914
        END IF                                                                 AT 0914
      END IF                                                                   AT 0914
C
  100 continue                                                                 AT 0914
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
c5------return
      return                                                                   AT 0919
      end                                                                      AT 091E
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<

C-Environment Agency (UK)-3.Auto-conversion for steady state VKD>>>>>>>>


      subroutine SBCF5W(A,IBOUND,HNEW,NCOL,NROW,NLAY,K,INBCF,FMT,ANAME,
     1                  IBCFOUT,HDRY)
C
C-----VERSION 0001 15FEB2001 SBCF5W
C     ******************************************************************
C     WRITES ARRAYS TO SECOND BCF FILE
C     ******************************************************************
C
C        SPECIFICATIONS:
```

```
C     ------------------------------------------------------------------
      double precision HNEW
      dimension A(NCOL,NROW),IBOUND(NCOL,NROW,NLAY),HNEW(NCOL,NROW,NLAY)
      character FMT*20,ANAME*24
C     ------------------------------------------------------------------
C
      IFIRST=1                                                           AT 005F
      ICONST=1                                                           AT 0066
C
C1------LOOP THROUGH ALL CELLS
      do J=1,NCOL                                                        AT 006D
        do I=1,NROW                                                      AT 008B
C
C2------IF K > 0 THEN 2-D ARRAY, SET VALUES TO ZERO AT DRY/NO-FLOW CELLS
C2------(THIS AVOIDS THE LIKELIHOOD OF CELLS WITH UNREALISTIC VKD
C2------PARAMETERS BECOMING ACTIVE DUE TO REWETTING IN SUBSEQUENT
C2------SIMULATIONS).
          if(K.gt.0)then                                                 AT 00A9
            if(IBOUND(J,I,K).eq.0.or.real(HNEW(J,I,K)).eq.HDRY)then      AT 00B5
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
              A(J,I)=0.0                                                 AT 00FD
              goto 10                                                    AT 011D
            endif                                                        AT 0122
          endif                                                          AT 0122
C
C3------CHECK IF ALL THE ACTIVE VALUES IN THE ARRAY ARE THE SAME.
          if(ICONST.eq.1)then                                            AT 0122
            if(IFIRST.eq.1)then                                          AT 012C
              ALAST=A(J,I)                                               AT 0136
              IFIRST=0                                                   AT 015A
            else                                                         AT 0161
              if(A(J,I).ne.ALAST)then                                    AT 0166
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
                ICONST=0                                                 AT 0190
                ALAST=1.0                                                AT 0197
              endif                                                      AT 01AB
            endif                                                        AT 01AB
          endif                                                          AT 01AB
 10       continue                                                       AT 01AB
        enddo                                                            AT 01AB
      enddo                                                              AT 01B0
C
C4------IF ALL ACTIVE VALUES ARE THE SAME WRITE A SINGLE VALUE.
      if(ICONST.eq.1)then                                                AT 01B5
        write(IBCFOUT,56) 0,ALAST,FMT,ANAME,K                            AT 01BF
      else                                                               AT 0247
C
C5------OTHERWISE WRITE OUT THE FULL ARRAY.
        write(IBCFOUT,56) INBCF,ALAST,FMT,ANAME,K                        AT 024C
        do I=1,NROW                                                      AT 02D2
          write(IBCFOUT,FMT) (A(J,I),J=1,NCOL)                           AT 02F6
        enddo                                                            AT 036B
      endif                                                              AT 0370
C
C-------HEADER FORMAT (PRINTING TO 2ND OUTPUT FILE IS NOT REPRESSED).
   56 format(i10,g10.3,a20,8x,' 0',a24,' Layer',i4)                      AT 0370
C
C6------RETURN
      return                                                             AT 03B3
      end                                                                AT 03B8
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
```

End of compilation - Clocked 0.1 seconds

APPENDIX B
Modified computer code for stream package with changes highlighted

Salford Software Ltd FTN77 Ver. 3.53   C:\Company\Tech\MF-VKD\Source_code\str-vkd1.for   14:34:01
Monday, 8 September 2003

COMPILER OPTIONS: listing intl no_persist nodclvar nomap nocheck logl dynm offset dreal noansi nopagethrow
                nosilent no_optimise warn73 nolink no_link

```
C-Changes to the code are highlighted with the following headings:
C
C-Environment Agency (UK)-13.Allow an inflow to any stream reach>>>>>>>>
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
C-Environment Agency (UK)-15.Save single flow if >1 stream in a cell>>>>
C
C These headings are followed by the original USGS MODFLOW-96 code
C (commented out), followed by the modified code. All the code changes
C are followed by the following:
C
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
C      SUBROUTINE STR1AL(ISUM,LENX,LCSTRM,ICSTRM,MXSTRM,NSTREM,IN,
C     1                   IOUT,ISTCB1,ISTCB2,NSS,NTRIB,NDIV,ICALC,CONST,
C     2                   LCTBAR,LCTRIB,LCIVAR,LCFGAR)
      SUBROUTINE STR1AL(ISUM,LENX,LCSTRM,ICSTRM,MXSTRM,NSTREM,IN,
     1                   IOUT,ISTCB1,ISTCB2,NSS,NTRIB,NDIV,ICALC,CONST,
     2                   LCTBAR,LCTRIB,LCIVAR,LCFGAR,iswabs,lctba2)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C                                                                      C
C-----VERSION    2 18DEC1990 STR1AL                                    C
C     ****************************************************************C
C     ALLOCATE ARRAY STORAGE FOR STREAMS                               C
C     ****************************************************************C
C                                                                      C
C     SPECIFICATIONS:                                                  C
C     -----------------------------------------------------------------C
C     -----------------------------------------------------------------C
C                                                                      C
C1------IDENTIFY PACKAGE AND INITIALIZE NSTREM.                        C
      WRITE(IOUT,1) IN                                                          AT 0018
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
c    1 FORMAT(1H0,'STRM -- STREAM PACKAGE, VERSION 2, 12/18/90 ',
c     1'INPUT READ FROM UNIT',I3)
    1 FORMAT(1H0,'STR-VKD1 -- STREAM PACKAGE, Allows inflows/outflows',         AT 004E
     1' and tributary inflows at any stream cell, VERSION 1, 18/7/2001',
     1'INPUT READ FROM UNIT',I3)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      NSTREM=0                                                                  AT 00FD
C                                                                      C
C2------ READ MXSTRM, NSS, NTRIB, ISTCB1, AND ISTCB2.                  C
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
c  100 READ(IN,3)MXSTRM,NSS,NTRIB,NDIV,ICALC,CONST,ISTCB1,ISTCB2
c    3 FORMAT(5I10,F10.0,2I10)
  100 READ(IN,3)MXSTRM,NSS,NTRIB,NDIV,ICALC,CONST,ISTCB1,ISTCB2,iswabs          AT 0106
    3 FORMAT(5I10,F10.0,3I10)                                                   AT 01BC
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      IF(MXSTRM.LT.0)MXSTRM=0                                                   AT 01DF
      IF(NSS.LT.0)NSS=0                                                         AT 01F1
      WRITE(IOUT,4)MXSTRM,NSS,NTRIB                                             AT 0203
    4 FORMAT(1H ,'MAXIMUM OF',I5,' STREAM NODES'//1X,'NUMBER OF STREAM S        AT 0259
     1EGMENTS IS ',I5//1X,'NUMBER OF STREAM TRIBUTARIES IS ',I5//)
      IF(NDIV.GT.0) WRITE(IOUT,5)                                               AT 02FA
    5 FORMAT(1H ,'DIVERSIONS FROM STREAMS HAVE BEEN SPECIFIED')                 AT 032C
      IF(ICALC.GT.0) WRITE(IOUT,6) CONST                                        AT 0371
    6 FORMAT(1H ,'STREAM STAGES WILL BE CALCULATED USING A CONSTANT OF          AT 03B3
     1',F10.4)
      IF(ISTCB1.GT.0) WRITE(IOUT,7) ISTCB1,ISTCB2                               AT 040A
    7 FORMAT(1X,'CELL BUDGETS WILL BE SAVED ON UNITS',I3,'AND',I3)              AT 045C
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
      if(iswabs.lt.0) write(iout,11)                                            AT 04AB
   11 format(1X,'SURFACE WATER ABSTRACTIONS AND DISCHARGES ALLOWED IN *A        AT 04DD
     1NY* STREAM CELL')
      if(iswabs.gt.0) write(iout,12)                                            AT 053A
   12 format(1X,'SURFACE WATER ABSTRACTIONS, DISCHARGES AND TRIBUTARY IN        AT 056C
     1FLOWS ALLOWED IN *ANY* STREAM CELL')
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C                                                                      C
C3------SET LCSTRM EQUAL TO ADDRESS OF FIRST UNUSED SPACE IN X.        C
  200 LCSTRM=ISUM                                                               AT 05DD
C                                                                      C
C4------CALCULATE AMOUNT OF SPACE NEEDED FOR STRM LIST.                C
      ISPA=11*MXSTRM                                                            AT 05E7
      ISUM=ISUM+ISPA                                                            AT 05F7
```

```
C                                                                    C
C5------CALCULATE AMOUNT OF SPACE NEEDED FOR ISTRM LIST.             C
      ICSTRM=ISUM                                                              AT 05F9
      ISPB=5*MXSTRM                                                            AT 0600
      ISUM=ISUM+ISPB                                                           AT 0609
C                                                                    C
C6------CALCULATE AMOUNT OF SPACE NEEEDED FOR ITRBAR LIST.            C
      LCTBAR=ISUM                                                              AT 060E
      ISPC=NSS*NTRIB                                                           AT 0615
      ISUM=ISUM+ISPC                                                           AT 0623
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
      lctba2=isum                                                              AT 0628
      if(iswabs.gt.0)isum=isum+ispc                                            AT 062F
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C                                                                    C
C7------CALCULATE AMOUNT OF SPACE NEEDED FOR ARTRIB LIST.             C
      LCTRIB=ISUM                                                              AT 0643
      ISPD=NSS                                                                 AT 064D
      ISUM=ISUM+ISPD                                                           AT 0655
C                                                                    C
C8A-----CALCULATE AMOUNT OF SPACE NEEDED FOR IDIVAR LIST.             C
      LCIVAR=ISUM                                                              AT 0657
      ISPE=NSS                                                                 AT 065E
      ISUM=ISUM+ISPE                                                           AT 0663
C                                                                    C
C8B-----CALCULATE AMOUNT IF SPACE NEEDED FOR NDFGAR LIST.             C
      LCFGAR=ISUM                                                              AT 0668
      ISPF=NSS                                                                 AT 066F
      ISUM=ISUM+ISPF                                                           AT 0674
      ISP=ISPA+ISPB+ISPC+ISPD+ISPE+ISPF                                        AT 0679
C                                                                    C
C9------PRINT AMOUNT OF SPACE USED BY STREAM PACKAGE.                 C
      WRITE (IOUT,8)ISP                                                        AT 068A
    8 FORMAT(1X,I8,' ELEMENTS IN X ARRAY ARE USED FOR STREAMS')                AT 06C1
      ISUM1=ISUM-1                                                             AT 0708
      WRITE(IOUT,9)ISUM1,LENX                                                  AT 0714
    9 FORMAT(1X,I8,' ELEMENTS OF X ARRAY USED OUT OF',I7)                      AT 075E
      IF(ISUM1.GT.LENX) WRITE(IOUT,10)                                         AT 07A1
   10 FORMAT(1X,'   ***X ARRAY MUST BE DIMENSIONED LARGER***')                 AT 07D8
C                                                                    C
C10-----RETURN.                                                       C
      RETURN                                                                   AT 081B
      END                                                                      AT 0820
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
c      SUBROUTINE STR1RP(STRM,ISTRM,NSTREM,MXSTRM,IN,IOUT,ITRBAR,NDIV,
c     1                   NSS,NTRIB,IDIVAR,ICALC,IPTFLG)
WARNING - Label 200 has not been referenced
WARNING - Label 100 has not been referenced


      SUBROUTINE STR1RP(STRM,ISTRM,NSTREM,MXSTRM,IN,IOUT,ITRBAR,NDIV,
     1                   NSS,NTRIB,IDIVAR,ICALC,IPTFLG,iswabs,itrba2)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C                                                                    C
C                                                                    C
C-----VERSION  2  18DEC1990 STR1RP                                    C
C     ***************************************************************C
C     READ STREAM DATA:  INCLUDES SEGMENT AND REACH NUMBERS, CELL     C
C         SEQUENCE OF SEGMENT AND REACH, FLOW INTO MODEL AT BOUNDARY, C
C         STREAM STAGE, STREAMBED CONDUCTANCE, AND STREAMBED TOP AND  C
C         BOTTOM ELEVATIONS                                           C
C     ***************************************************************C
C                                                                    C
C     SPECIFICATIONS:                                                C
C     ---------------------------------------------------------------C
      DIMENSION STRM(11,MXSTRM),ISTRM(5,MXSTRM),ITRBAR(NSS,NTRIB),
     1          IDIVAR(NSS)
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
      dimension itrba2(nss,ntrib)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C     ---------------------------------------------------------------C
C                                                                    C
C1A-----IF MXSTREAM IS LESS THAN 1 THEN STREAM IS INACTIVE. RETURN.   C
      IF(MXSTRM.LT.1) RETURN                                                   AT 0061
C                                                                    C
C1B-----READ ITMP(NUMBER OF STREAM CELLS OR FLAG TO REUSE DATA).      C
      READ(IN,1)ITMP,IRDFLG,IPTFLG                                             AT 006D
    1 FORMAT(3I10)                                                             AT 00C5
C                                                                    C
C2A-----IF ITMP <0 THEN REUSE DATA FROM LAST STRESS PERIOD.           C
      IF(ITMP.GE.0)GO TO 50                                                    AT 00DA
```

```
      WRITE(IOUT,2)                                                           AT 00E4
    2 FORMAT(1H0,'REUSING STREAM NODES FROM LAST STRESS PERIOD')              AT 010A
      RETURN                                                                  AT 014F
C                                                                       C
C2B-----IF ITMP=> ZERO THEN IT IS THE NUMBER OF STREAM REACHES.         C
   50 NSTREM=ITMP                                                             AT 0154
C                                                                       C
C3A-----IF NSTREM>MXSTRM THEN STOP.                                     C
      IF(NSTREM.LE.MXSTRM)GO TO 100                                           AT 015C
      WRITE(IOUT,99)NSTREM,MXSTRM                                             AT 0167
   99 FORMAT(1H0,'NSTREM(',I4,') IS GREATER THAN MXSTRM(',I4,')')             AT 01AD
      STOP                                                                    AT 01FE
C                                                                       C
C3B-----PRINT NUMBER OF STREAM CELLS IN THIS STRESS PERIOD.             C
  100 IF(IRDFLG.EQ.0) WRITE(IOUT,3)NSTREM                                     AT 0211
    3 FORMAT(1H0,//1X,I5,' STREAM NODES')                                     AT 0251
C                                                                       C
C4------IF THERE ARE NO STREAM CELLS THEN RETURN.                       C
      IF(NSTREM.EQ.0) RETURN                                                  AT 0286
C                                                                       C
C5------READ AND PRINT DATA FOR EACH STREAM CELL.                       C
      IF(IRDFLG.EQ.0) WRITE(IOUT,4)                                           AT 0292
    4 FORMAT(1H ,3X,'LAYER   ROW   COL   SEGMENT   REACH   STREAMFLOW         AT 02C2
     1     STREAM    STREAMBED     STREAMBED BOT  STREAMBED TOP',/27X,
     2'NUMBER   NUMBER                    STAGE   CONDUCTANCE      ELEVAT
     3ION     ELEVATION',/3X,110('-'))
      DO 250 II=1,NSTREM                                                      AT 03C3
      READ(IN,5)K,I,J,ISTRM(4,II),ISTRM(5,II),STRM(1,II),STRM(2,II),          AT 03E1
     1STRM(3,II),STRM(4,II),STRM(5,II)
    5 FORMAT(5I5,F15.0,4F10.0)                                                AT 0555
      IF(IRDFLG.EQ.0) WRITE(IOUT,6)K,I,J,ISTRM(4,II),ISTRM(5,II),             AT 057A
     1STRM(1,II),STRM(2,II),STRM(3,II),STRM(4,II),STRM(5,II)
    6 FORMAT(1X,3X,I4,2I7,2I9,7X,G11.4,G12.4,G11.4,4X,2G13.4)                 AT 06F8
      ISTRM(1,II)=K                                                           AT 0749
      ISTRM(2,II)=I                                                           AT 075B
      ISTRM(3,II)=J                                                           AT 076F
  250 CONTINUE                                                                AT 0783
C                                                                       C
C6----READ AND PRINT DATA IF STREAM STAGE IS CALCULATED.                C
      IF(ICALC.LE.0) GO TO 300                                                AT 0788
      IF(IRDFLG.EQ.0) WRITE(IOUT,7)                                           AT 0794
    7 FORMAT(1H0,3X,'LAYER',3X,'ROW',4X,'COL   ',' SEGMENT',3X,               AT 07C4
     1'REACH',8X,'STREAM',13X,'STREAM',10X,'ROUGH',/27X,'NUMBER',3X,
     2 'NUMBER',8X,'WIDTH',14X,'SLOPE',10X,'COEF.',/3X,110('-'))
      DO 280 II=1,NSTREM                                                      AT 08A7
      READ(IN,8) STRM(6,II),STRM(7,II),STRM(8,II)                             AT 08CB
    8 FORMAT(3F10.0)                                                          AT 096C
      IF(IRDFLG.EQ.0) WRITE(IOUT,9)ISTRM(1,II),ISTRM(2,II),ISTRM(3,II),       AT 0983
     1ISTRM(4,II),ISTRM(5,II),STRM(6,II),STRM(7,II),STRM(8,II)
    9 FORMAT(4X,I4,2I7,2I9,7X,G12.4,4X,G13.4,4X,G12.4)                        AT 0AE5
  280 CONTINUE                                                                AT 0B2E
C                                                                       C
C7------INITIALIZE ALL TRIBUTARY SEGMENTS TO ZERO.                      C
  300 DO 320 IK=1,NSS                                                         AT 0B33
      DO 320 JK=1,NTRIB                                                       AT 0B60
      ITRBAR(IK,JK)=0                                                         AT 0B8D
  320 CONTINUE                                                                AT 0BA7
C                                                                       C
C8-----INITIALIZE DIVERSION SEGMENT ARRAY TO ZERO.                      C
      DO 325 IK=1,NSS                                                         AT 0BAB
      IDIVAR(IK)=0                                                            AT 0BD8
  325 CONTINUE                                                                AT 0BE3
C                                                                       C
C9-----READ AND PRINT TRIBUTARY SEGMENTS.                               C
      IF(NTRIB.LE.0) GO TO 343                                                AT 0BE5
      IF(IRDFLG.EQ.0) WRITE(IOUT,10)NTRIB                                     AT 0BF1
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
c   10 FORMAT(1H0,30X,'MAXIMUM NUMBER OF TRIBUTARY STREAMS IS ',I5,//1X,
c      1 20X,'STREAM SEGMENT',15X,'TRIBUTARY STREAM SEGMENT NUMBERS')
c       DO 340 IK=1,NSS
c       READ(IN,11) (ITRBAR(IK,JK),JK=1,NTRIB)
c   11 FORMAT(10I5)
c       IF(IRDFLG.EQ.0) WRITE(IOUT,12)IK,(ITRBAR(IK,JK),JK=1,NTRIB)
c   12 FORMAT(20X,I5,20X,10I5)
   10 FORMAT(1H0,30X,'MAXIMUM NUMBER OF TRIBUTARY STREAMS IS ',I5,//1X,       AT 0C31
     1 20X,'STREAM SEGMENT',15X,'TRIBUTARY SEGMENTS (INTO REACH)')
      DO 340 IK=1,NSS                                                         AT 0CC2
      if(iswabs.gt.0)goto 331                                                 AT 0CEF
C                                                                       C
C9A----USE ORIGINAL FORMAT                                              C
      READ(IN,11) (ITRBAR(IK,JK),JK=1,NTRIB)                                  AT 0CFB
   11 FORMAT(10I5)                                                            AT 0D7A
```

```
      IF(IRDFLG.EQ.0) WRITE(IOUT,12)IK,(ITRBAR(IK,JK),1,JK=1,NTRIB)         AT 0D8F
      goto 340                                                              AT 0E3E
C                                                                     C
C9B----READ IN INFLOW REACH AND TRIBUTARY SEGMENTS                     C
331   READ(IN,16) (itrba2(ik,jk),ITRBAR(IK,JK),JK=1,NTRIB)                  AT 0E43
   16 FORMAT(10(i5,i5))                                                     AT 0EF0
      IF(IRDFLG.EQ.0) WRITE(IOUT,12)IK,(ITRBAR(IK,JK),itrba2(ik,jk),        AT 0F15
     1      JK=1,NTRIB)
   12 FORMAT(20X,I5,20X,10(I5,'(',i5,') '))                                 AT 0FE0
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
  340 CONTINUE                                                              AT 101F
C                                                                     C
C10----READ AND PRINT DIVERSION SEGMENTS NUMBERS.                      C
  343 IF(NDIV.LE.0) GO TO 350                                               AT 1024
      IF(IRDFLG.EQ.0) WRITE(IOUT,13)                                        AT 1030
   13 FORMAT(1H0,10X,'DIVERSION SEGMENT NUMBER',10X,                        AT 1060
     1       'UPSTREAM SEGMENT NUMBER')
     DO 345 IK=1,NSS                                                        AT 10B5
      READ(IN,14) IDIVAR(IK)                                                AT 10E2
   14 FORMAT(I10)                                                           AT 1123
      IF(IRDFLG.EQ.0) WRITE(IOUT,15) IK,IDIVAR(IK)                          AT 1138
   15 FORMAT(20X,I5,28X,I5)                                                 AT 1197
  345 CONTINUE                                                              AT 11BA
C                                                                     C
C11----SET FLOW OUT OF REACH, FLOW INTO REACH, AND FLOW THROUGH        C
C      STREAM BED TO ZERO.                                             C
  350 DO 360 II =1,NSTREM                                                   AT 11BF
      STRM(9,II)=0.0                                                        AT 11E3
      STRM(10,II)=0.0                                                       AT 1201
      STRM(11,II)=0.0                                                       AT 121F
  360 CONTINUE                                                              AT 123D
C                                                                     C
C12------RETURN                                                       C
      RETURN                                                                AT 123F
      END                                                                   AT 1244
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
c      SUBROUTINE STR1FM(NSTREM,STRM,ISTRM,HNEW,HCOF,RHS,IBOUND,MXSTRM,
c     1                 NCOL,NROW,NLAY,IOUT,NSS,ITRBAR,NTRIB,ARTRIB,
c     2                 IDIVAR,NDFGAR,ICALC,CONST)


      SUBROUTINE STR1FM(NSTREM,STRM,ISTRM,HNEW,HCOF,RHS,IBOUND,MXSTRM,
     1                 NCOL,NROW,NLAY,IOUT,NSS,ITRBAR,NTRIB,ARTRIB,
     2                 IDIVAR,NDFGAR,ICALC,CONST,iswabs,itrba2)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C                                                                     C
C-----VERSION   2 18DEC1990 STR1FM                                    C
C                                                                     C
C     ***************************************************************C
C     ADD STREAM TERMS TO RHS AND HCOF IF FLOW OCCURS IN MODEL CELL   C
C     ***************************************************************C
C                                                                     C
C     SPECIFICATIONS:                                                 C
C     ----------------------------------------------------------------C
C                                                                     C
      DOUBLE PRECISION HNEW
      DIMENSION STRM(11,MXSTRM),ISTRM(5,MXSTRM),HNEW(NCOL,NROW,NLAY),
     1          HCOF(NCOL,NROW,NLAY),RHS(NCOL,NROW,NLAY),
     2          IBOUND(NCOL,NROW,NLAY),ITRBAR(NSS,NTRIB),ARTRIB(NSS),
     3          IDIVAR(NSS),NDFGAR(NSS)
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
      dimension itrba2(nss,ntrib)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C     ----------------------------------------------------------------C
C                                                                     C
C1------IF NSTREM<=0 THERE ARE NO STREAMS. RETURN.                     C
      IF(NSTREM.LE.0)RETURN                                                 AT 00B6
C                                                                     C
C2A-----PROCESS EACH CELL IN THE STREAM LIST.                          C
C2B-----INITIALIZE NDFGAR ARRAY TO ZERO.                               C
      DO 5 I=1,NSS                                                          AT 00C2
      NDFGAR(I)=0                                                           AT 00EF
    5 CONTINUE                                                              AT 00FA
C                                                                     C
C3------DETERMINE LAYER, ROW, COLUMN OF EACH REACH.                    C
      DO 500 L=1,NSTREM                                                     AT 00FC
      LL=L-1                                                                AT 0129
      IL=ISTRM(1,L)                                                         AT 0130
      IR=ISTRM(2,L)                                                         AT 0148
      IC=ISTRM(3,L)                                                         AT 015F
C                                                                     C
```

```
C4----06FEB1990, CHECK FOR CELLS OUTSIDE MOVED TO C12, C16 AND C18.    C
C                                                                      C
C5------DETERMINE STREAM SEGMENT AND REACH NUMBER.                     C
      ISTSG=ISTRM(4,L)                                                           AT 0176
      NREACH=ISTRM(5,L)                                                          AT 018D
C                                                                      C
C-Environment Agency (UK)-13.Allow an inflow to any stream reach>>>>>>>>
cC6------SET FLOWIN EQUAL TO STREAM SEGMENT INFLOW IF FIRST REACH.      C
c      IF(NREACH.GT.1) GO TO 200
c      FLOWIN=STRM(1,L)
C6------SET FLOWIN EQUAL TO STREAM SEGMENT INFLOW IF any REACH.        C
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
c      FLOWIN=STRM(1,L)
c      IF(NREACH.GT.1) GO TO 200
      IF(iswabs.eq.0.and.NREACH.GT.1) GO TO 200                                  AT 01A4
      FLOWIN=STRM(1,L)                                                           AT 01B9
      IF(iswabs.lt.0.and.NREACH.GT.1) GO TO 200                                  AT 01DF
      IF(NREACH.GT.1) GO TO 60                                                   AT 01F8
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C                                                                      C
C7------STORE OUTFLOW FROM PREVIOUS SEGMENT IN ARTRIB IF SEGMENT >1.   C
      IF(ISTSG.EQ.1)GO TO 50                                                     AT 0205
      IFLG = ISTRM(4,LL)                                                         AT 0212
      ARTRIB(IFLG)=STRM(9,LL)                                                    AT 022C
C                                                                      C
C8A-----CHECK UPSTREAM SEGMENT FOR DIVERSIONS.                          C
      DO 40 NSFLG = 1,NSS                                                        AT 0259
      IF(IFLG.NE.IDIVAR(NSFLG)) GO TO 40                                         AT 0286
C                                                                      C
C8B-----DETERMINE AMOUNT OF FLOW TO BE DIVERTED.                        C
      DO 20 IDL=1,NSTREM                                                         AT 0299
      IF(NSFLG.NE.ISTRM(4,IDL)) GO TO 20                                         AT 02C6
      IF(ISTRM(5,IDL).NE.1) GO TO 20                                             AT 02E6
      DUM=ARTRIB(IFLG)-STRM(1,IDL)                                               AT 0304
C                                                                      C
C8C-----SUBTRACT FLOW FROM UPSTREAM SEGMENT IF THERE IS ENOUGH FLOW     C
C-------IN UPSTREAM SEGMENT.                                           C
      IF(DUM.GE.0.0) ARTRIB(IFLG)=DUM                                            AT 032E
      IF(DUM.LT.0.0) NDFGAR(IFLG)=1                                              AT 0358
   20 CONTINUE                                                                   AT 037F
   40 CONTINUE                                                                   AT 0384
   50 IF(IDIVAR(ISTSG).LE.0) GO TO 60                                            AT 0389
      NDFLG=IDIVAR(ISTSG)                                                        AT 039D
      IF(NDFGAR(NDFLG).EQ.1) FLOWIN=0.0                                          AT 03A7
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
c   60 IF(FLOWIN.GE.0.0) GO TO 300
   60 IF(FLOWIN.GE.0.0.and.iswabs.eq.0) GO TO 300                                AT 03C9
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C                                                                      C
C9-----SUM TRIBUTARY OUTFLOW AND USE AS INFLOW INTO DOWNSTREAM SEGMENT.C
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
c      FLOWIN =0.
      if(iswabs.eq.0)FLOWIN =0.                                                  AT 03EB
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      DO 100 ITRIB=1,NTRIB                                                       AT 040B
      INODE=ITRBAR(ISTSG,ITRIB)                                                  AT 0438
      IF(INODE.LE.0) GO TO 100                                                   AT 045A
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
      if(iswabs.gt.0)then                                                        AT 0463
        if(itrba2(istsg,itrib).ne.nreach)goto 100                                AT 046F
      endif                                                                      AT 049D
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      FLOWIN=FLOWIN+ARTRIB(INODE)                                                AT 049D
  100 CONTINUE                                                                   AT 04B7
C                                                                      C
C-Environment Agency (UK)-13.Allow an inflow to any stream reach>>>>>>>>
cC10-----IF REACH >1, SET INFLOW EQUAL TO OUTFLOW FROM UPSTREAM REACH.  C
c  200 IF(NREACH.GT.1) FLOWIN=STRM(9,LL)
C10-----IF REACH >1, add OUTFLOW FROM UPSTREAM REACH to inflow.         C
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
c  200 IF(NREACH.GT.1) FLOWIN=flowin+STRM(9,LL)
  200 if(iswabs.eq.0)then                                                        AT 04BC
        IF(NREACH.GT.1) FLOWIN=STRM(9,LL)                                        AT 04C8
      else                                                                       AT 04FD
        IF(NREACH.GT.1) FLOWIN=flowin+STRM(9,LL)                                 AT 0502
      endif                                                                      AT 0534
C                                                                      C
C10a---LIMIT SURFACE WATER ABSTRACTION IF NOT ENOUGH FLOW IN STREAM     C
      if(iswabs.ne.0.and.flowin.lt.0)then                                        AT 0534
        flowin=0.0                                                               AT 0556
      endif                                                                      AT 056A
```

```
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C                                                                      C
C11----COMPUTE STREAM STAGE IN REACH IF ICALC IS GREATER THAN 1.       C
  300 IF(ICALC.LE.0) GO TO 310                                                   AT 056A
      XNUM=((FLOWIN+STRM(9,L))/2.0)*STRM(8,L)                                    AT 0576
      DNOM=CONST*STRM(6,L)*(SQRT(STRM(7,L)))                                     AT 05B4
      DEPTH=(XNUM/DNOM)**0.6                                                     AT 05EA
      IF(DEPTH.LE.0.) DEPTH=0.                                                   AT 060F
      STRM(2,L)=DEPTH+STRM(5,L)                                                  AT 0633
  310 HSTR=STRM(2,L)                                                             AT 0665
C                                                                      C
C12----DETERMINE LEAKAGE THROUGH STREAMBED.                            C
      IF(IBOUND(IC,IR,IL).LE.0) GO TO 315                                        AT 068D
      IF(FLOWIN.LE.0.) HSTR=STRM(5,L)                                            AT 06C1
      CSTR=STRM(3,L)                                                             AT 06FC
      SBOT=STRM(4,L)                                                             AT 0724
      H=HNEW(IC,IR,IL)                                                           AT 0749
      T=HSTR-SBOT                                                                AT 0780
C                                                                      C
C13----COMPUTE LEAKAGE AS A FUNCTION OF STREAM STAGE AND HEAD IN CELL. C
      FLOBOT=CSTR*(HSTR-H)                                                       AT 0793
C                                                                      C
C14----RECOMPUTE LEAKAGE IF HEAD IN CELL IS BELOW STREAMBED BOTTOM.    C
      IQFLG=0                                                                    AT 07AC
      IF(H.GT.SBOT) GO TO 312                                                    AT 07B6
      IQFLG=1                                                                    AT 07CC
      FLOBOT=CSTR*T                                                              AT 07D6
C                                                                      C
C15----SET LEAKAGE EQUAL TO STREAM INFLOW IF LEAKAGE MORE THAN INFLOW. C
  312 IF(FLOBOT.LE.FLOWIN) GO TO 320                                             AT 07E9
      IQFLG=1                                                                    AT 07FF
      FLOBOT=FLOWIN                                                              AT 0809
C                                                                      C
C16-----STREAMFLOW OUT EQUALS STREAMFLOW IN MINUS LEAKAGE.             C
  315 IF(IBOUND(IC,IR,IL).LE.0) FLOBOT=0.                                        AT 0821
  320 FLOWOT=FLOWIN-FLOBOT                                                       AT 0869
      IF((ISTSG.GT.1).AND.(NREACH.EQ.1)) STRM(9,LL)=ARTRIB(IFLG)                 AT 087C
C                                                                      C
C17----STORE STREAM INFLOW, OUTFLOW AND LEAKAGE FOR EACH REACH.        C
      STRM(9,L)=FLOWOT                                                           AT 08C3
      STRM(10,L)=FLOWIN                                                          AT 08EB
      STRM(11,L)=FLOBOT                                                          AT 0910
C                                                                      C
C18----RETURN TO STEP 3 IF STREAM INFLOW IS LESS THAN OR EQUAL TO ZERO C
C        AND LEAKAGE IS GREATER THAN OR EQUAL TO ZERO OR IF CELL       C
C        IS NOT ACTIVE--IBOUND IS LESS THAN OR EQUAL TO ZERO.          C
      IF(IBOUND(IC,IR,IL).LE.0) GO TO 500                                        AT 0935
      IF((FLOWIN.LE.0.0).AND.(FLOBOT.GE.0.0)) GO TO 500                          AT 0969
C                                                                      C
C19------IF HEAD > BOTTOM THEN ADD TERMS TO RHS AND HCOF.              C
      IF(IQFLG.GT.0) GO TO 400                                                   AT 0995
      RHS(IC,IR,IL)=RHS(IC,IR,IL)-CSTR*HSTR                                      AT 09A2
      HCOF(IC,IR,IL)=HCOF(IC,IR,IL)-CSTR                                         AT 09DC
      GO TO 500                                                                  AT 0A0E
C                                                                      C
C20------IF HEAD < BOTTOM THEN ADD TERM ONLY TO RHS.                   C
  400 RHS(IC,IR,IL)=RHS(IC,IR,IL)-FLOBOT                                         AT 0A13
  500 CONTINUE                                                                   AT 0A45
C                                                                      C
C21-----RETURN.                                                        C
      RETURN                                                                     AT 0A4A
      END                                                                        AT 0A4F
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>
c      SUBROUTINE STR1BD(NSTREM,STRM,ISTRM,IBOUND,MXSTRM,HNEW,NCOL,NROW,
c     1  NLAY,DELT,VBVL,VBNM,MSUM,KSTP,KPER,ISTCB1,ISTCB2,ICBCFL,BUFF,
c     2  IOUT,NTRIB,NSS,ARTRIB,ITRBAR,IDIVAR,NDFGAR,ICALC,CONST,IPTFLG)


      SUBROUTINE STR1BD(NSTREM,STRM,ISTRM,IBOUND,MXSTRM,HNEW,NCOL,NROW,
     1  NLAY,DELT,VBVL,VBNM,MSUM,KSTP,KPER,ISTCB1,ISTCB2,ICBCFL,BUFF,
     2  IOUT,NTRIB,NSS,ARTRIB,ITRBAR,IDIVAR,NDFGAR,ICALC,CONST,IPTFLG,
     3  iswabs,itrba2)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-----VERSION  2  18DEC1990 STR1BD                                     C
C                                                                      C
C     *****************************************************************C
C     CALCULATE VOLUMETRIC BUDGET FOR STREAMS                          C
C     *****************************************************************C
C                                                                      C
C     SPECIFICATIONS:                                                  C
```

```
C      ----------------------------------------------------------------C
       CHARACTER*4 VBNM,TEXT,STRTXT
       DOUBLE PRECISION HNEW
       DIMENSION STRM(11,MXSTRM),ISTRM(5,MXSTRM),IBOUND(NCOL,NROW,NLAY),
      1          HNEW(NCOL,NROW,NLAY),VBVL(4,20),VBNM(4,20),
      2          BUFF(NCOL,NROW,NLAY),ARTRIB(NSS),ITRBAR(NSS,NTRIB),
      3             IDIVAR(NSS),NDFGAR(NSS)
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
       dimension itrba2(nss,ntrib)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
       DIMENSION TEXT(4),STRTXT(4)
       DATA TEXT(1),TEXT(2),TEXT(3),TEXT(4) /'  ST','REAM',' LEA','KAGE'/          AT 00E0
       DATA STRTXT(1),STRTXT(2),STRTXT(3),STRTXT(4) /'STRE','AM F',                AT 00E0
      1                                         'LOW ','OUT '/
C      ----------------------------------------------------------------C
C                                                                      C
C1------SET IBD=1 IF BUDGET TERMS SHOULD BE SAVED ON DISK.             C
       IBD=0                                                                       AT 00E0
       RATIN = 0.                                                                  AT 00EA
       RATOUT = 0.                                                                 AT 00FE
C                                                                      C
C2------IF NO REACHES, KEEP ZEROS IN ACCUMULATORS.                     C
       IF(NSTREM.EQ.0) GO TO 600                                                   AT 0112
C                                                                      C
C3A-----TEST TO SEE IF CELL-BY-CELL TERMS ARE NEEDED.                  C
       IF((ICBCFL.EQ.0).OR.(ISTCB1.LE.0)) GO TO 10                                 AT 0121
C                                                                      C
C3B-----CELL-BY-CELL TERMS ARE NEEDED, SET IBD AND CLEAR BUFFER.       C
       IBD = 1                                                                     AT 0139
       DO 5 IL=1,NLAY                                                              AT 0143
       DO 5 IR=1,NROW                                                              AT 0170
       DO 5 IC=1,NCOL                                                              AT 019D
       BUFF(IC,IR,IL)=0.                                                           AT 01CA
     5 CONTINUE                                                                    AT 01FF
C                                                                      C
C3C-----INITIALIZE NDFGAR ARRAY TO ZERO.                               C
       DO 7 I=1,NSS                                                                AT 0208
       NDFGAR(I)=0                                                                 AT 0235
     7 CONTINUE                                                                    AT 0240
C                                                                      C
C4------IF THERE ARE STREAMS THEN ACCUMULATE LEAKAGE TO OR FROM THEM.  C
    10 DO 500 L=1,NSTREM                                                           AT 0242
       LL=L-1                                                                      AT 0272
C                                                                      C
C5---DETERMINE REACH LOCATION.                                         C
       IL=ISTRM(1,L)                                                               AT 0279
       IR=ISTRM(2,L)                                                               AT 0294
       IC=ISTRM(3,L)                                                               AT 02AB
C                                                                      C
C6----06FEB1990, CHECK FOR CELLS OUTSIDE MOVED TO C14, C18 AND C29.    C
C                                                                      C
C7------DETERMINE SEGMENT AND REACH NUMBER.                            C
       ISTSG=ISTRM(4,L)                                                            AT 02C2
       NREACH=ISTRM(5,L)                                                           AT 02D9
C-Environment Agency (UK)-13.Allow an inflow to any stream reach>>>>>>>>
c       IF(NREACH.GT.1) GO TO 200
cC                                                                      C
cC8------SET FLOWIN EQUAL TO SEGMENT INFLOW IF FIRST REACH.             C
c       FLOWIN=STRM(1,L)
C                                                                      C
C8------SET FLOWIN EQUAL TO SEGMENT INFLOW IF any REACH.               C
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
c       FLOWIN=STRM(1,L)
c       IF(NREACH.GT.1) GO TO 200
        IF(iswabs.eq.0.and.NREACH.GT.1) GO TO 200                                  AT 02F0
        FLOWIN=STRM(1,L)                                                           AT 0305
        IF(iswabs.lt.0.and.NREACH.GT.1) GO TO 200                                  AT 032E
        IF(NREACH.GT.1) GO TO 60                                                   AT 0347
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C                                                                      C
C9------STORE OUTFLOW FROM PREVIOUS SEGMENT IN ARTRIB IF SEGMENT >1.   C
        IF(ISTSG.EQ.1) GO TO 50                                                    AT 0354
        IFLG = ISTRM(4,LL)                                                         AT 0361
       ARTRIB(IFLG)=STRM(9,LL)                                                     AT 037E
C                                                                      C
C10A----CHECK UPSTREAM SEGMENT FOR DIVERSIONS.                         C
        DO 40 NSFLG = 1,NSS                                                        AT 03AE
       IF(IFLG.NE.IDIVAR(NSFLG)) GO TO 40                                          AT 03DB
C                                                                      C
C10B----DETERMINE AMOUNT OF FLOW TO BE DIVERTED.                       C
        DO 20 IDL=1,NSTREM                                                         AT 03EE
```

```
      IF(NSFLG.NE.ISTRM(4,IDL)) GO TO 20                                AT 041E
      IF(ISTRM(5,IDL).NE.1) GO TO 20                                    AT 0441
      DUM=ARTRIB(IFLG)-STRM(1,IDL)                                      AT 045F
C                                                                      C
C10C----SUBTRACT FLOW FROM UPSTREAM SEGMENT IF THERE IS ENOUGH FLOW    C
C        IN UPSTREAM SEGMENT.                                          C
      IF(DUM.GE.0.0) ARTRIB(IFLG)=DUM                                   AT 048C
      IF(DUM.LT.0.0) NDFGAR(IFLG)=1                                     AT 04B6
   20 CONTINUE                                                          AT 04DD
   40 CONTINUE                                                          AT 04E2
   50 IF(IDIVAR(ISTSG).LE.0) GO TO 60                                   AT 04E7
      NDFLG=IDIVAR(ISTSG)                                               AT 04FB
      IF(NDFGAR(NDFLG).EQ.1) FLOWIN=0.0                                 AT 0505
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
c   60 IF(FLOWIN.GE.0.0) GO TO 300
   60 IF(FLOWIN.GE.0.0.and.iswabs.eq.0) GO TO 300                       AT 0527
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C                                                                      C
C11--SUM TRIBUTARY OUTFLOW AND USE AS INFLOW INTO DOWNSTREAM SEGMENT.  C
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
c      FLOWIN =0.
      if(iswabs.eq.0)FLOWIN =0.                                         AT 0549
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      DO 100 ITRIB=1,NTRIB                                              AT 0569
      INODE=ITRBAR(ISTSG,ITRIB)                                         AT 0596
      IF(INODE.LE.0) GO TO 100                                          AT 05B8
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
      if(iswabs.gt.0)then                                               AT 05C1
        if(itrba2(istsg,itrib).ne.nreach)goto 100                       AT 05CD
      endif                                                             AT 05FB
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      FLOWIN=FLOWIN+ARTRIB(INODE)                                       AT 05FB
  100 CONTINUE                                                          AT 0615
C                                                                      C
C-Environment Agency (UK)-13.Allow an inflow to any stream reach>>>>>>>>
cC12-----IF REACH >1, SET INFLOW EQUAL TO OUTFLOW FROM UPSTREAM REACH.  C
c  200 IF(NREACH.GT.1) FLOWIN=STRM(9,LL)
C12-----IF REACH >1, add OUTFLOW FROM UPSTREAM REACH to inflow.         C
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
c  200 IF(NREACH.GT.1) FLOWIN=flowin+STRM(9,LL)
  200 if(iswabs.eq.0)then                                               AT 061A
        IF(NREACH.GT.1) FLOWIN=STRM(9,LL)                               AT 0626
      else                                                              AT 065E
        IF(NREACH.GT.1) FLOWIN=flowin+STRM(9,LL)                        AT 0663
      endif                                                             AT 0698
C                                                                      C
C12a---LIMIT SURFACE WATER ABSTRACTION IF NOT ENOUGH FLOW IN STREAM     C
      if(iswabs.ne.0.and.flowin.lt.0)then                               AT 0698
        write(iout,11)istsg,nreach,-strm(1,l),-strm(1,l)+flowin         AT 06BA
 11     format('**SURFACE WATER ABSTRACTION FROM SEGMENT',I5,', REACH',  AT 0784
     1        I5,' REDUCED FROM',G12.4,' TO',G12.4,' **')
        flowin=0.0                                                      AT 0809
      endif                                                             AT 081D
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C                                                                      C
C13----COMPUTE STREAM STAGE IN REACH IF ICALC > 1.                      C
  300 IF(ICALC.LE.0) GO TO 310                                          AT 081D
      XNUM=((FLOWIN+STRM(9,L))/2.0)*STRM(8,L)                           AT 0829
      DNOM=CONST*STRM(6,L)*(SQRT(STRM(7,L)))                            AT 086A
      DEPTH=(XNUM/DNOM)**0.6                                            AT 08A0
      IF((DEPTH).LE.0) DEPTH=0.                                         AT 08C5
      STRM(2,L)=DEPTH+STRM(5,L)                                         AT 08E9
  310 HSTR=STRM(2,L)                                                    AT 091E
C                                                                      C
C14----DETERMINE LEAKAGE THROUGH STREAMBED.                             C
      IF(IBOUND(IC,IR,IL).LE.0) GO TO 315                               AT 0949
      IF(FLOWIN.LE.0.0) HSTR=STRM(5,L)                                  AT 0986
      CSTR=STRM(3,L)                                                    AT 09C1
      SBOT=STRM(4,L)                                                    AT 09EC
      H=HNEW(IC,IR,IL)                                                  AT 0A11
      T=HSTR-SBOT                                                       AT 0A4E
C                                                                      C
C15----COMPUTE LEAKAGE AS A FUNCTION OF STREAM STAGE AND HEAD IN CELL. C
      FLOBOT=CSTR*(HSTR-H)                                              AT 0A61
C                                                                      C
C16----RECOMPUTE LEAKAGE IF HEAD IN CELL IS BELOW STREAMBED BOTTOM.     C
      IF(H.GT.SBOT) GO TO 312                                           AT 0A7A
      FLOBOT=CSTR*T                                                     AT 0A90
C                                                                      C
C17----SET LEAKAGE EQUAL TO STREAM INFLOW IF LEAKAGE MORE THAN INFLOW. C
  312 IF(FLOBOT.LE.FLOWIN) GO TO 320                                    AT 0AA3
```

```
      FLOBOT=FLOWIN                                                      AT 0AB9
C                                                                    C
C18----STREAMFLOW OUT EQUALS STREAMFLOW IN MINUS LEAKAGE.            C
  315 IF(IBOUND(IC,IR,IL).LE.0) FLOBOT=0.                                AT 0AD1
  320 FLOWOT=FLOWIN-FLOBOT                                               AT 0B22
      IF((ISTSG.GT.1).AND.(NREACH.EQ.1)) STRM(9,LL)=ARTRIB(IFLG)         AT 0B35
C                                                                    C
C19----STORE STREAM INFLOW, OUTFLOW AND LEAKAGE FOR EACH REACH.       C
      STRM(9,L)=FLOWOT                                                   AT 0B7F
      STRM(10,L)=FLOWIN                                                  AT 0BAA
      STRM(11,L)=FLOBOT                                                  AT 0BCF
C                                                                    C
C20----IF LEAKAGE FROM STREAMS IS TO BE SAVED THEN ADD RATE TO BUFFER.  C
      IF(IBD.EQ.1) BUFF(IC,IR,IL)=BUFF(IC,IR,IL)+FLOBOT                  AT 0BF4
C                                                                    C
C21----DETERMINE IF FLOW IS INTO OR OUT OF MODEL CELL.                C
C       SKIP ESTIMATE OF LEAKAGE FROM STREAM IF LEAKAGE IS ZERO.      C
      IF(FLOBOT)494,500,496                                              AT 0C39
C                                                                    C
C22-----SUBTRACT FLOW RATE FROM RATOUT IF AQUIFER DISCHARGES TO STREAM.C
  494 RATOUT=RATOUT-FLOBOT                                               AT 0C59
      GO TO 500                                                          AT 0C6C
C                                                                    C
C23-----ADD FLOW RATE TO RATIN IF STREAM DISCHARGES TO AQUIFER.       C
  496 RATIN=RATIN+FLOBOT                                                 AT 0C71
  500 CONTINUE                                                           AT 0C84
C                                                                    C
C24-----IF BUDGET TERMS WILL BE SAVED THEN WRITE TO DISK.             C
      IF(IBD.EQ.1) CALL UBUDSV(KSTP,KPER,TEXT,ISTCB1,BUFF,NCOL,NROW,     AT 0C89
     1                          NLAY,IOUT)
C                                                                    C
C25A-----MOVE RATES INTO VBVL FOR PRINTING BY MODULE BAS_OT.          C
  600 VBVL(3,MSUM)=RATIN                                                 AT 0CC1
      VBVL(4,MSUM)=RATOUT                                                AT 0CE3
C                                                                    C
C25B-----MOVE PRODUCT OF RATE AND TIME STEP INTO VBVL ACCUMULATORS.   C
      VBVL(1,MSUM)=VBVL(1,MSUM)+RATIN*DELT                               AT 0CFF
      VBVL(2,MSUM)=VBVL(2,MSUM)+RATOUT*DELT                              AT 0D19
C                                                                    C
C25C-----MOVE BUDGET TERM LABELS INTO VBNM FOR PRINTING BY BAS_OT.    C
      VBNM(1,MSUM)=TEXT(1)                                               AT 0D32
      VBNM(2,MSUM)=TEXT(2)                                               AT 0D48
      VBNM(3,MSUM)=TEXT(3)                                               AT 0D63
      VBNM(4,MSUM)=TEXT(4)                                               AT 0D78
C                                                                    C
C26-----INCREASE BUDGET TERM COUNTER BY ONE.                          C
      MSUM=MSUM+1                                                        AT 0D8D
C                                                                    C
C27-----RESET IBD COUNTER TO ZERO.                                    C
      IBD=0                                                              AT 0D8F
C28----IF STREAM OUTFLOW FROM EACH REACH IS TO BE STORED ON DISK      C
C     THEN STORE OUTFLOW RATES TO BUFFER.                             C
      IF((ICBCFL.EQ.0).OR.(ISTCB2.LE.0)) GO TO 625                       AT 0D99
      IBD = 1                                                            AT 0DB1
      DO 605 IL=1,NLAY                                                   AT 0DBB
      DO 605 IR=1,NROW                                                   AT 0DE8
      DO 605 IC=1,NCOL                                                   AT 0E15
  605 BUFF(IC,IR,IL)=0.                                                  AT 0E42
C                                                                    C
C29-----SAVE STREAMFLOWS OUT OF EACH REACH ON DISK.                    C
      DO 615 L=1,NSTREM                                                  AT 0E87
      IC=ISTRM(3,L)                                                      AT 0EB7
      IR=ISTRM(2,L)                                                      AT 0ECE
      IL=ISTRM(1,L)                                                      AT 0EE5
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
c      IF(IBOUND(IC,IR,IL).LE.0) GO TO 615
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-Environment Agency (UK)-15.Save single flow if >1 stream in a cell>>>>
c      BUFF(IC,IR,IL)=BUFF(IC,IR,IL)+STRM(9,L)
      BUFF(IC,IR,IL)=STRM(9,L)                                           AT 0EFA
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
  615 CONTINUE                                                           AT 0F3E
      CALL UBUDSV(KSTP,KPER,STRTXT,ISTCB2,BUFF,NCOL,NROW,NLAY,IOUT)      AT 0F43
C                                                                    C
C30-----PRINT STREAMFLOW RATES AND LEAKAGE FOR EACH REACH.            C
  625 IF((ISTCB1.GE.0).OR.(ICBCFL.LE.0)) GO TO 800                       AT 0F6E
      IF(IPTFLG.GT.0) GO TO 800                                          AT 0F86
      IF(ICALC.GT.0) GO TO 700                                           AT 0F92
      WRITE(IOUT,650)                                                    AT 0F9E
  650 FORMAT(1H0,12X,'LAYER',6X,'ROW',5X,'COLUMN',5X,'STREAM',4X,        AT 0FC4
     1'REACH',6X,'FLOW INTO',4X,'FLOW INTO',6X,'FLOW OUT OF'/43X,
     2      'NUMBER',3X,'NUMBER',4X,'STREAM REACH',4X,'AQUIFER',
```

```
     3      6X,'STREAM REACH'//)
      DO 690 L=1,NSTREM                                                    AT 10B1
      IL=ISTRM(1,L)                                                        AT 10E1
      IR=ISTRM(2,L)                                                        AT 10F6
      IC=ISTRM(3,L)                                                        AT 110D
      WRITE(IOUT,675)IL,IR,IC,ISTRM(4,L),ISTRM(5,L),                       AT 1124
     1      STRM(10,L),STRM(11,L),STRM(9,L)
  675 FORMAT(1X,5X,5I10,8X,G9.3,5X,G9.3,8X,G9.3)                           AT 1269
  690 CONTINUE                                                             AT 12AA
      GO TO 800                                                            AT 12AF
  700 WRITE(IOUT,710)                                                      AT 12B4
  710 FORMAT(1H0,12X,'LAYER',6X,'ROW',5X,'COLUMN',5X,'STREAM',4X,          AT 12DA
     1'REACH',6X,'FLOW INTO',4X,'FLOW INTO',6X,'FLOW OUT OF',5X,
     2'HEAD IN'/43X,      'NUMBER',3X,'NUMBER',4X,'STREAM REACH',
     3 4X,'AQUIFER',6X,'STREAM REACH',5X,'STREAM'//)
      DO 750 L=1,NSTREM                                                    AT 13E5
      IL=ISTRM(1,L)                                                        AT 1415
      IR=ISTRM(2,L)                                                        AT 142A
      IC=ISTRM(3,L)                                                        AT 1441
      WRITE(IOUT,775)IL,IR,IC,ISTRM(4,L),ISTRM(5,L),                       AT 1458
     1      STRM(10,L),STRM(11,L),STRM(9,L),STRM(2,L)
  775 FORMAT(1X,5X,5I10,8X,G9.3,5X,G9.3,7X,G9.3,4X,F9.2)                   AT 15CC
  750 CONTINUE                                                             AT 1619
  800 CONTINUE                                                             AT 161E
C                                                                  C
C31-----RETURN.                                                     C
      RETURN                                                               AT 161E
      END                                                                  AT 1623
```

End of compilation - Clocked 0.0 seconds

APPENDIX C
Modified computer code for PCG solver package with changes highlighted

Salford Software Ltd FTN77 Ver. 3.53   C:\Company\Tech\MF-VKD\Source_code\pcg-vkd1.for   14:34:00
Monday, 8 September 2003

COMPILER OPTIONS: listing intl no_persist nodclvar nomap nocheck logl dynm offset dreal noansi
nopagethrow
                   nosilent no_optimise warn73 nolink no_link

```
C-Changes to the code are highlighted with the following headings:
C
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
C-Environment Agency (UK)-10.Convergence info output to screen>>>>>>>>>>
C-Environment Agency (UK)-11.Force converge if NOUTC outer itrs conv>>>>
C-Environment Agency (UK)-12.Debugging output of unconverged heads>>>>>>
C
C These headings are followed by the original USGS MODFLOW-96 code
C (commented out), followed by the modified code. All the code changes
C are followed by the following:
C
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-Environment Agency (UK)-12.Debugging output of unconverged heads>>>>>>
C-            NOTE: The debugging option produces a file of all the
C-            heads calculated in the iteration process. This file is
C-            reset each time that a time step converges, so that only
C-            the information from an unconverged time step is kept. The
c-            outer iteration number is written to the part of the file
C-            that usually contains the stress period number, the inner
C-            iteration no. to the time step location, the largest head
C-            change to the stress period time location, and the largest
C-            flow residual to the total time location. The output file
C-            needs to be specified in the name file as DATA(BINARY), and
C-            the unit no. should be in columns 41-50 in the 1st line of
C-            the PCG input file.
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C        $Date: 1996/03/29 18:30:00 $
C        $Revision: 2.3 $
C
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
c       SUBROUTINE PCG2AL(ISUM,LENX,LCV,LCSS,LCP,LCCD,LCHCHG,LCLHCH,
c     1       LCRCHG,LCLRCH,MXITER,ITER1,NCOL,NROW,NLAY,IN,IOUT,NPCOND,
c     A       LCIT1)
      SUBROUTINE PCG2AL(ISUM,LENX,LCV,LCSS,LCP,LCCD,LCHCHG,LCLHCH,
     1       LCRCHG,LCLRCH,MXITER,ITER1,NCOL,NROW,NLAY,IN,IOUT,NPCOND,
     A       LCIT1,noutc,ipcgdebug,iapart)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C-----VERSION 0002 01MAY1989 PCG2AL
C
C     ******************************************************************
C     ALLOCATE STORAGE IN THE X ARRAY FOR PCG ARRAYS
C     ******************************************************************
C
C        SPECIFICATIONS:
C     ------------------------------------------------------------------
C     ------------------------------------------------------------------
C
C-------PRINT A MESSAGE IDENTIFYING PCG PACKAGE
      WRITE(IOUT,1)                                                       AT 0018
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
c     1 FORMAT(1H0,'PCG2 -- CONJUGATE GRADIENT SOLUTION PACKAGE'
c     1,', VERSION 2.1, 6/1/95')
    1 FORMAT(1H0,'PCG-VKD1 -- CONJUGATE GRADIENT SOLUTION PACKAGE'        AT 003E
     1,', Including progress monitor & debugging option, 18/7/2001')
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
C-------READ AND PRINT MXITER,ITER1 AND NPCOND
c       READ(IN,2) MXITER,ITER1,NPCOND
c     2 FORMAT(3I10)
c       WRITE(IOUT,3) MXITER,ITER1,NPCOND
c     3 FORMAT(' MAXIMUM OF',I4,' CALLS OF SOLUTION ROUTINE'/
c     1       ,' MAXIMUM OF',I4,' INTERNAL ITERATIONS PER '
c     2       ,'CALL TO SOLUTION ROUTINE'/
c     3       ,' MATRIX PRECONDITIONING TYPE :',I5)
      READ(IN,2) MXITER,ITER1,NPCOND,noutc,ipcgdebug                      AT 00C5
    2 FORMAT(5I10)                                                        AT 013B
      WRITE(IOUT,3) MXITER,ITER1,NPCOND,noutc                             AT 0150
    3 FORMAT(' MAXIMUM OF',I4,' CALLS OF SOLUTION ROUTINE'/               AT 01B6
     1       ,' MAXIMUM OF',I4,' INTERNAL ITERATIONS PER '
     2       ,'CALL TO SOLUTION ROUTINE'/
     3       ,' MATRIX PRECONDITIONING TYPE :',I5/
     4       ,' CONVERGENCE WILL BE FORCED IF',I4,' CONSECUTIVE '
```

```
     5       ,'OUTER ITERATIONS ARE WITHIN THE CONVERGENCE CRITERIA')
      if(ipcgdebug.gt.0)then                                           AT 02EF
        if(iapart.ne.0)then                                            AT 02FB
          write(iout,*)                                                AT 0307
          write(iout,*)'*** DEBUGGING OPTION ACTIVE ***'               AT 0321
          write(iout,*)                                                AT 0353
        else                                                           AT 036D
          write(iout,*)                                                AT 0372
          write(iout,*)'*********************************************' AT 038C
          write(iout,*)'** DEBUGGING IS NOT ACTIVE BECAUSE IAPART=0 **' AT 03BE
          write(iout,*)'*********************************************' AT 03F0
          write(iout,*)                                                AT 0422
          ipcgdebug=0                                                  AT 043C
        endif                                                          AT 0445
      endif                                                            AT 0445
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C-------ALLOCATE SPACE FOR THE PCG ARRAYS
      ISOLD=ISUM                                                       AT 0445
      NRC=NROW*NCOL                                                    AT 044D
      ISIZ=NRC*NLAY                                                    AT 045B
      LCV=ISUM                                                         AT 0464
      ISUM=ISUM+ISIZ                                                   AT 046B
      LCSS=ISUM                                                        AT 0470
      ISUM=ISUM+ISIZ                                                   AT 047A
      LCP=ISUM                                                         AT 047F
      ISUM=ISUM+ISIZ                                                   AT 0489
      LCCD=ISUM                                                        AT 048E
      IF(NPCOND.NE.2) ISUM=ISUM+ISIZ                                   AT 0498
      LCHCHG=ISUM                                                      AT 04A9
      ISUM=ISUM+MXITER*ITER1                                           AT 04B3
      LCLHCH=ISUM                                                      AT 04C0
      ISUM=ISUM+3*MXITER*ITER1                                         AT 04C7
      LCRCHG=ISUM                                                      AT 04D8
      ISUM=ISUM+MXITER*ITER1                                           AT 04DF
      LCLRCH=ISUM                                                      AT 04EC
      ISUM=ISUM+3*MXITER*ITER1                                         AT 04F3
      LCIT1=ISUM                                                       AT 0504
      ISUM=ISUM+MXITER*ITER1                                           AT 050B
C
C-------CALCULATE AND PRINT THE SPACE USED IN THE X ARRAY
      ICG=ISUM-ISOLD                                                   AT 0518
      WRITE(IOUT,4) ICG                                                AT 0520
    4 FORMAT(1X,I7,' ELEMENTS IN X ARRAY ARE USED BY PCG')             AT 0557
      ISUM1=ISUM-1                                                     AT 0598
      WRITE(IOUT,5) ISUM1,LENX                                         AT 05A1
    5 FORMAT(1X,I7,' ELEMENTS OF X ARRAY USED OUT OF',I7)              AT 05E8
      IF(ISUM1.GT.LENX) WRITE(IOUT,6)                                  AT 062B
    6 FORMAT(1X,'   ***X ARRAY MUST BE DIMENSIONED LARGER***')         AT 065F
C
C-------RETURN
      RETURN                                                           AT 06A2
      END                                                              AT 06A7



      SUBROUTINE PCG2RP(MXITER,ITER1,HCLOSE,RCLOSE,NPCOND,NBPOL,
     1                RELAX,IPRPCG,IN,IOUT,MUTPCG,NITER,IT1,DAMP)
C
C-----VERSION 0002 01MAY1989 PCG2RP
C      01SEPT1990 IPCGCD OMITTED; NITER ADDED
C      01JUNE1995 DAMP ADDED
C
C      ******************************************************************
C      READ DATA FOR PCG
C      ******************************************************************
C
C         SPECIFICATIONS:
C      ------------------------------------------------------------------
      DIMENSION IT1(MXITER*ITER1)
C      ------------------------------------------------------------------
C
C-------READ HCLOSE,RCLOSE,RELAX,NBPOL,IPRPCG,MUTPCG
      READ(IN,1) HCLOSE,RCLOSE,RELAX,NBPOL,IPRPCG,MUTPCG,DAMP           AT 002C
    1 FORMAT(3F10.0,3I10,F10.0)                                        AT 00C2
C
C-------PRINT MXITER,ITER1,NPCOND,HCLOSE,RCLOSE,RELAX,NBPOL,IPRPCG,
C-------MUTPCG,DAMP
      WRITE(IOUT,100)                                                  AT 00E7
  100 FORMAT(1H0,///57X,'SOLUTION BY THE CONJUGATE-GRADIENT METHOD'    AT 010D
     1/57X,43('-'))
      WRITE(IOUT,115) MXITER                                           AT 0170
```

```
  115 FORMAT(1H0,38X,'MAXIMUM NUMBER OF CALLS TO PCG ROUTINE =',I9)            AT 01A6
      WRITE(IOUT,120) ITER1                                                    AT 01F1
  120 FORMAT(1H ,42X,'MAXIMUM ITERATIONS PER CALL TO PCG =',I9)                AT 0227
      WRITE(IOUT,122) NPCOND                                                   AT 026E
  122 FORMAT(1H ,49X,'MATRIX PRECONDITIONING TYPE =',I9)                       AT 02A4
      IF(NPCOND.EQ.2) WRITE(IOUT,123)                                          AT 02E5
  123 FORMAT(1H ,58X,'THE MATRIX WILL BE SCALED')                              AT 0317
      WRITE(IOUT,124) RELAX,NBPOL                                              AT 034E
  124 FORMAT(1H ,26X,'RELAXATION FACTOR (ONLY USED WITH',                      AT 0394
     1' PRECOND. TYPE 1) =',E15.5,/,
     2 1H ,19X,'PARAMETER OF POLYMOMIAL PRECOND.'
     3 ,' = 2 (2) OR IS CALCULATED :',I9)
      WRITE(IOUT,125) HCLOSE                                                   AT 044B
  125 FORMAT(1H ,43X,'HEAD CHANGE CRITERION FOR CLOSURE =',E15.5)              AT 0481
      WRITE(IOUT,127) RCLOSE                                                   AT 04CA
  127 FORMAT(1H ,39X,'RESIDUAL CHANGE CRITERION FOR CLOSURE =',E15.5)          AT 0500
      IF(IPRPCG.LE.0)IPRPCG=999                                                AT 054D
      WRITE(IOUT,130) IPRPCG,MUTPCG                                            AT 055F
  130 FORMAT(1H ,30X,'PCG HEAD AND RESIDUAL CHANGE PRINTOUT INTERVAL ='        AT 05A5
     1,I9,/,1H ,23X,'PRINTING FROM SOLVER IS LIMITED(1) OR ',
     2'SUPPRESSED (>1) =',I9)
      IF(DAMP.LE.0.0) DAMP=1.0                                                 AT 064C
      WRITE(IOUT,135) DAMP                                                     AT 0671
 135  FORMAT(1H ,59X,'DAMPING PARAMETER =',E15.5)                              AT 06A7
      NITER=0                                                                  AT 06E0
C
      RETURN                                                                   AT 06E9
      END                                                                      AT 06EE
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
C      SUBROUTINE PCG2AP(HNEW,IBOUND,CR,CC,CV,HCOF,RHS,V,SS,P,CD,
C     1      HCHG,LHCH,RCHG,LRCH,KITER,NITER,HCLOSE,RCLOSE,ICNVG,
C     2      KSTP,KPER,IPRPCG,MXITER,ITER1,NPCOND,NBPOL,NSTP,NCOL,NROW,
C     3      NLAY,NODES,RELAX,IOUT,MUTPCG,IU,IP,SN,SP,SR,IT1,DAMP)
```

```
      SUBROUTINE PCG2AP(HNEW,IBOUND,CR,CC,CV,HCOF,RHS,V,SS,P,CD,
     1      HCHG,LHCH,RCHG,LRCH,KITER,NITER,HCLOSE,RCLOSE,ICNVG,
     2      KSTP,KPER,IPRPCG,MXITER,ITER1,NPCOND,NBPOL,NSTP,NCOL,NROW,
     3      NLAY,NODES,RELAX,IOUT,MUTPCG,IU,IP,SN,SP,SR,IT1,DAMP,noutc,
     4      nout,ipcgdebug,ixsec,chedfm,buff,lbhdsv)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-----VERSION 0002 01MAY1989 PCG2AP
C     01JULY1990 COMMENT STATEMENTS ADDED AND MODIFIED
C     01SEPT1990 IPCGCD OMITTED; STATEMENT 590 ADDED
C     27SEPT1990 STATEMENT IN DO 155 LOOP CHANGED
C     01SEPT1991 ADDED STATEMENTS RELATED TO SENSITIVITY CALCULATIONS
C                AT THE END OF THE 115 LOOP.  CHANGED THE 510 FORMAT
C                STATEMENT AND THE PRECEDING IF STATEMENT
C     20MAR1992  CHANGED 510 FORMAT STATEMENT; OMITTED 2 LINES IN DO 160
C                LOOP
C     01MAY1993  ADDED DEL TO CALCULATION OF THE CHOLESKY DIAGONAL
C     15JUNE1993 MADE CELLS SURROUNDED BY DRY CELLS INACTIVE
C     01JUNE1995 ADDED DAMP
C
C     ****************************************************************
C     SOLUTION BY THE CONJUGATE GRADIENT METHOD -
C                                          UP TO ITER1 ITERATIONS
C     ****************************************************************
C
C        SPECIFICATIONS:
C     -----------------------------------------------------------------
      PARAMETER (DZERO=0.D0,DONE=1.D0)
      DOUBLE PRECISION HNEW,HHCOF,RRHS,DEL
      DOUBLE PRECISION Z,B,D,E,F,H,S,ALPHA
      DOUBLE PRECISION ZHNEW,BHNEW,DHNEW,FHNEW,HHNEW,SHNEW
      DOUBLE PRECISION SRNEW,SROLD,SSCR,SSCC,SSCV,VCC,VCR,VCV
      DOUBLE PRECISION CDCC,CDCR,CDCV
      DOUBLE PRECISION PN,VN,HCHGN,RCHGN,PAP
      DOUBLE PRECISION FCC,FCR,FCV,FV
C
      DIMENSION HNEW(NODES), IBOUND(NODES), CR(NODES), CC(NODES),
     1  CV(NODES), HCOF(NODES), RHS(NODES),IT1(MXITER*ITER1),
     2  V(NODES), SS(NODES), P(NODES), CD(NODES), HCHG(MXITER*ITER1),
     3  LHCH(3,MXITER*ITER1), RCHG(MXITER*ITER1), LRCH(3,MXITER*ITER1)
C-Environment Agency (UK)-12.Debugging output of unconverged heads>>>>>>
      dimension buff(nodes)
      character text*16,chedfm*20,fname*100,acc*10
      text='Unconverged Head'                                                  AT 0059
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C     -----------------------------------------------------------------
C
```

```
C      IF(NITER.EQ.0) THEN
C        WRITE(IOUT,895)
C        WRITE(IOUT,900) (I,CC(I),CR(I),CV(I),HCOF(I),
C     1     RHS(I),HNEW(I),IBOUND(I),I=1,50)
C      ENDIF
  895 FORMAT('    I',5X,'CC',10X,'CR',10X,'CV',9X,'HCOF',8X,'RHS',9X,           AT 006C
     1   'HNEW',4X,'IBOUND (IBND=0 NOT PRINTED)')
  900 FORMAT(I5,5E12.3,D12.3,I5)                                                AT 00EB
C-Environment Agency (UK)-12.Debugging output of unconverged heads>>>>>>
      if(ipcgdebug.gt.0.and.kiter.eq.1)then                                     AT 0116
        inquire(ipcgdebug,access=acc,name=fname)                                AT 0131
        if(acc.eq.'DIRECT')then                                                 AT 016A
          inquire(ipcgdebug,recl=irecl)                                         AT 0183
          close(ipcgdebug,status='delete')                                      AT 01A4
          irecl=((ncol*nrow)+11)*4                                              AT 01C9
          open(ipcgdebug,file=fname,access='direct',recl=irecl)                 AT 01E0
        else                                                                    AT 0223
          close(ipcgdebug,status='delete')                                      AT 0228
          open(ipcgdebug,file=fname,access='sequential')                        AT 024D
        endif                                                                   AT 0284
      endif                                                                     AT 0284
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-------ASSIGN VARIABLE EQUAL TO THE NUMBER OF CELLS IN ONE LAYER
      NRC=NROW*NCOL                                                             AT 0284
C-------INITIALIZE VARIABLES USED TO CALCULATE ITERATION PARAMETERS
      SRNEW=DZERO                                                               AT 0295
      BPOLY=0.                                                                  AT 02A9
      IF(NPCOND.NE.1) RELAX=1.                                                  AT 02BD
      NORM=0                                                                    AT 02D9
      IF(NPCOND.EQ.2) NORM=1                                                    AT 02E3
C-------INITIALIZE VARIABLE USED TO TEST FOR NEGATIVE CHOLESKY DIAGONAL
      CD1=0.                                                                    AT 02F9
C------CLEAR PCG WORK ARRAYS.
      DO 100 N=1,NODES                                                          AT 030D
      SS(N)=0.                                                                  AT 033A
      P(N)=0.                                                                   AT 0350
  100 V(N)=0.                                                                   AT 0366
C------FOR NPCOND=1, INITIALIZE CHOLESKY DIAGONAL
      IF(NPCOND.EQ.1) THEN                                                      AT 0384
        DO 105 N=1,NODES                                                        AT 0390
  105   CD(N)=0.                                                                AT 03BD
      ENDIF                                                                     AT 03DB
C
C------CALCULATE THE RESIDUAL. IF NORM=1, CALCULATE THE DIAGONALS OF
C------THE A MATRIX,AND STORE THEM IN HCOF.
      DO 115 K=1,NLAY                                                           AT 03DB
      DO 115 I=1,NROW                                                           AT 0408
      DO 115 J=1,NCOL                                                           AT 0435
C
C-------CALCULATE 1 DIMENSIONAL SUBSCRIPT OF CURRENT CELL AND
C-------SKIP CALCULATIONS IF CELL IS INACTIVE
      N=J+(I-1)*NCOL+(K-1)*NRC                                                  AT 0462
      IF(IBOUND(N).EQ.0) THEN                                                   AT 0487
        CC(N)=0.                                                                AT 0498
        CR(N)=0.                                                                AT 04AE
        IF(N.LE.(NODES-NRC)) CV(N)=0.                                           AT 04C4
        IF(N.GE.2) CR(N-1)=0.                                                   AT 04ED
        IF(N.GE.NCOL+1) CC(N-NCOL)=0.                                           AT 0516
        IF(N.LE.(NODES-NRC).AND.N.GE.NRC+1) CV(N-NRC)=0.                        AT 0546
        HCOF(N)=0.                                                              AT 0592
        RHS(N)=0.                                                               AT 05AE
        GO TO 115                                                               AT 05C4
      ENDIF                                                                     AT 05C9
C
C-------CALCULATE 1 DIMENSIONAL SUBSCRIPTS FOR LOCATING THE 6
C-------SURROUNDING CELLS
      NRN=N+NCOL                                                                AT 05C9
      NRL=N-NCOL                                                                AT 05DA
      NCN=N+1                                                                   AT 05E8
      NCL=N-1                                                                   AT 05F5
      NLN=N+NRC                                                                 AT 0602
      NLL=N-NRC                                                                 AT 0614
C
C-------CALCULATE 1 DIMENSIONAL SUBSCRIPTS FOR CONDUCTANCE TO THE 6
C-------SURROUNDING CELLS.
      NCF=N                                                                     AT 0626
      NCD=N-1                                                                   AT 0632
      NRB=N-NCOL                                                                AT 0639
      NRH=N                                                                     AT 0647
      NLS=N                                                                     AT 0653
      NLZ=N-NRC                                                                 AT 0659
C
```

```
C-----GET CONDUCTANCES TO NEIGHBORING CELLS
C-------NEIGHBOR IS 1 ROW BACK
      B=DZERO                                                  AT 0665
      BHNEW=DZERO                                              AT 0679
      IF(I.NE.1) THEN                                          AT 068D
        B=CC(NRB)                                              AT 069A
        BHNEW=B*(HNEW(NRL)-HNEW(N))                            AT 06BA
      ENDIF                                                    AT 06E1
C
C-------NEIGHBOR IS 1 ROW AHEAD
      H=DZERO                                                  AT 06E1
      HHNEW=DZERO                                              AT 06F5
      IF(I.NE.NROW) THEN                                       AT 0709
        H=CC(NRH)                                              AT 071A
        HHNEW=H*(HNEW(NRN)-HNEW(N))                            AT 073A
      ENDIF                                                    AT 0761
C
C-------NEIGHBOR IS 1 COLUMN BACK
      D=DZERO                                                  AT 0761
      DHNEW=DZERO                                              AT 0775
      IF(J.NE.1) THEN                                          AT 0789
        D=CR(NCD)                                              AT 0796
        DHNEW=D*(HNEW(NCL)-HNEW(N))                            AT 07B6
      ENDIF                                                    AT 07DD
C
C-------NEIGHBOR IS 1 COLUMN AHEAD
      F=DZERO                                                  AT 07DD
      FHNEW=DZERO                                              AT 07F1
      IF(J.NE.NCOL) THEN                                       AT 0805
        F=CR(NCF)                                              AT 0816
        FHNEW=F*(HNEW(NCN)-HNEW(N))                            AT 0836
      ENDIF                                                    AT 085D
C
C-------NEIGHBOR IS 1 LAYER BEHIND
      Z=DZERO                                                  AT 085D
      ZHNEW=DZERO                                              AT 0871
      IF(K.NE.1) THEN                                          AT 0885
        Z=CV(NLZ)                                              AT 0892
        ZHNEW=Z*(HNEW(NLL)-HNEW(N))                            AT 08B2
      ENDIF                                                    AT 08D9
C
C-------NEIGHBOR IS 1 LAYER AHEAD
      S=DZERO                                                  AT 08D9
      SHNEW=DZERO                                              AT 08ED
      IF(K.NE.NLAY) THEN                                       AT 0901
        S=CV(NLS)                                              AT 0912
        SHNEW=S*(HNEW(NLN)-HNEW(N))                            AT 0932
      ENDIF                                                    AT 0959
C
      IF(I.EQ.NROW) CC(N)=0.                                   AT 0959
      IF(J.EQ.NCOL) CR(N)=0.                                   AT 0986
C-------15JUN1993 SKIP CALCULATIONS AND MAKE CELL INACTIVE IF ALL
C                 SURROUNDING CELLS ARE INACTIVE
      IF(B+H+D+F+Z+S.EQ.0.) THEN                               AT 09B3
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
        IBOUND(N)=0                                            AT 09E7
        HCOF(N)=0.                                             AT 09FB
        RHS(N)=0.                                              AT 0A11
        GO TO 115                                              AT 0A27
      ENDIF                                                    AT 0A2C
C
C-------CALCULATE THE RESIDUAL AND STORE IT IN RHS.  TO SCALE A,
C-------CALCULATE THE DIAGONAL OF THE A MATRIX, AND STORE IT IN HCOF.
      E=-Z-B-D-F-H-S                                           AT 0A2C
      RRHS=RHS(N)                                              AT 0A59
      HHCOF=HNEW(N)*HCOF(N)                                    AT 0A79
      RHS(N)=RRHS-ZHNEW-BHNEW-DHNEW-HHCOF-FHNEW-HHNEW-SHNEW     AT 0A94
      IF(NORM.EQ.1) HCOF(N)=HCOF(N)+E                          AT 0AC5
      IF(IBOUND(N).LT.0) RHS(N)=0.                             AT 0AE1
C-------ADDED FOR SENSITIVITY CALCULATIONS 9/1/91
      IF(IU.NE.0.AND.IP.GT.0) THEN                             AT 0B0E
        IF(I.EQ.1.AND.J.EQ.1.AND.K.EQ.1) THEN                  AT 0B26
          SN=0.                                                AT 0B4D
          SP=0.                                                AT 0B5D
          SR=0.                                                AT 0B6D
        ENDIF                                                  AT 0B7D
        SR=SR+RHS(N)                                           AT 0B7D
        IF(RRHS.LT.0.) SN=SN+RRHS                              AT 0B95
        IF(RRHS.GT.0.) SP=SP+RRHS                              AT 0BB9
        IF(-ZHNEW.LT.0.) SN=SN-ZHNEW                           AT 0BDD
        IF(-ZHNEW.GT.0.) SP=SP-ZHNEW                           AT 0C03
        IF(-BHNEW.LT.0.) SN=SN-BHNEW                           AT 0C29
```

```
          IF(-BHNEW.GT.0.) SP=SP-BHNEW                                      AT 0C4F
          IF(-DHNEW.LT.0.) SN=SN-DHNEW                                      AT 0C75
          IF(-DHNEW.GT.0.) SP=SP-DHNEW                                      AT 0C9B
          IF(-HHCOF.LT.0.) SN=SN-HHCOF                                      AT 0CC1
          IF(-HHCOF.GT.0.) SP=SP-HHCOF                                      AT 0CE7
          IF(-FHNEW.LT.0.) SN=SN-FHNEW                                      AT 0D0D
          IF(-FHNEW.GT.0.) SP=SP-FHNEW                                      AT 0D33
          IF(-HHNEW.LT.0.) SN=SN-HHNEW                                      AT 0D59
          IF(-HHNEW.GT.0.) SP=SP-HHNEW                                      AT 0D7F
          IF(-SHNEW.LT.0.) SN=SN-SHNEW                                      AT 0DA5
          IF(-SHNEW.GT.0.) SP=SP-SHNEW                                      AT 0DCB
        ENDIF                                                               AT 0DF1
    115 CONTINUE                                                            AT 0DF1
C
C-------SCALE CC,CR,CV,RHS AND HNEW IF NORM=1.
        IF(NORM.EQ.1) THEN                                                  AT 0E00
          DO 120 K=1,NLAY                                                   AT 0E0D
          DO 120 I=1,NROW                                                   AT 0E3A
          DO 120 J=1,NCOL                                                   AT 0E67
            N=J+(I-1)*NCOL+(K-1)*NRC                                        AT 0E94
            IF(IBOUND(N).EQ.0) GO TO 120                                    AT 0EB9
            HHCOF=SQRT(-HCOF(N))                                            AT 0ECA
            IF(N.LE.(NODES-NCOL).AND.CC(N).GT.0.)                           AT 0EDF
     #        CC(N)=CC(N)/(HHCOF*(SQRT(-HCOF(N+NCOL))))
            IF(CR(N).GT.0.) CR(N)=CR(N)/(HHCOF*(SQRT(-HCOF(N+1))))          AT 0F2F
            IF(N.LE.(NODES-NRC).AND.CV(N).GT.0.)                            AT 0F74
     #        CV(N)=CV(N)/(HHCOF*(SQRT(-HCOF(N+NRC))))
            HNEW(N)=HNEW(N)*HHCOF                                           AT 0FDC
            RHS(N)=RHS(N)/HHCOF                                             AT 0FF7
    120   CONTINUE                                                          AT 100C
        ENDIF                                                               AT 101B
C
C-------CALCULATE PARAMETER B OF THE POLYNOMIAL PRECONDITIONING METHOD
        IF(NPCOND.NE.2) GO TO 152                                           AT 101B
        IF(NBPOL.EQ.2) THEN                                                 AT 1027
          BPOLY=2                                                           AT 1033
          GO TO 151                                                         AT 1047
        ENDIF                                                               AT 104C
        DO 150 K=1,NLAY                                                     AT 104C
        DO 150 I=1,NROW                                                     AT 1079
        DO 150 J=1,NCOL                                                     AT 10A6
C
        N=J+(I-1)*NCOL+(K-1)*NRC                                            AT 10D3
        IF(IBOUND(N).LE.0)GO TO 150                                         AT 10F8
C
        NCF=N                                                               AT 1109
        NCD=N-1                                                             AT 110F
        NRB=N-NCOL                                                          AT 1116
        NRH=N                                                               AT 1124
        NLS=N                                                               AT 1130
        NLZ=N-NRC                                                           AT 1136
C
        B=DZERO                                                             AT 1142
        IF(I.NE.1) B=CC(NRB)                                                AT 1156
        H=DZERO                                                             AT 117D
        IF(I.NE.NROW) H=CC(NRH)                                             AT 1191
        D=DZERO                                                             AT 11C2
        IF(J.NE.1) D=CR(NCD)                                                AT 11D6
        F=DZERO                                                             AT 1203
        IF(J.NE.NCOL) F=CR(NCF)                                             AT 1217
        Z=DZERO                                                             AT 1248
        IF(K.NE.1) Z=CV(NLZ)                                                AT 125C
        S=DZERO                                                             AT 1289
        IF(K.NE.NLAY) S=CV(NLS)                                             AT 129D
C
C-------NOTE : ABS. VAL. OF THE DIAG. OF THE SCALED A MATRIX IS 1.
        HHCOF=HCOF(N)                                                       AT 12CE
        IF(NORM.EQ.1) HHCOF=DONE                                            AT 12EE
        T=DABS(Z)+DABS(B)+DABS(D)+ABS(HHCOF)+DABS(F)+DABS(H)+DABS(S)        AT 130F
        IF(T.GT.BPOLY) BPOLY=T                                              AT 135A
    150 CONTINUE                                                            AT 1382
    151 CONTINUE                                                            AT 1391
C
C-------CALCULATE ITERATION PARAMETERS FOR POLYNOMIAL PRECONDITIONING
C-------METHOD FOR A NEGATIVE DEFINITE MATRIX.
          C0=(15./32.)*(BPOLY**3)                                           AT 1391
          C1=(27./16.)*(BPOLY**2)                                           AT 13B0
          C2=(9./4.)*BPOLY                                                  AT 13C9
    152 CONTINUE                                                            AT 13DC
C
C-------START INTERNAL ITERATIONS
        IITER=0                                                             AT 13DC
```

```
       IF(KITER.EQ.1) NITER=0                                                AT 13E6
       ICNVG=0                                                               AT 1401
       IICNVG=0                                                              AT 140D
 153   CONTINUE                                                              AT 1417
       IITER=IITER+1                                                         AT 1417
       NITER=NITER+1                                                         AT 141D
 C
 C-------INITIALIZE VARIABLES THAT TRACK MAXIMUM HEAD CHANGE AND RESIDUAL
 C-------VALUE DURING EACH ITERATIONS
       BIGH=0.                                                               AT 1425
       BIGR=0.                                                               AT 1439
 C-------INITIALIZE DEL (ADDED 01MAY1993)
       DEL=0.                                                                AT 144D
 C
 C
 C-------CHECK NPCOND FOR PRECONDITIONING TYPE AND EXECUTE PROPER CODE
       IF(NPCOND.EQ.2) GO TO 165                                             AT 1461
 C
 C-------CHOLESKY PRECONDITIONING
 C
 C-------STEP THROUGH CELLS TO CALCULATE THE DIAGONAL OF THE CHOLESKY
 C-------MATRIX (FIRST INTERNAL ITERATION ONLY) AND THE INTERMEDIATE
 C-------SOLUTION.  STORE THEM IN CD AND V, RESPECTIVELY.
 154   DO 155 K=1,NLAY                                                       AT 146D
       DO 155 I=1,NROW                                                       AT 149A
       DO 155 J=1,NCOL                                                       AT 14C7
 C
       N=J+(I-1)*NCOL+(K-1)*NRC                                              AT 14F4
       IF(IBOUND(N).LE.0)GO TO 155                                           AT 1519
 C
 C-------CALCULATE V
       H=DZERO                                                               AT 152A
       VCC=DZERO                                                             AT 153E
       IC=N-NCOL                                                             AT 1552
       IF(I.NE.1) THEN                                                       AT 155A
         H=CC(IC)                                                            AT 1567
         IF(CD(IC).NE.0.) VCC=H*V(IC)/CD(IC)                                 AT 1581
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
       ENDIF                                                                 AT 15BC
 C
       F=DZERO                                                               AT 15BC
       VCR=DZERO                                                             AT 15D0
       IR=N-1                                                                AT 15E4
       IF(J.NE.1) THEN                                                       AT 15F1
         F=CR(IR)                                                            AT 15FE
         IF(CD(IR).NE.0.) VCR=F*V(IR)/CD(IR)                                 AT 1618
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
       ENDIF                                                                 AT 1653
 C
       S=DZERO                                                               AT 1653
       VCV=DZERO                                                             AT 1667
       IL=N-NRC                                                              AT 167B
       IF(K.NE.1) THEN                                                       AT 168D
         S=CV(IL)                                                            AT 169A
         IF(CD(IL).NE.0.) VCV=S*V(IL)/CD(IL)                                 AT 16B4
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
       ENDIF                                                                 AT 16EF
       V(N)=RHS(N)-VCR-VCC-VCV                                               AT 16EF
 C
 C-------CALCULATE CD - FIRST INTERNAL ITERATION ONLY
       IF(IITER.EQ.1) THEN                                                   AT 171C
           CDCR=DZERO                                                        AT 1729
           CDCC=DZERO                                                        AT 173D
           CDCV=DZERO                                                        AT 1751
           FCC=DZERO                                                         AT 1765
           FCR=DZERO                                                         AT 1779
           FCV=DZERO                                                         AT 178D
           IF(IR.GT.0.AND.CD(IR).NE.0.) CDCR=(F**2)/CD(IR)                   AT 17A1
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
           IF(IC.GT.0.AND.CD(IC).NE.0.) CDCC=(H**2)/CD(IC)                   AT 17E1
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
           IF(IL.GT.0.AND.CD(IL).NE.0.) CDCV=(S**2)/CD(IL)                   AT 182B
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
           IF(NPCOND.EQ.1) THEN                                              AT 1875
               IF(IR.GT.0) THEN                                              AT 1881
                   FV=CV(IR)                                                 AT 188E
 C                 MODIFIED FROM HILL(1990) 9/27/90: 2 REPLACES 1
                   IF(K.EQ.NLAY.AND.((J+I).GT.2)) FV=DZERO                   AT 18AE
                   IF(CD(IR).NE.0.) FCR=(F/CD(IR))*(CC(IR)+FV)               AT 18E8
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
               ENDIF                                                         AT 1931
               IF(IC.GT.0) THEN                                              AT 1931
```

```
                    FV=CV(IC)                                                   AT 193E
                    IF(K.EQ.NLAY.AND.(I.GT.1)) FV=DZERO                         AT 195E
                    IF(CD(IC).NE.0.) FCC=(H/CD(IC))*(CR(IC)+FV)                 AT 1990
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
                ENDIF                                                           AT 19D9
                IF(IL.GT.0) THEN                                                AT 19D9
                    IF(CD(IL).NE.0.) FCV=(S/CD(IL))*(CR(IL)+CC(IL))             AT 19E6
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
                ENDIF                                                           AT 1A33
            ENDIF                                                               AT 1A33
            IF(NORM.EQ.0) THEN                                                  AT 1A33
               B=DZERO                                                          AT 1A40
               H=DZERO                                                          AT 1A54
               D=DZERO                                                          AT 1A68
               F=DZERO                                                          AT 1A7C
               Z=DZERO                                                          AT 1A90
               S=DZERO                                                          AT 1AA4
               IF(I.NE.1) B=CC(IC)                                              AT 1AB8
               IF(I.NE.NROW) H=CC(N)                                            AT 1AE5
               IF(J.NE.1) D=CR(IR)                                              AT 1B16
               IF(J.NE.NCOL) F=CR(N)                                            AT 1B43
               IF(K.NE.1) Z=CV(IL)                                              AT 1B74
               IF(K.NE.NLAY) S=CV(N)                                            AT 1BA1
               HHCOF=HCOF(N)-Z-B-D-F-H-S                                        AT 1BD2
            ENDIF                                                               AT 1C0D
            IF(NORM.EQ.1) HHCOF=-DONE                                           AT 1C0D
            CD(N)=(DONE+DEL)*HHCOF-CDCR-CDCC-CDCV-RELAX*(FCR+FCC+FCV)           AT 1C2E
           IF(CD1.EQ.0..AND.CD(N).NE.0.) CD1=CD(N)                              AT 1C7C
C--------.LT. CHANGED TO .LE. 01SEPT1991
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
           IF(CD(N)*CD1.LE.0.) THEN                                             AT 1CBA
C--------CHANGED 510 FORMAT 01SEPT1991 AND 20MAR1992
C--------CHANGED 510 FORMAT AND WRITE STATEMENT AND ADDED DEL 01MAY1993
             DEL=1.5*DEL+.001                                                   AT 1CE0
             IF(DEL.GT..5) THEN                                                 AT 1CF9
               WRITE(IOUT,510)                                                  AT 1D09
               STOP                                                             AT 1D2F
             ENDIF                                                              AT 1D42
             GO TO 154                                                          AT 1D42
           ENDIF                                                                AT 1D47
        ENDIF                                                                   AT 1D47
510   FORMAT(' MATRIX IS SEVERELY NON-DAGONALLY DOMINANT.  CHECK INPUT',        AT 1D47
     1     ' FILES.  STOP EXECUTION')
C
  155 CONTINUE                                                                  AT 1DAE
C
C-------STEP THROUGH EACH CELL AND SOLVE FOR S OF THE CONJUGATE
C-------GRADIENT ALGORITHM BY BACK SUBSTITUTION. STORE RESULT IN SS.
      DO 160 KK=NLAY,1,-1                                                       AT 1DBD
      DO 160 II=NROW,1,-1                                                       AT 1DE0
      DO 160 JJ=NCOL,1,-1                                                       AT 1E03
C
      N=JJ+(II-1)*NCOL+(KK-1)*NRC                                               AT 1E26
      IF(IBOUND(N).LE.0)GO TO 160                                               AT 1E4F
C
      NC=N+1                                                                    AT 1E60
      NR=N+NCOL                                                                 AT 1E67
      NL=N+NRC                                                                  AT 1E75
      SSCR=DZERO                                                                AT 1E87
      SSCC=DZERO                                                                AT 1E9B
      SSCV=DZERO                                                                AT 1EAF
      IF(JJ.NE.NCOL) SSCR=CR(N)*SS(NC)/CD(N)                                    AT 1EC3
      IF(II.NE.NROW) SSCC=CC(N)*SS(NR)/CD(N)                                    AT 1F08
      IF(KK.NE.NLAY) SSCV=CV(N)*SS(NL)/CD(N)                                    AT 1F4A
      VN=V(N)/CD(N)                                                             AT 1F8C
      SS(N)=VN-SSCR-SSCC-SSCV                                                   AT 1FAD
  160 CONTINUE                                                                  AT 1FCA
C-------SKIP OVER OTHER PRECONDITIONING TYPES
      GO TO 199                                                                 AT 1FD9
  165 CONTINUE                                                                  AT 1FDE
C
C-------POLYNOMIAL PRECONDITIONING
      DO 170 N=1,NODES                                                          AT 1FDE
      V(N)=RHS(N)                                                               AT 200B
  170 CONTINUE                                                                  AT 2027
      CALL SPCG2E(IBOUND,RHS,HCOF,CR,CC,CV,V,SS,C2,NORM,NCOL,NROW,              AT 2029
     1            NLAY,NODES)
      CALL SPCG2E(IBOUND,RHS,HCOF,CR,CC,CV,SS,V,C1,NORM,NCOL,NROW,              AT 207B
     1            NLAY,NODES)
      CALL SPCG2E(IBOUND,RHS,HCOF,CR,CC,CV,V,SS,C0,NORM,NCOL,NROW,              AT 20CD
     1            NLAY,NODES)
```

```
  199 CONTINUE                                                            AT 211F
C
C-------CALCULATE P OF THE CONJUGATE GRADIENT ALGORITHM
      SROLD=SRNEW                                                         AT 211F
      SRNEW=DZERO                                                         AT 2137
      DO 200 N=1,NODES                                                    AT 214B
        IF(IBOUND(N).LE.0)GO TO 200                                       AT 2178
        SRNEW=SRNEW+SS(N)*RHS(N)                                          AT 2189
  200 CONTINUE                                                            AT 21AA
C
      IF(IITER.EQ.1) THEN                                                 AT 21AC
         DO 205 N=1,NODES                                                 AT 21B9
  205    P(N)=SS(N)                                                       AT 21E6
      ELSE                                                                AT 220A
         DO 210 N=1,NODES                                                 AT 220F
  210    P(N)=SS(N)+(SRNEW/SROLD)*P(N)                                    AT 223C
      ENDIF                                                               AT 2269
C
C-------CALCULATE ALPHA OF THE CONJUGATE GRADIENT ROUTINE.
C-------FOR THE DENOMINATOR OF ALPHA, MULTIPLY THE MATRIX A BY THE
C-------VECTOR P, AND STORE IN V; THEN MULTIPLY P BY V.  STORE IN PAP.
      PAP=DZERO                                                           AT 2269
      DO 290 K=1,NLAY                                                     AT 227D
      DO 290 I=1,NROW                                                     AT 22AA
      DO 290 J=1,NCOL                                                     AT 22D7
C
      N=J+(I-1)*NCOL+(K-1)*NRC                                            AT 2304
      V(N)=0.                                                             AT 2329
      IF(IBOUND(N).LE.0)GO TO 290                                         AT 233F
C
      NRN=N+NCOL                                                          AT 2350
      NRL=N-NCOL                                                          AT 2358
      NCN=N+1                                                             AT 2366
      NCL=N-1                                                             AT 2373
      NLN=N+NRC                                                           AT 2380
      NLL=N-NRC                                                           AT 2392
C
      NCF=N                                                               AT 23A4
      NCD=NCL                                                             AT 23B0
      NRB=NRL                                                             AT 23BC
      NRH=N                                                               AT 23C8
      NLS=N                                                               AT 23D4
      NLZ=NLL                                                             AT 23DA
C
      B=DZERO                                                             AT 23E6
      IF(I.NE.1) B=CC(NRB)                                                AT 23FA
      H=DZERO                                                             AT 2427
      IF(I.NE.NROW)H=CC(NRH)                                              AT 243B
      D=DZERO                                                             AT 246C
      IF(J.NE.1) D=CR(NCD)                                                AT 2480
      F=DZERO                                                             AT 24AD
      IF(J.NE.NCOL) F=CR(NCF)                                             AT 24C1
      Z=DZERO                                                             AT 24F2
      IF(K.NE.1) Z=CV(NLZ)                                                AT 2506
      S=DZERO                                                             AT 2533
      IF(K.NE.NLAY) S=CV(NLS)                                             AT 2547
C
      IF(NORM.EQ.0) PN=P(N)                                               AT 2578
      IF(NORM.EQ.1) PN=DZERO                                              AT 25A5
      BHNEW=DZERO                                                         AT 25C6
      HHNEW=DZERO                                                         AT 25DA
      DHNEW=DZERO                                                         AT 25EE
      FHNEW=DZERO                                                         AT 2602
      ZHNEW=DZERO                                                         AT 2616
      SHNEW=DZERO                                                         AT 262A
      IF(NRL.GT.0) BHNEW=B*(P(NRL)-PN)                                    AT 263E
      IF(NRN.LE.NODES) HHNEW=H*(P(NRN)-PN)                                AT 266E
      IF(NCL.GT.0) DHNEW=D*(P(NCL)-PN)                                    AT 269C
      IF(NCN.LE.NODES) FHNEW=F*(P(NCN)-PN)                                AT 26CC
      IF(NLL.GT.0) ZHNEW=Z*(P(NLL)-PN)                                    AT 26FA
      IF(NLN.LE.NODES) SHNEW=S*(P(NLN)-PN)                                AT 272A
C
C-------CALCULATE THE PRODUCT OF MATRIX A AND VECTOR P AND STORE
C-------RESULT IN V.
      PN=HCOF(N)*P(N)                                                     AT 2758
      IF(NORM.EQ.1) PN=-P(N)                                              AT 2779
      VN=ZHNEW+BHNEW+DHNEW+PN+FHNEW+HHNEW+SHNEW                           AT 2793
      V(N)=VN                                                             AT 27C4
      PAP=PAP+P(N)*VN                                                     AT 27E4
  290 CONTINUE                                                            AT 27FD
C
C-------CALCULATE ALPHA
```

```
          ALPHA=1.                                                        AT 280C
          IF(PAP.EQ.0..AND.MXITER.EQ.1) THEN                              AT 2820
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
            WRITE(IOUT,520)                                               AT 2842
            STOP                                                          AT 2868
          ENDIF                                                           AT 287B
    520     FORMAT(/,' CONJUGATE-GRADIENT METHOD FAILED.',/,' SET MXITER',  AT 287B
     1           ' GREATER THAN ONE AND TRY AGAIN.  STOP EXECUTION')
          IF(PAP.NE.0.) ALPHA=SRNEW/PAP                                   AT 28FC
C
C-------CALCULATE NEW HEADS AND RESIDUALS, AND SAVE THE LARGEST
C-------CHANGE IN HEAD AND THE LARGEST VALUE OF THE RESIDUAL.
WARNING - The use of .EQ. or .NE. with non-integer operands can produce misleading results
          DO 300 K=1,NLAY                                                 AT 2925
          DO 300 I=1,NROW                                                 AT 2952
          DO 300 J=1,NCOL                                                 AT 297F
C
          N=J+(I-1)*NCOL+(K-1)*NRC                                        AT 29AC
          IF(IBOUND(N).LE.0) GO TO 300                                    AT 29D1
C
C-------HEAD
          HCHGN=ALPHA*P(N)                                                AT 29E2
          IF(DABS(HCHGN).GT.ABS(BIGH)) THEN                               AT 29F9
              BIGH=HCHGN                                                  AT 2A11
              IH=I                                                        AT 2A29
              JH=J                                                        AT 2A35
              KH=K                                                        AT 2A41
              NH=N                                                        AT 2A4D
          ENDIF                                                           AT 2A53
          HNEW(N)=HNEW(N)+DAMP*HCHGN                                      AT 2A53
C
C--------RESIDUAL (V IS THE PRODUCT OF MATRIX A AND VECTOR P)
          RCHGN=-ALPHA*V(N)                                               AT 2A73
          RHS(N)=RHS(N)+DAMP*RCHGN                                        AT 2A8C
          IF(ABS(RHS(N)).GT.ABS(BIGR)) THEN                               AT 2A9D
              BIGR=RHS(N)                                                 AT 2AB7
              IR=I                                                        AT 2AD1
              JR=J                                                        AT 2ADD
              KR=K                                                        AT 2AE9
              NR=N                                                        AT 2AF5
          ENDIF                                                           AT 2AFB
    300 CONTINUE                                                          AT 2AFB
C
C-------UNSCALE LARGEST CHANGE IN HEAD AND LARGEST RESIDUAL, AND
C-------CHECK THE CONVERGENCE CRITERION
           IF(NORM.EQ.1) THEN                                             AT 2B0A
              BIGH=BIGH/SQRT(-HCOF(NH))                                   AT 2B17
              BIGR=BIGR*SQRT(-HCOF(NR))                                   AT 2B3A
           ENDIF                                                          AT 2B55
           IF(MXITER.EQ.1) THEN                                           AT 2B55
              IF(ABS(BIGH).LE.HCLOSE.AND.ABS(BIGR).LE.RCLOSE) ICNVG=1     AT 2B61
           ELSE                                                           AT 2BA1
              IF(IITER.EQ.1.AND.                                          AT 2BA6
     1           ABS(BIGH).LE.HCLOSE.AND.ABS(BIGR).LE.RCLOSE) ICNVG=1
           ENDIF                                                          AT 2BF3
           IF(ABS(BIGH).LE.HCLOSE.AND.ABS(BIGR).LE.RCLOSE) IICNVG=1       AT 2BF3
C
C-------STORE THE LARGEST UNSCALED HEAD CHANGE AND RESIDUAL VALUE
C-------(THIS ITERATION) AND THEIR LOCATIONS.
          II=NITER                                                        AT 2C31
          HCHG(II)=BIGH                                                   AT 2C3F
          LHCH(1,II)=KH                                                   AT 2C59
          LHCH(2,II)=IH                                                   AT 2C6E
          LHCH(3,II)=JH                                                   AT 2C85
C
          RCHG(II)=BIGR                                                   AT 2C9C
          LRCH(1,II)=KR                                                   AT 2CBC
          LRCH(2,II)=IR                                                   AT 2CD1
          LRCH(3,II)=JR                                                   AT 2CE8
C
          IT1(II)=0                                                       AT 2CFF
          IF(IITER.EQ.1) IT1(II)=1                                        AT 2D10

C-Environment Agency (UK)-12.Debugging output of unconverged heads>>>>>>
          if(ipcgdebug.gt.0)then                                          AT 2D25
C-------copy HNEW to BUFF
            DO 311 N=1,NODES                                              AT 2D31
              IF(NORM.EQ.1.and.IBOUND(N).ne.0)then                        AT 2D5E
                HHCOF=SQRT(-HCOF(N))                                      AT 2D7C
                buff(N)=HNEW(N)/HHCOF                                     AT 2D91
              else                                                        AT 2DA5
                buff(N)=HNEW(N)                                           AT 2DAA
```

```
           ENDIF                                                              AT 2DC9
   311   CONTINUE                                                             AT 2DC9
C-------write heads to debug file
         IF(IXSEC.EQ.0) THEN                                                  AT 2DCE
           DO 79 K=1,NLAY                                                     AT 2DDA
             KK=K                                                             AT 2E07
             i=nrc*(k-1)+1                                                    AT 2E0D
             IF(CHEDFM.EQ.' ') THEN                                           AT 2E1C
               CALL ULASAV(buff(i),TEXT,iiter,kiter,bigh,bigr,NCOL,           AT 2E30
     1                 NROW,KK,ipcgdebug)
             ELSE                                                             AT 2E7E
               CALL ULASV2(BUFF(i),TEXT,iiter,kiter,bigh,bigr,NCOL,           AT 2E83
     1                 NROW,KK,ipcgdebug,CHEDFM,LBHDSV,IBOUND(i))
             END IF                                                           AT 2EE8
    79     CONTINUE                                                           AT 2EE8
         ELSE                                                                 AT 2EED
           IF(CHEDFM.EQ.' ') THEN                                             AT 2EF2
              CALL ULASAV(BUFF,TEXT,iiter,kiter,bigh,bigr,NCOL,               AT 2F06
     1                 NLAY,-1,ipcgdebug)
           ELSE                                                               AT 2F47
              CALL ULASV2(BUFF,TEXT,iiter,kiter,bigh,bigr,NCOL,               AT 2F4C
     1                   NLAY,-1,ipcgdebug,CHEDFM,LBHDSV,IBOUND)
           END IF                                                             AT 2F9F
         END IF                                                               AT 2F9F
       endif                                                                  AT 2F9F
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<

C-------GO TO NEXT INTERNAL ITERATION IF CONVERGENCE HAS NOT BEEN
C-------REACHED AND IITER IS LESS THAN ITER1
       IF(MXITER.EQ.1) THEN                                                   AT 2F9F
         IF(ICNVG.EQ.0.AND.IITER.LT.ITER1) GO TO 153                          AT 2FAB
       ELSE                                                                   AT 2FCB
         IF(IICNVG.EQ.0.AND.IITER.LT.ITER1) GO TO 153                         AT 2FD0
       ENDIF                                                                  AT 2FEE
C
C-------UNSCALE CR,CC,CV AND HNEW
       IF(NORM.EQ.1) THEN                                                     AT 2FEE
       DO 310 N=1,NODES                                                       AT 2FFB
       IF(IBOUND(N).EQ.0) GO TO 310                                           AT 3028
       HHCOF=SQRT(-HCOF(N))                                                   AT 3039
       IF(N.LE.(NODES-NCOL).AND.CC(N).GT.0.)                                  AT 304E
      #   CC(N)=CC(N)*(HHCOF*(SQRT(-HCOF(N+NCOL))))
       IF(N.LE.(NODES-1).AND.CR(N).GT.0.)                                     AT 30A3
      #   CR(N)=CR(N)*(HHCOF*(SQRT(-HCOF(N+1))))
       IF(N.LE.(NODES-NRC).AND.CV(N).GT.0.)                                   AT 30F8
      #   CV(N)=CV(N)*(HHCOF*(SQRT(-HCOF(N+NRC))))
       HNEW(N)=HNEW(N)/HHCOF                                                  AT 315E
   310 CONTINUE                                                               AT 3179
       ENDIF                                                                  AT 317E
C
C-Environment Agency (UK)-11.Force converge if NOUTC outer itrs conv>>>>
       IF(MXITER.ne.1.and.IICNVG.EQ.1) nout=nout+1                            AT 317E
       if(nout.eq.noutc.and.noutc.ne.0)then                                   AT 319C
         write(IOUT,503)nout                                                  AT 31B5
         ICNVG=1                                                              AT 31EB
       endif                                                                  AT 31F7
  503  format(' PCG2 CONVERGENCE STOPPED AFTER',i6,' CONSECUTIVE OUTER '      AT 31F7
      1      ,'ITERATIONS WITH ERROR CRITERIA ACHIEVED.'/
      2      ,' CAREFULLY LOOK AT MASS BALANCE TO VALIDATE RESULTS.')
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C-------IF END OF TIME STEP, PRINT # OF ITERATIONS THIS STEP
       IF(ICNVG.EQ.0 .AND. KITER.NE.MXITER) GO TO 600                         AT 32B0
       IF(MUTPCG.GT.1) GO TO 590                                              AT 32D2
       IF(KSTP.EQ.1) WRITE(IOUT,500)                                          AT 32DE
   500 FORMAT(1H0)                                                            AT 3313
       WRITE(IOUT,501) KITER,KSTP,KPER,NITER                                  AT 3328
   501 FORMAT(1X,I5,' CALLS TO PCG ROUTINE FOR TIME STEP',I4,                 AT 339A
      1' IN STRESS PERIOD',I3,/1X,I5,' TOTAL ITERATIONS')
       IF(MUTPCG.EQ.1) GO TO 590                                              AT 3421
C
C-------PRINT HEAD CHANGE EACH ITERATION IF PRINTOUT INTERVAL IS REACHED
       IF(ICNVG.EQ.0 .OR. KSTP.EQ.NSTP .OR. MOD(KSTP,IPRPCG).EQ.0)            AT 342D
      1      CALL SPCG2P(HCHG,LHCH,RCHG,LRCH,
      2                  ITER1,NITER,MXITER,IOUT,NPCOND,BPOLY,IT1)
C-Environment Agency (UK)-10.Convergence info output to screen>>>>>>>>>>
c590   NITER=0
590   write(*,502)KPER,KSTP,NITER,HCHG(NITER),RCHG(NITER)                     AT 349C
 502  format(' |',i7,' |',i5,' |',i6,' |',g11.4,' |',g11.4,' |')              AT 353D
       NITER=0                                                                AT 3592
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
```

```
C-------RETURN
600   RETURN                                                                 AT 359E
C
      END                                                                    AT 35A3
WARNING - Label 895 has not been referenced
WARNING - Label 900 has not been referenced


      SUBROUTINE SPCG2P(HCHG,LHCH,RCHG,LRCH,ITER1,
     1                  NITER,MXITER,IOUT,NPCOND,BPOLY,IT1)
C
C
C-----VERSION 0001 01MAY1988 SPCG2P
C     ******************************************************************
C     PRINT MAXIMUM HEAD CHANGE AND RESIDUAL VALUE FOR EACH ITERATION
C                          DURING A TIME STEP
C     ******************************************************************
C
C        SPECIFICATIONS:
C     ------------------------------------------------------------------
C
      DIMENSION HCHG(MXITER*ITER1), LHCH(3,MXITER*ITER1)
      DIMENSION RCHG(MXITER*ITER1), LRCH(3,MXITER*ITER1)
      DIMENSION IT1(MXITER*ITER1)
C     ------------------------------------------------------------------
C
      IF(NPCOND.EQ.2) WRITE(IOUT,2) BPOLY                                    AT 0033
    2 FORMAT(1H0,'B OF THE POLYNOMIAL PRECONDITIONING METHOD: ',E12.4)       AT 0075
      WRITE(IOUT,5)                                                          AT 00C2
    5 FORMAT(1H0,'MAXIMUM HEAD CHANGE FOR EACH ITERATION (1 INDICATES ',     AT 00E8
     A      'THE FIRST INNER ITERATION):'/
     1    1H0,4('    HEAD CHANGE  LAYER,ROW,COL')/1X,120('-'))
      WRITE (IOUT,10) (IT1(J),HCHG(J),(LHCH(I,J),I=1,3),J=1,NITER)           AT 01A3
      WRITE(IOUT,11)                                                         AT 024E
      WRITE(IOUT,15)                                                         AT 0274
   15 FORMAT(1H0,'MAXIMUM RESIDUAL FOR EACH ITERATION (1 INDICATES ',        AT 029A
     A      'THE FIRST INNER ITERATION):'/
     1    1H0,4('     RESIDUAL    LAYER,ROW,COL')/1X,120('-'))
      WRITE (IOUT,10) (IT1(J),RCHG(J),(LRCH(I,J),I=1,3),J=1,NITER)           AT 0353
      WRITE(IOUT,11)                                                         AT 03FE
C
      RETURN                                                                 AT 0424
C
   10 FORMAT((1X,4(2X,I1,1X,G12.4,' (',I3,',',I3,',',I3,')')))               AT 0429
   11 FORMAT(1H0)                                                            AT 0490
C
      END                                                                    AT 04A5


      SUBROUTINE SPCG2E(IBOUND,RHS,HCOF,CR,CC,CV,VIN,VOUT,C,NORM,NCOL,
     1                  NROW,NLAY,NODES)
C
C
C-----VERSION 0001 01MAY1989 SPCG2E
C     ******************************************************************
C           MATRIX MULTIPLICATIONS FOR POLYNOMIAL PRECONDITIONING
C     ******************************************************************
C
C        SPECIFICATIONS:
C     ------------------------------------------------------------------
C
      DOUBLE PRECISION VN,CRHS,Z,B,D,F,H,S,ZV,BV,DV,FV,HV,SV,DZERO
      DIMENSION IBOUND(NODES),CR(NODES),CC(NODES),CV(NODES),
     1  RHS(NODES),VIN(NODES),VOUT(NODES),HCOF(NODES)
C     ------------------------------------------------------------------
C
      DZERO=0.                                                               AT 0026
      NRC=NROW*NCOL                                                          AT 0034
      DO 290 K=1,NLAY                                                        AT 0042
      DO 290 I=1,NROW                                                        AT 0060
      DO 290 J=1,NCOL                                                        AT 007E
C
      N=J+(I-1)*NCOL+(K-1)*NRC                                               AT 009C
      VOUT(N)=0.                                                             AT 00B5
      IF(IBOUND(N).LE.0)GO TO 290                                            AT 00C8
C
      NRN=N+NCOL                                                             AT 00D6
      NRL=N-NCOL                                                             AT 00DB
      NCN=N+1                                                                AT 00E6
      NCL=N-1                                                                AT 00F0
```

```
      NLN=N+NRC                                                     AT 00FA
      NLL=N-NRC                                                     AT 0106
C
      NCF=N                                                         AT 0112
      NCD=NCL                                                       AT 011B
      NRB=NRL                                                       AT 0127
      NRH=N                                                         AT 0133
      NLS=N                                                         AT 013C
      NLZ=NLL                                                       AT 0142
C
      B=DZERO                                                       AT 014E
      BV=DZERO                                                      AT 0160
      IF(I.NE.1.AND.IBOUND(NRL).GE.0) THEN                          AT 0172
        B=CC(NRB)                                                   AT 018D
        BV=B*VIN(NRL)                                               AT 01AA
      ENDIF                                                         AT 01C4
      H=DZERO                                                       AT 01C4
      HV=DZERO                                                      AT 01D6
      IF(I.NE.NROW.AND.IBOUND(NRN).GE.0) THEN                       AT 01E8
        H=CC(NRH)                                                   AT 0207
        HV=H*VIN(NRN)                                               AT 0224
      ENDIF                                                         AT 023B
      D=DZERO                                                       AT 023B
      DV=DZERO                                                      AT 024D
      IF(J.NE.1.AND.IBOUND(NCL).GE.0) THEN                          AT 025F
        D=CR(NCD)                                                   AT 027D
        DV=D*VIN(NCL)                                               AT 029A
      ENDIF                                                         AT 02B4
      F=DZERO                                                       AT 02B4
      FV=DZERO                                                      AT 02C6
      IF(J.NE.NCOL.AND.IBOUND(NCN).GE.0) THEN                       AT 02D8
        F=CR(NCF)                                                   AT 02FA
        FV=F*VIN(NCN)                                               AT 0317
      ENDIF                                                         AT 0331
      Z=DZERO                                                       AT 0331
      ZV=DZERO                                                      AT 0343
C       IF STATEMENT REARRANGED 01JUN1993
      IF(K.NE.1) THEN                                               AT 0355
        IF(IBOUND(NLL).GE.0) THEN                                   AT 035F
          Z=CV(NLZ)                                                 AT 0373
          ZV=Z*VIN(NLL)                                             AT 0390
        ENDIF                                                       AT 03AA
      ENDIF                                                         AT 03AA
      S=DZERO                                                       AT 03AA
      SV=DZERO                                                      AT 03BC
      IF(K.NE.NLAY.AND.IBOUND(NLN).GE.0) THEN                       AT 03CE
        S=CV(NLS)                                                   AT 03F0
        SV=S*VIN(NLN)                                               AT 040D
      ENDIF                                                         AT 0427
C
C-------CALCULATE THE PRODUCT OF MATRIX A AND VECTOR VIN AND STORE
C------ RESULT IN VOUT
      VN=HCOF(N)*VIN(N)                                             AT 0427
      IF(NORM.EQ.1) VN=-VIN(N)                                      AT 043F
      CRHS=C*RHS(N)                                                 AT 0458
      VOUT(N)=CRHS+ZV+BV+DV+VN+FV+HV+SV                             AT 046E
  290 CONTINUE                                                      AT 04A0
      RETURN                                                        AT 04AF
      END                                                           AT 04B4
```

End of compilation - Clocked 0.0 seconds

APPENDIX D
Modified computer code for MODFLOW utilities package with changes highlighted

Salford Software Ltd FTN77 Ver. 3.53   C:\Company\Tech\MF-VKD\Source_code\utl-vkd2.for   14:34:01
Monday, 8 September 2003

COMPILER OPTIONS: listing intl no_persist nodclvar nomap nocheck logl dynm offset dreal noansi nopagethrow
                   nosilent no_optimise warn73 nolink no_link

```
C-Changes to the code are highlighted with the following headings:
C
C-Environment Agency (UK)-9.Allow direct access binary output files>>>>>
C-Environment Agency (UK)-10.Allow double precn dir access bin output>>>
C
C These headings are followed by the original USGS MODFLOW-96 code
C (commented out), followed by the modified code. All the code changes
C are followed by the following:
C
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
      SUBROUTINE UBUDSV(KSTP,KPER,TEXT,IBDCHN,BUFF,NCOL,NROW,NLAY,IOUT)
C
C
C-----VERSION 1039 26JUNE1992 UBUDSV
C     ******************************************************************
C     RECORD CELL-BY-CELL FLOW TERMS FOR ONE COMPONENT OF FLOW.
C     ******************************************************************
C
C        SPECIFICATIONS:
C     ------------------------------------------------------------------
C-Environment Agency (UK)-9.Allow direct access binary output files>>>>>
c      CHARACTER*16 TEXT
c      DIMENSION BUFF(NCOL,NROW,NLAY)
      character text*4,acc*10
      DIMENSION BUFF(NCOL,NROW,NLAY),text(4)
      character fname*100
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C     ------------------------------------------------------------------
C
C1------WRITE AN UNFORMATTED RECORD IDENTIFYING DATA.
      WRITE(IOUT,1) TEXT,IBDCHN,KSTP,KPER                               AT 004C
C-Environment Agency (UK)-9.Allow direct access binary output files>>>>>
c    1 FORMAT(1X,'UBUDSV SAVING "',A16,'" ON UNIT',I3,
c    1     ' AT TIME STEP',I3,', STRESS PERIOD',I3)
    1 FORMAT(1X,'UBUDSV SAVING "',4A4,'" ON UNIT',I3,                   AT 00B8
     1     ' AT TIME STEP',I3,', STRESS PERIOD',I3)
c
c1a-----CHECK ACCESS OF FILE
      inquire(ibdchn,access=acc)                                        AT 012B
      if(acc.eq.'DIRECT')then                                           AT 014F
        inquire(ibdchn,recl=irecl,nextrec=nr)                           AT 0165
        if(nr.eq.1)then                                                 AT 018C
          inquire(ibdchn,name=fname)                                    AT 0196
          close(ibdchn,status='delete')                                 AT 01BD
C-Environment Agency (UK)-10.Allow double precn dir access bin output>>>
c          irecl=((ncol*nrow*nlay)+9)*4
          irecl=((ncol*nrow*nlay*2)+9)*4                                AT 01E2
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
          open(ibdchn,file=fname,access='direct',recl=irecl)            AT 01FE
        endif                                                           AT 023E
        write(ibdchn,rec=nr)KSTP,KPER,TEXT,NCOL,NROW,nLAY,              AT 023E
     1              (((BUFf(IC,IR,il),IC=1,NCOL),IR=1,NROW),il=1,nlay)
        goto 999                                                        AT 03B1
      endif                                                             AT 03B6
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      WRITE(IBDCHN) KSTP,KPER,TEXT,NCOL,NROW,NLAY                       AT 03B6
C
C2------WRITE AN UNFORMATTED RECORD CONTAINING VALUES FOR
C2------EACH CELL IN THE GRID.
      WRITE(IBDCHN) BUFF                                                AT 0454
C
C-Environment Agency (UK)-9.Allow direct access binary output files>>>>>
 999  continue                                                          AT 0482
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C3------RETURN
      RETURN                                                            AT 0482
      END                                                               AT 0487


      SUBROUTINE ULASAV(BUF,TEXT,KSTP,KPER,PERTIM,TOTIM,NCOL,
     1                   NROW,ILAY,ICHN)
C
C-----VERSION 1642 12MAY1987 ULASAV
```

```
C      ****************************************************************
C      SAVE 1 LAYER ARRAY ON DISK
C      ****************************************************************
C
C         SPECIFICATIONS:
C      ----------------------------------------------------------------
       CHARACTER*4 TEXT
       DIMENSION BUF(NCOL,NROW),TEXT(4)
C-Environment Agency (UK)-9.Allow direct access binary output files>>>>>
       character*10 ACC
       character fname*100
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C      ----------------------------------------------------------------
C-Environment Agency (UK)-9.Allow direct access binary output files>>>>>
c0------CHECK ACCESS OF HEADS FILE
       inquire(ichn,access=acc)                                          AT 003B
       if(acc.eq.'DIRECT')then                                           AT 005F
         inquire(ichn,recl=irecl,nextrec=nr)                             AT 0075
         if(nr.eq.1)then                                                 AT 009C
           inquire(ichn,name=fname)                                      AT 00A6
           close(ichn,status='delete')                                   AT 00CD
C-Environment Agency (UK)-10.Allow double precn dir access bin output>>>
c           irecl=((ncol*nrow)+11)*4
           irecl=((ncol*nrow*2)+13)*4                                    AT 00F2
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
           open(ichn,file=fname,access='direct',recl=irecl)              AT 0108
         endif                                                           AT 0148
         write(ichn,rec=nr)KSTP,KPER,PERTIM,TOTIM,TEXT,NCOL,NROW,ILAY,   AT 0148
      1                        ((BUF(IC,IR),IC=1,NCOL),IR=1,NROW)
         goto 999                                                        AT 02A6
       endif                                                             AT 02AB
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C1------WRITE AN UNFORMATTED RECORD CONTAINING IDENTIFYING
C1------INFORMATION.
       WRITE(ICHN) KSTP,KPER,PERTIM,TOTIM,TEXT,NCOL,NROW,ILAY            AT 02AB
C
C2------WRITE AN UNFORMATTED RECORD CONTAINING ARRAY VALUES
C2------THE ARRAY IS DIMENSIONED (NCOL,NROW)
       WRITE(ICHN) ((BUF(IC,IR),IC=1,NCOL),IR=1,NROW)                    AT 0373
C
C-Environment Agency (UK)-9.Allow direct access binary output files>>>>>
 999  continue                                                           AT 041A
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C3------RETURN
       RETURN                                                            AT 041A
       END                                                               AT 041F
```

APPENDIX E
Modified computer code for main MODFLOW program with changes highlighted

Salford Software Ltd FTN77 Ver. 3.53 C:\Company\Tech\MF-VKD\Source_code\mf-vkd3.for 14:34:00
Monday, 8 September 2003

COMPILER OPTIONS: listing intl no_persist nodclvar nomap nocheck logl dynm offset dreal noansi nopagethrow
nosilent no_optimise warn73 nolink no_link

```
C     ******************************************************************
C     MODFLOW-VKD Version 3 (14 MAY 2003)
C           MODIFIED TO INCLUDE VARIATIONS IN HYDRAULIC CONDUCTIVITY AND
C           STORAGE WITH DEPTH.
C           BY THE ENVIRONMENT AGENCY (UK) & WATER MANAGEMENT
C           CONSULTANTS LTD. - A. Taylor, P. Hulme and A. Hughes
C     MODFLOW-VKD Version 1 documented in:
C           Environment Agency, 2002a, Enhancements to MODFLOW -
c           Variation in hydraulic conductivity and storage with depth.
c           WMC Report 1621/R1 prepared for the Environment Agency,
c           National Groundwater and Contaminated Land Centre,
c           Project NC/00/23.
c     and
c           Environment Agency, 2002b, Enhancements to MODFLOW -
c           User guide for MODFLOW-VKD - A modified version of
c           MODFLOW-96 to include variations in hydraulic properties
c           with depth. WMC Report 1621/R2 prepared for the Environment
c           Agency, National Groundwater and Contaminated Land Centre,
c           Project NC/00/23.
C
C-Changes to the code are highlighted with the following headings:
C
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
C-Environment Agency (UK)-1.Variable hydraulic cond. w/depth (VKD)>>>>>>
C-Environment Agency (UK)-2.Variable storage with depth (VSD)>>>>>>>>>>>
C-Environment Agency (UK)-3.Auto-conversion for steady state VKD>>>>>>>>
C-Environment Agency (UK)-4.Spatially variable anisotropy>>>>>>>>>>>>>>>
C-Environment Agency (UK)-5.Inter-nodal transmissivities>>>>>>>>>>>>>>>>
C-Environment Agency (UK)-6.X & Y-dir transmissivities>>>>>>>>>>>>>>>>>>
C-Environment Agency (UK)-7.Explicit transmissivity calculation>>>>>>>>>
C-Environment Agency (UK)-8.Output calculated transmissivities>>>>>>>>>>
C-Environment Agency (UK)-9.Allow direct access binary output files>>>>>
C-Environment Agency (UK)-10.Convergence info output to screen>>>>>>>>>>
C-Environment Agency (UK)-11.Force converge if NOUTC outer itrs conv>>>>
C-Environment Agency (UK)-12.Debugging output of unconverged heads>>>>>>
C-Environment Agency (UK)-13.Allow an inflow to any stream reach>>>>>>>>
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
C-Environment Agency (UK)-15.Save single flow if >1 stream in a cell>>>>
C-Environment Agency (UK)-16.K gradient independant of kbase>>>>>>>>>>>>
C-Environment Agency (UK)-17.Output calc trans to binary file>>>>>>>>>>>
C
C These headings are followed by the original USGS MODFLOW-96 code
C (commented out), followed by the modified code. All the code changes
C are followed by the following:
C
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C-Based on USGS:
C     ******************************************************************
C     MAIN CODE FOR U.S. GEOLOGICAL SURVEY MODULAR MODEL -- MODFLOW-96
C           BY MICHAEL G. MCDONALD AND ARLEN W. HARBAUGH
C     MODFLOW-88 documented in:
C           McDonald, M.G. and Harbaugh, A.W., 1988, A modular
C           three-dimensional finite-difference ground-water flow
C           model: U.S. Geological Survey Techniques of Water
C           Resources Investigations, Book 6, Chapter A1, 586 p.
C     MODFLOW-96 documented in:
C           Harbaugh, A.W. and McDonald, M.G., 1996, User's
C           documentation for the U.S. Geological Survey modular
C           finite-difference ground-water flow model: U.S. Geological
C           Survey Open-File Report 96-485
C-----VERSION 0950 23MAY1996 MAIN
C-----VERSION 1401 03DEC1996 -- added PCG2, STR1, IBS1, CHD1, GFD1,
C                               HFB1, TLK1, DE45, and RES1 as documented
C                               in USGS reports
C     ******************************************************************
C
C        SPECIFICATIONS:
C     -------------------------------------------------------------------
C1------SPECIFY THE SIZE OF THE X ARRAY.  TO CHANGE THE SIZE OF THE
C1------X ARRAY, CHANGE VALUE OF LENX IN THE NEXT STATEMENT.
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
c      PARAMETER (LENX=1500000)
      PARAMETER (LENX=10000000)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
```

```
      COMMON X(LENX)
      COMMON /FLWCOM/LAYCON(200)
      CHARACTER*16 VBNM(40)
      CHARACTER*80 HEADNG(2)
      DIMENSION VBVL(4,40),IUNIT(40)
      DOUBLE PRECISION DUMMY
      EQUIVALENCE (DUMMY,X(1))
      CHARACTER*20 CHEDFM,CDDNFM
      CHARACTER*80 FNAME
C-Environment Agency (UK)-3.Auto-conversion for steady state VKD>>>>>>>>
      logical exists2
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      LOGICAL EXISTS
      CHARACTER*4 CUNIT(40)
      DATA CUNIT/'BCF ','WEL ','DRN ','RIV ','EVT ','TLK ','GHB ',           AT 0017
     1           'RCH ','SIP ','DE4 ','SOR ','OC  ','PCG ','GFD ',
     2           '    ','HFB ','RES ','STR ','IBS ','CHD ','FHB ',
     3           '    ','    ','    ','    ','    ','    ','    ',
     4           '    ','    ','    ','    ','    ','    ','    ',
     5           '    ','    ','    ','    ','    '/
C     ------------------------------------------------------------------
C     set string for use if RCS ident command
      FNAME =                                                                AT 0017
     &'$Id: modflw96.f,v 3.2 1998/01/09 19:19:39 rsregan Exp rsregan $'
      FNAME =                                                                AT 0027
     &   '@(#)MODFLOW-96 - Modular 3-D Finite-Difference GW Flow Model'
      FNAME = '@(#)MODFLOW-96 - USGS TWRI, Book 6, Chap. A1, McDonald an      AT 0043
     &d Harbaugh'
      FNAME = '@(#)MODFLOW-96 - USGS OFR 96-485, Harbaugh and McDonald'      AT 0053
      FNAME =                                                                AT 0073
     &    '@(#)MODFLOW-96 - Contact: h2osoft@usgs.gov'
      FNAME = '@(#)MODFLOW-96 - Version: 3.0 1996/12/03 includes MOC'        AT 0093
      FNAME =                                                                AT 00B3
     &    '@(#)MODFLOW-96 - Version: 3.1 1997/03/11 fixed HFB1FM call'
      FNAME = '@(#)MODFLOW-96 - Version: 3.2x 1998/01/09 includes FHB'       AT 00D3
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
      FNAME = '@(#)MODFLOW-VKD - Version: 1.0 18/07/2001 includes VKD'       AT 00F3
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C     ------------------------------------------------------------------
      INUNIT=99                                                              AT 0113
      IBUNIT=98                                                              AT 011A
      IBOUTS=97                                                              AT 0121
C-Environment Agency (UK)-3.Auto-conversion for steady state VKD>>>>>>>>
      ibcfoutunit=96                                                         AT 0128
      inoutunit=95                                                           AT 012F
      ioutbas=94                                                             AT 0136
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      IBATCH=0                                                               AT 013D
      INQUIRE(FILE='modflow.bf',EXIST=EXISTS)                                AT 0144
      IF(EXISTS) THEN                                                        AT 016A
         IBATCH=1                                                            AT 0174
         OPEN(UNIT=IBUNIT,FILE='modflow.bf',STATUS='OLD')                    AT 017B
         OPEN(UNIT=IBOUTS,FILE='modbatch.rpt')                               AT 01B1
         WRITE(IBOUTS,*) ' USGS MODFLOW MODEL BATCH-MODE REPORT'             AT 01D7
      END IF                                                                 AT 020A
C
C2------OPEN FILE OF FILE NAMES.
50    IF(IBATCH.GT.0) THEN                                                   AT 020A
         READ(IBUNIT,'(A)',END=500) FNAME                                    AT 0214
         IF(FNAME.EQ.' ') GO TO 50                                           AT 0273
         WRITE(IBOUTS,'(1X,/1X,A)') FNAME                                    AT 0281
      ELSE                                                                   AT 02E0
         WRITE(*,*) ' Enter the name of the NAME FILE:'                      AT 02E5
         READ(*,'(A)') FNAME                                                 AT 0319
      END IF                                                                 AT 036D
      INQUIRE(FILE=FNAME,EXIST=EXISTS)                                       AT 036D
      IF(.NOT.EXISTS) THEN                                                   AT 0392
         IF(IBATCH.GT.0) THEN                                                AT 039C
            WRITE(IBOUTS,*) ' Specified name file does not exist.'           AT 03A6
            WRITE(IBOUTS,*) ' Processing will continue with the next ',      AT 03D9
     1                      'name file in modflow.bf.'
         ELSE                                                                AT 0424
            WRITE(*,*) ' File does not exist'                                AT 0429
         END IF                                                              AT 045D
         GO TO 50                                                            AT 045D
      END IF                                                                 AT 0462
C-Environment Agency (UK)-3.Auto-conversion for steady state VKD>>>>>>>>
c       OPEN(UNIT=INUNIT,FILE=FNAME,STATUS='OLD')
 55   OPEN(UNIT=INUNIT,FILE=FNAME,STATUS='OLD')                              AT 0462
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C3------DEFINE PROBLEM--ROWS,COLUMNS,LAYERS,STRESS PERIODS,PACKAGES.
```

```
      CALL BAS5DF(ISUM,HEADNG,NPER,ITMUNI,TOTIM,NCOL,NROW,NLAY,                   AT 0497
     1        NODES,INBAS,IOUT,IUNIT,CUNIT,INUNIT,IXSEC,ICHFLG,IFREFM)
C
C4------ALLOCATE SPACE IN "X" ARRAY.
      CALL BAS5AL(ISUM,LENX,LCHNEW,LCHOLD,LCIBOU,LCCR,LCCC,LCCV,                  AT 051D
     1              LCHCOF,LCRHS,LCDELR,LCDELC,LCSTRT,LCBUFF,LCIOFL,
     2              INBAS,ISTRT,NCOL,NROW,NLAY,IOUT,IAPART,IFREFM)
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
c      IF(IUNIT(1).GT.0) CALL BCF5AL(ISUM,LENX,LCSC1,LCHY,
c     1      LCBOT,LCTOP,LCSC2,LCTRPY,IUNIT(1),ISS,
c     2      NCOL,NROW,NLAY,IOUT,IBCFCB,LCWETD,IWDFLG,LCCVWD,
c     3      WETFCT,IWETIT,IHDWET,HDRY,IAPART,IFREFM)
C-Environment Agency (UK)-16.K gradient independant of kbase>>>>>>>>>>>>
c      IF(IUNIT(1).GT.0) CALL BCF5AL(ISUM,LENX,LCSC1,LCHY,
c     1      LCBOT,LCTOP,LCSC2,LCTRPY,IUNIT(1),ISS,
c     2      NCOL,NROW,NLAY,IOUT,IBCFCB,LCWETD,IWDFLG,LCCVWD,
c     3      WETFCT,IWETIT,IHDWET,HDRY,IAPART,IFREFM,itrpy,lckgrad,ihold,
c     4      itrans,lcmid,lckmax,lcsgrad,lcsmax,lcsmid)
      IF(IUNIT(1).GT.0) CALL BCF5AL(ISUM,LENX,LCSC1,LCHY,                       AT 05C4
     1      LCBOT,LCTOP,LCSC2,LCTRPY,IUNIT(1),ISS,
     2      NCOL,NROW,NLAY,IOUT,IBCFCB,LCWETD,IWDFLG,LCCVWD,
     3      WETFCT,IWETIT,IHDWET,HDRY,IAPART,IFREFM,itrpy,lckgrad,ihold,
     4      itrans,lcmid,lckmax,lcsgrad,lcsmax,lcsmid,ikbase)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      IF(IUNIT(2).GT.0) CALL WEL5AL(ISUM,LENX,LCWELL,MXWELL,NWELLS,               AT 06C8
     1                 IUNIT(2),IOUT,IWELCB,NWELVL,IWELAL,IFREFM)
      IF(IUNIT(3).GT.0) CALL DRN5AL(ISUM,LENX,LCDRAI,NDRAIN,MXDRN,                AT 0728
     1                 IUNIT(3),IOUT,IDRNCB,NDRNVL,IDRNAL,IFREFM)
      IF(IUNIT(4).GT.0) CALL RIV5AL(ISUM,LENX,LCRIVR,MXRIVR,NRIVER,               AT 0788
     1            IUNIT(4),IOUT,IRIVCB,NRIVVL,IRIVAL,IFREFM)
      IF(IUNIT(5).GT.0) CALL EVT5AL(ISUM,LENX,LCIEVT,LCEVTR,LCEXDP,               AT 07E8
     1            LCSURF,NCOL,NROW,NEVTOP,IUNIT(5),IOUT,IEVTCB,IFREFM)
      IF(IUNIT(6).GT.0) CALL TLK1AL(ISUM,LENX,NCOL,NROW,NLAY,                     AT 0856
     1          LCRAT,LCZCB,LCA1,LCB1,LCALPH,LCBET,LCRM1,LCRM2,LCRM3,
     2          LCRM4,LCTL,LCTLK,LCSLU,LCSLD,NODES1,NM1,NM2,NUMC,
     3          NTM1,ITLKSV,ITLKRS,ITLKCB,ISS,IUNIT(6),IOUT)
      IF(IUNIT(7).GT.0) CALL GHB5AL(ISUM,LENX,LCBNDS,NBOUND,MXBND,                AT 093B
     1            IUNIT(7),IOUT,IGHBCB,NGHBVL,IGHBAL,IFREFM)
      IF(IUNIT(8).GT.0) CALL RCH5AL(ISUM,LENX,LCIRCH,LCRECH,NRCHOP,               AT 099B
     1            NCOL,NROW,IUNIT(8),IOUT,IRCHCB,IFREFM)
      IF(IUNIT(9).GT.0) CALL SIP5AL(ISUM,LENX,LCEL,LCFL,LCGL,LCV,                 AT 09FB
     1          LCHDCG,LCLRCH,LCW,MXITER,NPARM,NCOL,NROW,NLAY,
     2          IUNIT(9),IOUT,IFREFM)
      IF(IUNIT(10).GT.0) CALL DE45AL(ISUM,LENX,LCAU,LCAL,LCIUPP,                  AT 0A85
     1           LCIEQP,LCD4B,LCLRCH,LCHDCG,
     2           MXUP,MXLOW,MXEQ,MXBW,IUNIT(10),ITMX,ID4DIR,
     3           NCOL,NROW,NLAY,IOUT,ID4DIM)
      IF(IUNIT(11).GT.0) CALL SOR5AL(ISUM,LENX,LCA,LCRES,LCHDCG,LCLRCH,           AT 0B2B
     1       LCIEQP,MXITER,NCOL,NLAY,NSLICE,MBW,IUNIT(11),IOUT,IFREFM)
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
c      IF(IUNIT(13).GT.0) CALL PCG2AL(ISUM,LENX,LCV,LCSS,LCP,LCCD,
c     1       LCHCHG,LCLHCH,LCRCHG,LCLRCH,MXITER,ITER1,NCOL,NROW,NLAY,
c     2       IUNIT(13),IOUT,NPCOND,LCIT1)
      IF(IUNIT(13).GT.0) CALL PCG2AL(ISUM,LENX,LCV,LCSS,LCP,LCCD,                 AT 0BA7
     1       LCHCHG,LCLHCH,LCRCHG,LCLRCH,MXITER,ITER1,NCOL,NROW,NLAY,
     2       IUNIT(13),IOUT,NPCOND,LCIT1,noutc,ipcgdebug,iapart)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      IF(IUNIT(14).GT.0) CALL GFD1AL(ISUM,LENX,LCSC1,LCCDTR,LCCDTC,               AT 0C54
     1      LCBOT,LCTOP,LCSC2,IUNIT(14),ISS,NCOL,NROW,NLAY,IOUT,IGFDCB)
      IF(IUNIT(16).GT.0) CALL HFB1AL(ISUM,LENX,LCHFBR,NHFB,IUNIT(16),      *HFB*  AT 0CD0
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     1           IOUT)                                                     *HFB*
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
      IF(IUNIT(17).GT.0) CALL RES1AL(ISUM,LENX,LCIRES,LCIRSL,LCBRES,              AT 0D0D
     1  LCCRES,LCBBRE,LCHRES,LCHRSE,IUNIT(17),IOUT,NRES,IRESCB,
     2  NRESOP,IRESPT,NPTS,NCOL,NROW)
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
c      IF(IUNIT(18).GT.0) CALL STR1AL(ISUM,LENX,LCSTRM,ICSTRM,MXSTRM,    STR1
c     1                NSTREM,IUNIT(18),IOUT,ISTCB1,ISTCB2,NSS,NTRIB,    STR1
c     2                 NDIV,ICALC,CONST,LCTBAR,LCTRIB,LCIVAR,LCFGAR)    STR1
      IF(IUNIT(18).GT.0) CALL STR1AL(ISUM,LENX,LCSTRM,ICSTRM,MXSTRM,    STR1      AT 0D9E
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     1                NSTREM,IUNIT(18),IOUT,ISTCB1,ISTCB2,NSS,NTRIB,    STR1
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     2                 NDIV,ICALC,CONST,LCTBAR,LCTRIB,LCIVAR,LCFGAR,    STR1
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     3                 iswabs,lctba2)                                   STR1
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      IF (IUNIT(19).GT.0) CALL IBS1AL(ISUM,LENX,LCHC,LCSCE,LCSCV,       IBS       AT 0E44
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
```

```
     1            LCSUB,NCOL,NROW,NLAY,IIBSCB,IIBSOC,ISS,IUNIT(19),IOUT) IBS
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
      IF(IUNIT(20).GT.0) CALL CHD1AL(ISUM,LENX,LCCHDS,NCHDS,MXCHD,      CHD       AT 0EB9
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     1            IUNIT(20),IOUT)                                       CHD
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
      IF(IUNIT(21).GT.0) CALL FHB1AL(ISUM,LENX,LCFLLC,LCBDTM,LCFLRT,              AT 0EFD
     1           LCBDFV,LCBDHV,LCHDLC,LCSBHD,NBDTIM,NFLW,NHED,IUNIT(21),
     2           IOUT,IFHBCB,NFHBX1,NFHBX2,IFHBD3,IFHBD4,IFHBD5,
     3           IFHBSS,ISS)
C
C5------IF THE "X" ARRAY IS NOT BIG ENOUGH THEN STOP.
      IF(ISUM-1.GT.LENX) STOP                                                     AT 0FAA
C
C6------READ AND PREPARE INFORMATION FOR ENTIRE SIMULATION.
      CALL BAS5RP(X(LCIBOU),X(LCHNEW),X(LCSTRT),X(LCHOLD),                        AT 0FD0
     1        ISTRT,INBAS,HEADNG,NCOL,NROW,NLAY,VBVL,X(LCIOFL),
     2        IUNIT(12),IHEDFM,IDDNFM,IHEDUN,IDDNUN,IOUT,IPEROC,ITSOC,
     3        CHEDFM,CDDNFM,IBDOPT,IXSEC,LBHDSV,LBDDSV,IFREFM)
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
c      IF(IUNIT(1).GT.0) CALL BCF5RP(X(LCIBOU),X(LCHNEW),X(LCSC1),
c     1            X(LCHY),X(LCCR),X(LCCC),X(LCCV),X(LCDELR),
c     2       X(LCDELC),X(LCBOT),X(LCTOP),X(LCSC2),X(LCTRPY),IUNIT(1),
c     3       ISS,NCOL,NROW,NLAY,IOUT,X(LCWETD),IWDFLG,X(LCCVWD))
C-Environment Agency (UK)-16.K gradient independant of kbase>>>>>>>>>>>>
c      IF(IUNIT(1).GT.0) CALL BCF5RP(X(LCIBOU),X(LCHNEW),X(LCSC1),
c     1            X(LCHY),X(LCCR),X(LCCC),X(LCCV),X(LCDELR),
c     2       X(LCDELC),X(LCBOT),X(LCTOP),X(LCSC2),X(LCTRPY),IUNIT(1),
c     3       ISS,NCOL,NROW,NLAY,IOUT,X(LCWETD),IWDFLG,X(LCCVWD),itrpy,
c     4       x(lckgrad),x(lcmid),x(lckmax),x(lcsgrad),x(lcsmax),x(lcsmid))
      IF(IUNIT(1).GT.0) CALL BCF5RP(X(LCIBOU),X(LCHNEW),X(LCSC1),                 AT 10CA
     1           X(LCHY),X(LCCR),X(LCCC),X(LCCV),X(LCDELR),
     2      X(LCDELC),X(LCBOT),X(LCTOP),X(LCSC2),X(LCTRPY),IUNIT(1),
     3      ISS,NCOL,NROW,NLAY,IOUT,X(LCWETD),IWDFLG,X(LCCVWD),itrpy,
     4      x(lckgrad),x(lcmid),x(lckmax),x(lcsgrad),x(lcsmax),x(lcsmid),
     5      ikbase)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      IF(IUNIT(6).GT.0) CALL TLK1RP(X(LCRAT),X(LCZCB),X(LCA1),X(LCB1),            AT 1244
     1           X(LCALPH),X(LCBET),X(LCRM1),X(LCRM2),X(LCRM3),X(LCRM4),
     2           NODES1,NM1,NM2,NUMC,NTM1,ITLKRS,DELTM1,X(LCBUFF),
     3           X(LCDELC),X(LCDELR),TLKTIM,NROW,NCOL,IUNIT(6),IOUT)
      IF(IUNIT(9).GT.0) CALL SIP5RP(NPARM,MXITER,ACCL,HCLOSE,X(LCW),              AT 1363
     1           IUNIT(9),IPCALC,IPRSIP,IOUT,IFREFM)
      IF(IUNIT(10).GT.0) CALL DE45RP(IUNIT(10),MXITER,NITER,ITMX,                 AT 13C5
     1             ACCL,HCLOSE,IFREQ,IPRD4,IOUT,MUTD4)
      IF(IUNIT(11).GT.0) CALL SOR5RP(MXITER,ACCL,HCLOSE,IUNIT(11),                AT 1420
     1          IPRSOR,IOUT,IFREFM)
      IF(IUNIT(13).GT.0) CALL PCG2RP(MXITER,ITER1,HCLOSE,RCLOSE,                  AT 1466
     1          NPCOND,NBPOL,RELAX,IPRPCG,IUNIT(13),IOUT,MUTPCG,
     2          NITER,X(LCIT1),DAMP)
      IF(IUNIT(14).GT.0) CALL GFD1RP(X(LCIBOU),X(LCHNEW),X(LCSC1),                AT 14E4
     1           X(LCCDTR),X(LCCDTC),X(LCCR),X(LCCC),X(LCCV),X(LCDELR),
     2           X(LCDELC),X(LCBOT),X(LCTOP),X(LCSC2),
     3           IUNIT(14),ISS,NCOL,NROW,NLAY,NODES,IOUT)
      IF(IUNIT(16).GT.0) CALL HFB1RP(X(LCCR),X(LCCC),X(LCDELR),          *HFB*  AT 15E0
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     1          X(LCDELC),X(LCHFBR),IUNIT(16),NCOL,NROW,NLAY,NODES,      *HFB*
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     1          NHFB,IOUT)                                               *HFB*
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
      IF(IUNIT(19).GT.0) CALL IBS1RP(X(LCDELR),X(LCDELC),X(LCHNEW),     IBS       AT 166C
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     1      X(LCHC),X(LCSCE),X(LCSCV),X(LCSUB),NCOL,NROW,NLAY,          IBS
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     2      NODES,IIBSOC,ISUBFM,ICOMFM,IHCFM,ISUBUN,ICOMUN,IHCUN,       IBS
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     3      IUNIT(19),IOUT)                                             IBS
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
      IF(IUNIT(21).GT.0) CALL FHB1RP(X(LCIBOU),NROW,NCOL,NLAY,                    AT 173E
     &           X(LCFLLC),X(LCBDTM),NBDTIM,X(LCFLRT),NFLW,NHED,
     &           X(LCHDLC),X(LCSBHD),IUNIT(21),IOUT,
     &           NFHBX1,NFHBX2,IFHBD3,IFHBD5)
C
C-Environment Agency (UK)-10.Convergence info output to screen>>>>>>>>>>
      IF(IUNIT(13).GT.0)then                                                      AT 17FB
        write(*,5)' +--------+------+-------+------------+------------+'          AT 1808
      endif                                                                       AT 1848
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C7------SIMULATE EACH STRESS PERIOD.
      DO 300 KPER=1,NPER                                                          AT 1848
```

```
      KKPER=KPER                                                         AT 1876
C
C-Environment Agency (UK)-10.Convergence info output to screen>>>>>>>>>>
      IF(IUNIT(13).GT.0)then                                             AT 187C
        write(*,5)' | Stress | Time | Iter-  |  Max Head  |  Max Flow  |'   AT 1889
        write(*,5)' | Period | Step | ation  |   Change   |  Residual  |'   AT 18C9
        write(*,5)' +--------+------+-------+------------+------------+'   AT 1909
      endif                                                              AT 1949
 5    format(a52)                                                        AT 1949
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C7A-----READ STRESS PERIOD TIMING INFORMATION.
      CALL BAS5ST(NSTP,DELT,TSMULT,PERTIM,KKPER,INBAS,IOUT,IFREFM)       AT 195E
C
C7B-----READ AND PREPARE INFORMATION FOR STRESS PERIOD.
      IF(IUNIT(2).GT.0) CALL WEL5RP(X(LCWELL),NWELLS,MXWELL,IUNIT(2),    AT 199E
     1             IOUT,NWELVL,IWELAL,IFREFM)
      IF(IUNIT(3).GT.0) CALL DRN5RP(X(LCDRAI),NDRAIN,MXDRN,IUNIT(3),     AT 19F2
     1                 IOUT,NDRNVL,IDRNAL,IFREFM)
      IF(IUNIT(4).GT.0) CALL RIV5RP(X(LCRIVR),NRIVER,MXRIVR,IUNIT(4),    AT 1A46
     1            IOUT,NRIVVL,IRIVAL,IFREFM)
      IF(IUNIT(5).GT.0) CALL EVT5RP(NEVTOP,X(LCIEVT),X(LCEVTR),          AT 1A9A
     1            X(LCEXDP),X(LCSURF),X(LCDELR),X(LCDELC),NCOL,NROW,
     1            IUNIT(5),IOUT,IFREFM)
      IF(IUNIT(7).GT.0) CALL GHB5RP(X(LCBNDS),NBOUND,MXBND,IUNIT(7),     AT 1B2D
     1              IOUT,NGHBVL,IGHBAL,IFREFM)
      IF(IUNIT(8).GT.0) CALL RCH5RP(NRCHOP,X(LCIRCH),X(LCRECH),          AT 1B81
     1            X(LCDELR),X(LCDELC),NROW,NCOL,IUNIT(8),IOUT,IFREFM)
      IF(IUNIT(17).GT.0) CALL RES1RP(X(LCIRES),X(LCIRSL),X(LCBRES),      AT 1BF8
     1  X(LCCRES),X(LCBBRE),X(LCHRSE),X(LCIBOU),X(LCDELR),X(LCDELC),
     2  NRES,NRESOP,NPTS,NCOL,NROW,NLAY,PERLEN,DELT,NSTP,TSMULT,
     3  IUNIT(17),IOUT)
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
c      IF(IUNIT(18).GT.0) CALL STR1RP(X(LCSTRM),X(ICSTRM),NSTREM,        STR1
c     1                     MXSTRM,IUNIT(18),IOUT,X(LCTBAR),NDIV,NSS,    STR1
c     2                     NTRIB,X(LCIVAR),ICALC,IPTFLG)                STR1
      IF(IUNIT(18).GT.0) CALL STR1RP(X(LCSTRM),X(ICSTRM),NSTREM,        STR1      AT 1CDF
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     1                     MXSTRM,IUNIT(18),IOUT,X(LCTBAR),NDIV,NSS,    STR1
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     2                     NTRIB,X(LCIVAR),ICALC,IPTFLG,iswabs,         STR1
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     3                     x(lctba2))                                   STR1
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      IF(IUNIT(20).GT.0) CALL CHD1RP(X(LCCHDS),NCHDS,MXCHD,X(LCIBOU),   CHD       AT 1D80
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     1            NCOL,NROW,NLAY,PERLEN,DELT,NSTP,TSMULT,IUNIT(20),IOUT)CHD
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
C
C7C-----SIMULATE EACH TIME STEP.
      DO 200 KSTP=1,NSTP                                                 AT 1DFE
      KKSTP=KSTP                                                         AT 1E2C
C
C7C1----CALCULATE TIME STEP LENGTH. SET HOLD=HNEW..
      CALL BAS5AD(DELT,TSMULT,TOTIM,PERTIM,X(LCHNEW),X(LCHOLD),KKSTP,    AT 1E32
     1             NCOL,NROW,NLAY)
      IF(IUNIT(6).GT.0) CALL TLK1AD(X(LCRAT),X(LCZCB),X(LCA1),X(LCB1),   AT 1E8E
     1          X(LCALPH),X(LCBET),X(LCRM1),X(LCRM2),X(LCRM3),X(LCRM4),
     2          X(LCTL),X(LCTLK),X(LCSLU),X(LCSLD),NM1,NM2,NUMC,NTM1,
     3          DELTM1,X(LCHNEW),X(LCIBOU),X(LCTOP),
     4          NROW,NCOL,NLAY,DELT,TLKTIM,IUNIT(6),IOUT)
      IF(IUNIT(20).GT.0) CALL CHD1FM(NCHDS,MXCHD,X(LCCHDS),X(LCIBOU),   CHD       AT 1FE5
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     1          X(LCHNEW),X(LCHOLD),PERLEN,PERTIM,DELT,NCOL,NROW,NLAY)  CHD
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
      IF(IUNIT(1).GT.0) CALL BCF5AD(X(LCIBOU),X(LCHOLD),X(LCBOT),        AT 206A
     1             X(LCWETD),IWDFLG,ISS,NCOL,NROW,NLAY)
      IF(IUNIT(17).GT.0) CALL RES1AD(X(LCHRES),X(LCHRSE),X(LCIRES),      AT 20DA
     1 X(LCBRES),X(LCDELR),X(LCDELC),NRES,IRESPT,NCOL,NROW,
     1      PERLEN,PERTIM,TOTIM,KKSTP,KKPER,IOUT)
      IF(IUNIT(21).GT.0) CALL FHB1AD(X(LCHNEW),X(LCHOLD),NCOL,NROW,NLAY, AT 2189
     &          ISS,TOTIM,DELT,X(LCBDTM),NBDTIM,X(LCFLRT),
     &          X(LCBDFV),X(LCBDHV),NFLW,X(LCSBHD),X(LCHDLC),NHED,
     &          NFHBX1,NFHBX2,IFHBD3,IFHBD4,IFHBD5,IFHBSS)
C
C-Environment Agency (UK)-11.Force converge if NOUTC outer itrs conv>>>>
      nout=0                                                             AT 2277
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
c
C7C2----ITERATIVELY FORMULATE AND SOLVE THE EQUATIONS.
      DO 100 KITER=1,MXITER                                              AT 2281
```

```
      KKITER=KITER                                                      AT 22AF
C
C7C2A---FORMULATE THE FINITE DIFFERENCE EQUATIONS.
      CALL BAS5FM(X(LCHCOF),X(LCRHS),NODES)                             AT 22B5
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
c      IF(IUNIT(1).GT.0) CALL BCF5FM(X(LCHCOF),X(LCRHS),X(LCHOLD),
c     1          X(LCSC1),X(LCHNEW),X(LCIBOU),X(LCCR),X(LCCC),X(LCCV),
c     2          X(LCHY),X(LCTRPY),X(LCBOT),X(LCTOP),X(LCSC2),
c     3          X(LCDELR),X(LCDELC),DELT,ISS,KKITER,KKSTP,KKPER,NCOL,
c     4          NROW,NLAY,IOUT,X(LCWETD),IWDFLG,X(LCCVWD),WETFCT,
c     5          IWETIT,IHDWET,HDRY,X(LCBUFF))
      IF(IUNIT(1).GT.0) CALL BCF5FM(X(LCHCOF),X(LCRHS),X(LCHOLD),       AT 22E0
     1          X(LCSC1),X(LCHNEW),X(LCIBOU),X(LCCR),X(LCCC),X(LCCV),
     2          X(LCHY),X(LCTRPY),X(LCBOT),X(LCTOP),X(LCSC2),
     3          X(LCDELR),X(LCDELC),DELT,ISS,KKITER,KKSTP,KKPER,NCOL,
     4          NROW,NLAY,IOUT,X(LCWETD),IWDFLG,X(LCCVWD),WETFCT,
     5          IWETIT,IHDWET,HDRY,X(LCBUFF),itrpy,x(lckgrad),ihold,
     6          x(lcmid),x(lckmax),x(lcsgrad),x(lcsmax),x(lcsmid))
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      IF(IUNIT(14).GT.0) CALL GFD1FM(X(LCHCOF),X(LCRHS),X(LCHOLD),      AT 24C6
     1          X(LCSC1),X(LCHNEW),X(LCIBOU),X(LCCR),X(LCCC),X(LCCV),
     2          X(LCCDTR),X(LCCDTC),X(LCBOT),X(LCTOP),X(LCSC2),
     3          DELT,ISS,KKITER,KKSTP,KKPER,NCOL,NROW,NLAY,IOUT)
      IF(IUNIT(16).GT.0) CALL HFB1FM(X(LCHNEW),X(LCCR),X(LCCC),      *HFB*  AT 25DE
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     1          X(LCBOT),X(LCTOP),X(LCDELR),X(LCDELC),X(LCHFBR),     *HFB*
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     2          NCOL,NROW,NLAY,NHFB)                                 *HFB*
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
      IF(IUNIT(6).GT.0) CALL TLK1FM(X(LCRAT),X(LCTL),X(LCTLK),X(LCSLU),  AT 267F
     1          X(LCSLD),NUMC,X(LCHNEW),X(LCIBOU),X(LCTOP),X(LCCV),
     2          X(LCHCOF),X(LCRHS),NROW,NCOL,NLAY)
      IF(IUNIT(2).GT.0) CALL WEL5FM(NWELLS,MXWELL,X(LCRHS),X(LCWELL),   AT 274A
     1           X(LCIBOU),NCOL,NROW,NLAY,NWELVL)
      IF(IUNIT(3).GT.0) CALL DRN5FM(NDRAIN,MXDRN,X(LCDRAI),X(LCHNEW),   AT 27B3
     1         X(LCHCOF),X(LCRHS),X(LCIBOU),NCOL,NROW,NLAY,NDRNVL)
      IF(IUNIT(4).GT.0) CALL RIV5FM(NRIVER,MXRIVR,X(LCRIVR),X(LCHNEW),  AT 2838
     1            X(LCHCOF),X(LCRHS),X(LCIBOU),NCOL,NROW,NLAY,NRIVVL)
      IF(IUNIT(5).GT.0) CALL EVT5FM(NEVTOP,X(LCIEVT),X(LCEVTR),         AT 28BD
     1            X(LCEXDP),X(LCSURF),X(LCRHS),X(LCHCOF),X(LCIBOU),
     1            X(LCHNEW),NCOL,NROW,NLAY)
      IF(IUNIT(7).GT.0) CALL GHB5FM(NBOUND,MXBND,X(LCBNDS),X(LCHCOF),   AT 295E
     1            X(LCRHS),X(LCIBOU),NCOL,NROW,NLAY,NGHBVL)
      IF(IUNIT(8).GT.0) CALL RCH5FM(NRCHOP,X(LCIRCH),X(LCRECH),         AT 29D5
     1            X(LCRHS),X(LCIBOU),NCOL,NROW,NLAY)
      IF(IUNIT(17).GT.0) CALL RES1FM(X(LCIRES),X(LCIRSL),X(LCBRES),     AT 2A3E
     1   X(LCCRES),X(LCBBRE),X(LCHRES),X(LCIBOU),X(LCHNEW),X(LCHCOF),
     2   X(LCRHS),NRES,NRESOP,NCOL,NROW,NLAY)
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
c      IF(IUNIT(18).GT.0) CALL STR1FM(NSTREM,X(LCSTRM),X(ICSTRM),      STR1
c     1                      X(LCHNEW),X(LCHCOF),X(LCRHS),X(LCIBOU),    STR1
c     2                MXSTRM,NCOL,NROW,NLAY,IOUT,NSS,X(LCTBAR),        STR1
c     3                NTRIB,X(LCTRIB),X(LCIVAR),X(LCFGAR),ICALC,CONST)  STR1
      IF(IUNIT(18).GT.0) CALL STR1FM(NSTREM,X(LCSTRM),X(ICSTRM),      STR1      AT 2B02
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     1                      X(LCHNEW),X(LCHCOF),X(LCRHS),X(LCIBOU),    STR1
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     2                MXSTRM,NCOL,NROW,NLAY,IOUT,NSS,X(LCTBAR),        STR1
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     3                NTRIB,X(LCTRIB),X(LCIVAR),X(LCFGAR),ICALC,CONST,  STR1
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     4                iswabs,x(lctba2))                                STR1
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      IF(IUNIT(19).GT.0) CALL IBS1FM(X(LCRHS),X(LCHCOF),X(LCHNEW),    IBS       AT 2BFE
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     1      X(LCHOLD),X(LCHC),X(LCSCE),X(LCSCV),X(LCIBOU),             IBS
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     2      NCOL,NROW,NLAY,DELT)                                       IBS
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
      IF(IUNIT(21).GT.0) CALL FHB1FM(X(LCRHS),X(LCIBOU),X(LCFLLC),      AT 2C9F
     1 X(LCBDFV),NFLW,NCOL,NROW,NLAY,IFHBD4)
C
C7C2B---MAKE ONE CUT AT AN APPROXIMATE SOLUTION.
      IF(IUNIT(9).GT.0) CALL SIP5AP(X(LCHNEW),X(LCIBOU),X(LCCR),X(LCCC),  AT 2D0F
     1     X(LCCV),X(LCHCOF),X(LCRHS),X(LCEL),X(LCFL),X(LCGL),X(LCV),
     2     X(LCW),X(LCHDCG),X(LCLRCH),NPARM,KKITER,HCLOSE,ACCL,ICNVG,
     3     KKSTP,KKPER,IPCALC,IPRSIP,MXITER,NSTP,NCOL,NROW,NLAY,NODES,
     4     IOUT)
      IF(IUNIT(10).GT.0) CALL DE45AP(X(LCHNEW),X(LCIBOU),X(LCAU),       AT 2E58
     1  X(LCAL),X(LCIUPP),X(LCIEQP),X(LCD4B),MXUP,MXLOW,MXEQ,MXBW,
     2  X(LCCR),X(LCCC),X(LCCV),X(LCHCOF),X(LCRHS),ACCL,KKITER,ITMX,
```

```
     3  MXITER,NITER,HCLOSE,IPRD4,ICNVG,NCOL,NROW,NLAY,IOUT,X(LCLRCH),
     4  X(LCHDCG),IFREQ,KKSTP,KKPER,DELT,NSTP,ID4DIR,ID4DIM,MUTD4)
      IF(IUNIT(11).GT.0) CALL SOR5AP(X(LCHNEW),X(LCIBOU),X(LCCR),          AT 2FDC
     1     X(LCCC),X(LCCV),X(LCHCOF),X(LCRHS),X(LCA),X(LCRES),X(LCIEQP),
     2     X(LCHDCG),X(LCLRCH),KKITER,HCLOSE,ACCL,ICNVG,KKSTP,KKPER,
     3     IPRSOR,MXITER,NSTP,NCOL,NROW,NLAY,NSLICE,MBW,IOUT)
C-Environment Agency (UK)-0.General change to enable new options>>>>>>>>
c      IF(IUNIT(13).GT.0) CALL PCG2AP(X(LCHNEW),X(LCIBOU),X(LCCR),
c     1      X(LCCC),X(LCCV),X(LCHCOF),X(LCRHS),X(LCV),X(LCSS),X(LCP),
c     2      X(LCCD),X(LCHCHG),X(LCLHCH),X(LCRCHG),X(LCLRCH),KKITER,
c     3      NITER,HCLOSE,RCLOSE,ICNVG,KKSTP,KKPER,IPRPCG,MXITER,ITER1,
c     4      NPCOND,NBPOL,NSTP,NCOL,NROW,NLAY,NODES,RELAX,IOUT,MUTPCG,
c     5      0,0,SN,SP,SR,X(LCIT1),DAMP)
      IF(IUNIT(13).GT.0) CALL PCG2AP(X(LCHNEW),X(LCIBOU),X(LCCR),          AT 3102
     1      X(LCCC),X(LCCV),X(LCHCOF),X(LCRHS),X(LCV),X(LCSS),X(LCP),
     2      X(LCCD),X(LCHCHG),X(LCLHCH),X(LCRCHG),X(LCLRCH),KKITER,
     3      NITER,HCLOSE,RCLOSE,ICNVG,KKSTP,KKPER,IPRPCG,MXITER,ITER1,
     4      NPCOND,NBPOL,NSTP,NCOL,NROW,NLAY,NODES,RELAX,IOUT,MUTPCG,
     5      0,0,SN,SP,SR,X(LCIT1),DAMP,noutc,nout,ipcgdebug,ixsec,
     6      chedfm,x(lcbuff),lbhdsv)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C-Environment Agency (UK)-10.Convergence info output to screen>>>>>>>>>>
      IF(IUNIT(13).GT.0.and.NITER.gt.0)then                                AT 32EA
        write(*,501)KKPER,KKSTP,NITER,X(LCHCHG+NITER-1),                   AT 3304
     1      X(LCRCHG+NITER-1)
      endif                                                                AT 33B6
 501  format(' |',i7,' |',i5,' |',i6,' |',g11.4,' |',g11.4,' |')           AT 33B6
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C7C2C---IF CONVERGENCE CRITERION HAS BEEN MET STOP ITERATING.
      IF(ICNVG.EQ.1) GO TO 110                                             AT 340B
  100 CONTINUE                                                             AT 3418
      KITER=MXITER                                                         AT 341D
  110 CONTINUE                                                             AT 3429
C
C7C3----DETERMINE WHICH OUTPUT IS NEEDED.
      CALL BAS5OC(NSTP,KKSTP,ICNVG,X(LCIOFL),NLAY,IBUDFL,ICBCFL,           AT 3429
     1   IHDDFL,IUNIT(12),IOUT,KKPER,IPEROC,ITSOC,IBDOPT,IXSEC,IFREFM)
C
C7C4----CALCULATE BUDGET TERMS. SAVE CELL-BY-CELL FLOW TERMS.
      MSUM=1                                                               AT 34A8
      IF(IUNIT(6).GT.0) CALL TLK1BD(X(LCRAT),X(LCTL),X(LCTLK),             AT 34B2
     1          X(LCSLU),X(LCSLD),NUMC,ITLKCB,X(LCHNEW),X(LCBUFF),
     2          X(LCIBOU),X(LCTOP),X(LCCV),VBNM,VBVL,MSUM,NCOL,NROW,
     3          NLAY,DELT,KSTP,KPER,ICBCFL,IOUT)
C7C4A---THE ORIGINAL BCF BUDGET MODULE HAS BEEN REPLACED BY THREE
C7C4A---SUBMODULES: SBCF5S, SBCF5F, AND SBCF5B .
      IF(IUNIT(1).GT.0) THEN                                               AT 35B4
C-Environment Agency (UK)-2.Variable storage with depth (VSD)>>>>>>>>>>>
c         CALL SBCF5S(VBNM,VBVL,MSUM,X(LCHNEW),X(LCIBOU),X(LCHOLD),
c     1      X(LCSC1),X(LCTOP),X(LCSC2),DELT,ISS,NCOL,NROW,NLAY,KKSTP,
c     2      KKPER,IBCFCB,ICBCFL,X(LCBUFF),IOUT,PERTIM,TOTIM)
         CALL SBCF5S(VBNM,VBVL,MSUM,X(LCHNEW),X(LCIBOU),X(LCHOLD),         AT 35C1
     1     X(LCSC1),X(LCTOP),X(LCSC2),DELT,ISS,NCOL,NROW,NLAY,KKSTP,
     2     KKPER,IBCFCB,ICBCFL,X(LCBUFF),IOUT,PERTIM,TOTIM,x(lcbot),
     3     x(lcmid),x(lcsgrad),x(lcsmax),x(lcsmid))
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
         CALL SBCF5F(VBNM,VBVL,MSUM,X(LCHNEW),X(LCIBOU),X(LCCR),           AT 36E0
     1     X(LCCC),X(LCCV),X(LCTOP),DELT,NCOL,NROW,NLAY,KKSTP,KKPER,
     2     IBCFCB,X(LCBUFF),IOUT,ICBCFL,PERTIM,TOTIM,ICHFLG)
         IBDRET=0                                                          AT 37B9
         IC1=1                                                             AT 37C3
         IC2=NCOL                                                          AT 37CD
         IR1=1                                                             AT 37D9
         IR2=NROW                                                          AT 37E3
         IL1=1                                                             AT 37EF
         IL2=NLAY                                                          AT 37F9
         DO 155 IDIR=1,3                                                   AT 3805
         CALL SBCF5B(X(LCHNEW),X(LCIBOU),X(LCCR),X(LCCC),X(LCCV),          AT 3822
     1       X(LCTOP),NCOL,NROW,NLAY,KKSTP,KKPER,IBCFCB,X(LCBUFF),
     2       IOUT,ICBCFL,DELT,PERTIM,TOTIM,IDIR,IBDRET,ICHFLG,
     3       IC1,IC2,IR1,IR2,IL1,IL2)
155      CONTINUE                                                          AT 3918
      END IF                                                               AT 391D
      IF(IUNIT(14).GT.0) CALL GFD1BD(VBNM,VBVL,MSUM,X(LCHNEW),             AT 391D
     1     X(LCIBOU),X(LCHOLD),X(LCSC1),X(LCCR),X(LCCC),X(LCCV),
     2     X(LCTOP),X(LCSC2),DELT,ISS,NCOL,NROW,NLAY,KKSTP,KKPER,
     3     IGFDCB,ICBCFL,X(LCBUFF),IOUT)
      IF(IUNIT(2).GT.0) CALL WEL5BD(NWELLS,MXWELL,VBNM,VBVL,MSUM,          AT 3A1F
     1     X(LCWELL),X(LCIBOU),DELT,NCOL,NROW,NLAY,KKSTP,KKPER,IWELCB,
     1     ICBCFL,X(LCBUFF),IOUT,PERTIM,TOTIM,NWELVL,IWELAL)
```

```
      IF(IUNIT(3).GT.0) CALL DRN5BD(NDRAIN,MXDRN,VBNM,VBVL,MSUM,                AT 3AE2
     1     X(LCDRAI),DELT,X(LCHNEW),NCOL,NROW,NLAY,X(LCIBOU),KKSTP,
     2     KKPER,IDRNCB,ICBCFL,X(LCBUFF),IOUT,PERTIM,TOTIM,NDRNVL,
     3     IDRNAL)
      IF(IUNIT(4).GT.0) CALL RIV5BD(NRIVER,MXRIVR,X(LCRIVR),X(LCIBOU),          AT 3BB3
     1     X(LCHNEW),NCOL,NROW,NLAY,DELT,VBVL,VBNM,MSUM,KKSTP,KKPER,
     2     IRIVCB,ICBCFL,X(LCBUFF),IOUT,PERTIM,TOTIM,NRIVVL,IRIVAL)
      IF(IUNIT(5).GT.0) CALL EVT5BD(NEVTOP,X(LCIEVT),X(LCEVTR),                 AT 3C84
     1     X(LCEXDP),X(LCSURF),X(LCIBOU),X(LCHNEW),NCOL,NROW,NLAY,
     2     DELT,VBVL,VBNM,MSUM,KKSTP,KKPER,IEVTCB,ICBCFL,X(LCBUFF),IOUT,
     3     PERTIM,TOTIM)
      IF(IUNIT(7).GT.0) CALL GHB5BD(NBOUND,MXBND,VBNM,VBVL,MSUM,                AT 3D6A
     1     X(LCBNDS),DELT,X(LCHNEW),NCOL,NROW,NLAY,X(LCIBOU),KKSTP,
     2     KKPER,IGHBCB,ICBCFL,X(LCBUFF),IOUT,PERTIM,TOTIM,NGHBVL,
     3     IGHBAL)
      IF(IUNIT(8).GT.0) CALL RCH5BD(NRCHOP,X(LCIRCH),X(LCRECH),                 AT 3E3B
     1     X(LCIBOU),NROW,NCOL,NLAY,DELT,VBVL,VBNM,MSUM,KKSTP,KKPER,
     2     IRCHCB,ICBCFL,X(LCBUFF),IOUT,PERTIM,TOTIM)
      IF(IUNIT(17).GT.0) CALL RES1BD(X(LCIRES),X(LCIRSL),X(LCBRES),             AT 3EF7
     1      X(LCCRES),X(LCBBRE),X(LCHRES),X(LCIBOU),X(LCHNEW),
     2      X(LCBUFF),VBVL,VBNM,MSUM,KSTP,KPER,NRES,NRESOP,
     3      NCOL,NROW,NLAY,DELT,IRESCB,ICBCFL,IOUT)
C-Environment Agency (UK)-14.Allow in/outflow or trib at any reach>>>>>>
c      IF(IUNIT(18).GT.0) CALL STR1BD(NSTREM,X(LCSTRM),X(ICSTRM),       STR1
c     1  X(LCIBOU),MXSTRM,X(LCHNEW),NCOL,NROW,NLAY,DELT,VBVL,VBNM,MSUM, STR1
c     2  KKSTP,KKPER,ISTCB1,ISTCB2,ICBCFL,X(LCBUFF),IOUT,NTRIB,NSS,     STR1
c     3  X(LCTRIB),X(LCTBAR),X(LCIVAR),X(LCFGAR),ICALC,CONST,IPTFLG)    STR1
      IF(IUNIT(18).GT.0) CALL STR1BD(NSTREM,X(LCSTRM),X(ICSTRM),       STR1      AT 3FF2
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     1  X(LCIBOU),MXSTRM,X(LCHNEW),NCOL,NROW,NLAY,DELT,VBVL,VBNM,MSUM, STR1
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     2  KKSTP,KKPER,ISTCB1,ISTCB2,ICBCFL,X(LCBUFF),IOUT,NTRIB,NSS,     STR1
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     3  X(LCTRIB),X(LCTBAR),X(LCIVAR),X(LCFGAR),ICALC,CONST,IPTFLG,    STR1
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     4  iswabs,x(lctba2))                                              STR1
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
      IF(IUNIT(19).GT.0) CALL IBS1BD(X(LCIBOU),X(LCHNEW),X(LCHOLD),    IBS       AT 412F
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     1      X(LCHC),X(LCSCE),X(LCSCV),X(LCSUB),X(LCDELR),X(LCDELC),    IBS
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     2      NCOL,NROW,NLAY,DELT,VBVL,VBNM,MSUM,KSTP,KPER,IIBSCB,       IBS
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     3      ICBCFL,X(LCBUFF),IOUT)                                     IBS
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
      IF(IUNIT(21).GT.0) CALL FHB1BD(X(LCFLLC),X(LCBDFV),NFLW,                  AT 422A
     1     VBNM,VBVL,MSUM,X(LCIBOU),DELT,NCOL,NROW,NLAY,KKSTP,KKPER,
     2     IFHBCB,ICBCFL,X(LCBUFF),IOUT,IFHBD4)
C
C-Environment Agency (UK)-8.Output calculated transmissivities>>>>>>>>>>
C7C4B--PRINT INTERNODAL TRANSMISSIVITIES IF ITRANS IS NOT ZERO
C-Environment Agency (UK)-17.Output calc trans to binary file>>>>>>>>>>>
c      if(itrans.ne.0)call bcf5ot(x(lccc),x(lccr),x(lcBUFF),kKSTP,
c     1     kKPER,NCOL,NROW,NLAY,IOUT,IXSEC,x(lcdelc),x(lcdelr),itrpy,
c     2     x(lctrpy),ihedfm,x(lccv))
      if(itrans.ne.0)call bcf5ot(x(lccc),x(lccr),x(lcBUFF),kKSTP,               AT 42DF
     1     kKPER,NCOL,NROW,NLAY,IOUT,IXSEC,x(lcdelc),x(lcdelr),itrpy,
     2     x(lctrpy),ihedfm,x(lccv),itrans,X(LCIOFL),CHEDFM,LBHDSV,
     3     X(LCIBOU),PERTIM,TOTIM)
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
c
C7C5---PRINT AND OR SAVE HEADS AND DRAWDOWNS. PRINT OVERALL BUDGET.
      CALL BAS5OT(X(LCHNEW),X(LCSTRT),ISTRT,X(LCBUFF),X(LCIOFL),                AT 43DA
     1     MSUM,X(LCIBOU),VBNM,VBVL,KKSTP,KKPER,DELT,PERTIM,TOTIM,
     2     ITMUNI,NCOL,NROW,NLAY,ICNVG,IHDDFL,IBUDFL,IHEDFM,IHEDUN,
     3     IDDNFM,IDDNUN,IOUT,CHEDFM,CDDNFM,IXSEC,LBHDSV,LBDDSV)
C
C7C5A--PRINT AND OR SAVE SUBSIDENCE, COMPACTION, AND CRITICAL HEAD.
      IF(IUNIT(19).GT.0) CALL IBS1OT(NCOL,NROW,NLAY,PERTIM,TOTIM,KSTP, IBS       AT 44F3
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     1      KPER,NSTP,X(LCBUFF),X(LCSUB),X(LCHC),IIBSOC,ISUBFM,ICOMFM, IBS
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
     2      IHCFM,ISUBUN,ICOMUN,IHCUN,IUNIT(19),IOUT)                  IBS
WARNING - Characters found after column 72 (use -NO_WARN73 to suppress)
C
C7C6----IF ITERATION FAILED TO CONVERGE THEN STOP.
C-Environment Agency (UK)-10.Convergence info output to screen>>>>>>>>>>
c      IF(ICNVG.EQ.0) STOP
      IF(ICNVG.EQ.0)then                                                        AT 45A9
        write(*,*)                                                              AT 45B6
```

```
        write(*,*)' *** Failed to converge ***'                                   AT 45D2
        STOP                                                                      AT 4606
      endif                                                                       AT 4619
      IF(IUNIT(13).GT.0)then                                                      AT 4619
        write(*,5)' +--------+------+-------+------------+------------+'          AT 4626
      endif                                                                       AT 4666
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
c
  200 CONTINUE                                                                    AT 4666
  300 CONTINUE                                                                    AT 466B
C
C7C7----WRITE RESTART RECORDS
C7C7A---WRITE RESTART RECORDS FOR TRANSIENT-LEAKAGE PACKAGE
      IF(IUNIT(6).GT.0) CALL TLK1OT(X(LCRM1),X(LCRM2),                            AT 4670
     1     X(LCRM3),X(LCRM4),NM1,NM2,ITLKSV,DELTM1,TLKTIM,IOUT)

C-Environment Agency (UK)-3.Auto-conversion for steady state VKD>>>>>>>>
C7D-----IF USING AUTOMATIC CONVERSION OPTION, CLOSE ALL CONNECTED FILES,
C7D-----(EXCEPT THE MAIN OUTPUT FILE), CALL ROUTINE TO WRITE NEW INPUT
C7D-----FILES AND RETURN TO THE START OF THE PROGRAM TO RUN THE SECOND
C7D-----PART OF THE SIMULATION WITH VKD ACTIVE.
      if(iss.eq.2.or.iss.eq.3)then                                                AT 46E7
         do 355 i=1,ibouts-1                                                      AT 4701
            inquire(unit=i,opened=exists)                                         AT 472D
            inquire(unit=i,named=exists2)                                         AT 474F
            if(exists.and.exists2)then                                            AT 4774
              if(i.ne.iout)close(i)                                               AT 478B
            endif                                                                 AT 47B6
  355    continue                                                                 AT 47B6
         call BCF5VK(x(lchnew),inunit,iunit(1),inbas,ibcfoutunit,                 AT 47BB
     1        inoutunit,ioutbas,ncol,nrow,nlay,ifrefm,fname,cunit(1),
     2        IXSEC,NPER,iout,x(lcIBOU),x(lcHOLD),x(lcTRPY),x(lcDELR),
     3        x(lcDELC),x(lcSC1),x(lcCC),x(lcHY),x(lcBOT),x(lcCV),
     4        x(lcTOP),x(lcSC2),x(lckgrad),x(lcWETD),x(lcbuff),x(lccr),
     5        x(lcmid),x(lckmax))
         do 350 i=1,ibouts-1                                                      AT 4936
            inquire(unit=i,opened=exists)                                         AT 4962
            inquire(unit=i,named=exists2)                                         AT 4984
            if(exists.and.exists2)then                                            AT 49A9
              close(i)                                                            AT 49C0
            endif                                                                 AT 49D9
  350    continue                                                                 AT 49D9
         go to 55                                                                 AT 49DE
      end if                                                                      AT 49E3
C-Environment Agency (UK)-End of this section of modified code <<<<<<<<<
C
C8------END OF SIMULATION
      IF(IBATCH.GT.0) THEN                                                        AT 49E3
         WRITE(IBOUTS,*) ' Normal termination of simulation.'                     AT 49ED
         DO 400 I=1,IBOUTS-1                                                      AT 4A20
            INQUIRE(UNIT=I,OPENED=EXISTS)                                         AT 4A4C
            IF(EXISTS) CLOSE(I)                                                   AT 4A6E
  400    CONTINUE                                                                 AT 4A91
         GO TO 50                                                                 AT 4A93
      END IF                                                                      AT 4A98
  500 STOP                                                                        AT 4A98
C
      END                                                                         AT 4AAB
```

End of compilation - Clocked 0.0 seconds

APPENDIX F
Further testing of the MODFLOW VKD code

FURTHER TESTING OF THE MODFLOW VKD CODE

ENVIRONMENT AGENCY

NATIONAL GROUNDWATER & CONTAMINATED LAND CENTRE

FURTHER TESTING OF THE MODFLOW VKD CODE

September 2003

1821/R2

Prepared for:

Environment Agency
National Groundwater & Contaminated Land Centre
Olton Court
10 Warwick Road
Olton
Solihull B92 7HX

Prepared by:

Water Management Consultants Ltd
23 Swan Hill
Shrewsbury SY1 1NN

CONTENTS

TABLES

FIGURES

APPENDICES

1 INTRODUCTION

1.1 Background

This report describes the work undertaken by Water Management Consultants (WMC) following a request received from the Environment Agency (EA), National Groundwater and Contaminated Land Centre (NGWCLC) for further testing of MODFLOW-VKD.

MODFLOW-VKD was developed, documented and tested by WMC under the NGWCLC Project NC/00/23. The code of MODFLOW-VKD includes modifications that allow the simulation of variable hydraulic conductivity and storage with depth within a model layer. The code and User Guide for MODFLOW-VKD (EA, 2002a and EA, 2003) were provided by WMC to the NGWCLC initially in April 2002.

The work described in this report implements the contents of the proposal detailed in the letter *Further testing of the MODFLOW VKD Code, Groundwater Technical Services Contract #11589*, which was submitted to the NGWCLC on 30 October 2002.

This report was prepared as an appendix to be inserted in the September 2003 revised version of the MODFLOW-VKD User Guide, initially published in March 2002 (EA, 2002). For the Agency Sarah Evers was Project Manager and Paul Hulme the Project Board.

1.2 Objectives

The main objective of the work is to ascertain that the modifications introduced to produce variable hydraulic conductivity (K) and storage (S) with depth have not altered the core function of MODFLOW. This involves examining whether the results produced by the VKD version of the code used for simulations with no requirement for variable K and S with depth are the same as those generated by the unaltered MODFLOW. Additional objectives include checking whether the use of different compilers to create the executable for MODFLOW-VKD affects the model output and defining how MODFLOW-VKD can be used in conjunction with the particle-tracking program, MODPATH (USGS, 1994) and the optimisation and sensitivity analysis program, UCODE (USGS, 1998).

1.3 Structure of report

Section 2 presents the methodology that was employed to compare the results produced by the unmodified and VKD version of MODFLOW. Section 3 describes the execution of the test problems and details the implications of using MODFLOW-VKD in combination with MODPATH (USGS, 1994) and UCODE (USGS, 1998), whilst Section 4 contains a summary, conclusions and recommendations related to the further testing of MODFLOW-VKD. Appendix A includes the tables describing each test problem.

2 METHODOLOGY

The VKD version of MODFLOW enables the simulation of variable hydraulic conductivity and storage with depth within a model layer. A series of tests are performed to verify that the VKD modifications have not altered the basic function of MODFLOW.

The following tests were employed:

- 12 of the 20 US EPA instructional problems available with associated documentation at http://www.epa.gov/ada/csmos/models/modflow.html. These tests were developed to give users experience in the use of MODFLOW and provide a documented testing of the code.
- 7 problems available at the USGS web site where MODFLOW 96 can be downloaded, http://water.usgs.gov/software/modflow-96.html. Documentation for 4 of these tests is available at the same web site. These tests are designed to illustrate and test specific boundary conditions such as the lake and inter-bed storage packages.
- The Wirral groundwater model. This is a relatively large time-variant model of the Permo-Triassic Sandstone developed by the EA North-West Region with assistance from WMC (EA, 2002).
- The GPZ model for the Tadcaster brewery boreholes. The model was developed and documented by WMC for the EA North-East Region (EA, 1997). This test is employed to run MODPATH (USGS, 1994) in combination with MODFLOW-VKD.
- A variation on one of the original test models used for testing MODFLOW-VKD (EA, 2002). This test is employed to run the USGS optimisation and sensitivity analysis program, UCODE (USGS, 1998), in combination with MODFLOW-VKD.

The tests are based on a model that was run using the following versions of MODFLOW:

- unmodified MODFLOW-96 code (MF96), obtained from http://water.usgs.gov/software/modflow-96.html and compiled with the Salford FORTRAN77 compiler to produce the executable MODFLW96.exe.
- MODFLOW-VKD (MF-VKD), compiled with the Salford 77 compiler to produce the executable MF-VKD1.exe[1]. This executable was provided by Water Management Consultants to the EA, NGWCLC in April 2002. The MF-VKD1.exe executable was the first release of the VKD code. Subsequently, further development has continued such that at the time of writing the current VKD executable is MF-VKD7.exe
- The executables of MF96 and MF-VKD compiled with the Compaq FORTRAN77 compiler. These were employed on two tests (problem 1 and 2 of the USGS suite) in order to verify whether the use of a different compiler would generate differences in the results produced by the two model versions.

For each test, the following model runs were performed:

A. Use of MF96.

B. Use of MF-VKD with the VKD function inactive. This means that no data required to account for a variation of K and S with depth were introduced. The input dataset was modified only by removing the comments on the first line of the .BCF file, in order to make its input format compatible with MF-VKD.

C. Use of MF-VKD with the VKD function active. This means that data required to account for a variation of K and S with depth were introduced. In order to make the comparison between the results of this run and those generated by MF96 meaningful, the VKD parameters were set so that the hydraulic properties of the test model would not alter.

In addition to the above runs, further simulations were performed on some of the test problems, as follows:

C1. Problems 1 and 2 in the USGS suite. Use of MF-VKD (with the VKD function active) compiled using the Compaq FORTRAN compiler, which is available to the Environment Agency.

D. GPZ Tadcaster model. Use of the auto-conversion option available within MF-VKD to produce a depth-dependent K field, which reproduces the transmissivity associated with the model steady-state solution.

The pre-conjugate gradient (PCG2) solver is not employed in most of the test models in the US EPA and USGS suite, despite being the most efficient in the execution of real, complex models. For this reason, an additional run based on the use of the PCG2 package was performed on Test Problem 8 and 12 (problems 13 and 20 of the EPA series). Note that the Environment Agency discourages the use of the SOR and SIP matrix solvers for MODFLOW due to potential inaccuracies in solutions as discussed in Osiensky and Williams (1997).

Three types of modifications to the test model dataset were applied:

- Modifications aimed at producing output files that would contain sufficient information on flows and heads to allow a thorough comparison between the results produced by the unmodified and VKD version of MODFLOW. These modifications are detailed in Section 2.1.
- Modifications aimed at making the format of the input dataset compatible with MF-VKD. These consisted only in removing the comments in the first line of the .BCF file when they were placed in positions employed in MF-VKD to define some VKD parameters.
- Modifications aimed at reproducing the hydraulic properties of the test model using VKD parameters. The means of reproducing the hydraulic properties of a model using VKD parameters is detailed in Section 2.2.

2.1 Set-up and comparison of output

The test model datasets were modified in order to generate the output files with the following extensions:

- OUT file with heads, drawdowns and volumetric budgets printed at the end of each time step.
- CBB file with all flow components printed at the end of each time step.

In order to produce these output files the input dataset was modified as follows:

- NAM file:

  Replace the first line with
  LIST 6 *_new_X.OUT
  Replace the last line with
  DATA (BINARY) 50 *_new_X.CBB

The symbol * represents the root name for the test model and X is the run type (A, B, C, C1, D).

- BAS file:

  Set ISTRT = 1 (flag to save starting heads, so that drawdowns can be calculated).

- OC file:

  For each time step, set IHDDFL = 1, ICBCFL =1, IBUDFL =1 (save or print heads and drawdowns, cell-by-cell flow and volumetric budget). Set INCODE = 0 (read output head/ddn specifications on one line and apply to all model layers), Hdpr =1 and Ddpr = 1 (print or save heads and drawdowns) in the first time step, INCODE = -1 (apply same output head/ddn specifications as in previous time step) in the following time steps.

- BCF and other flow component files (e.g. .WEL, .DRN):

  Set I*CB = 50 (unit where flow component is saved), where * represents the extension of the flow component file (e.g. IWELCB, IDRNCB).

Detailed instructions on how to specify the output settings in the MODFLOW files are provided in the MODFLOW Manual (USGS, 1996).

The output files were compared according to the following procedure:

- OUT files.

  These are ASCII files that can be viewed in any text editor. Ultra Edit-32©, version 6.20b was used (IDM Computer Solutions, 1999), since it can open large files and has a 'Compare files…' facility that creates a text file listing the lines where the files exhibit differences. The comparison files were named comp_out_*_XY.txt, where * represents the root name of the model dataset and X and Y are the runs (A, B, C, C1, D) being compared.

- CBB files.

  These are binary files that cannot be directly viewed with a text editor. An ASCII version of these files was created using the utility program cbc2gv.exe developed by WMC, which was given the same root name as the .CBB file and a .DAT extension. The ASCII files were then compared using the 'Compare files…' facility within Ultra Edit-32. The comparison files were named comp_cbb_*_XY.txt, where * represents the root name of the model dataset and X and Y are the runs (A, B, C, C1, D) being compared.

  The .CBB files produced in the two tests where MF-VKD1 is compiled by the Compaq FORTRAN compiler are in a different binary format from those generated by MF-VKD compiled by the Salford FORTRAN compiler. As a consequence, they cannot be recognised by cbc2gv.exe, which operates on .CBB files in the binary format produced with MF compiled using the Salford compiler. In order to create the ASCII version of the .CBB files, these were therefore produced as direct access binary files. This was achieved by modifying the last line of the name file as follows:

  from
  ```
      DATA (BINARY) 50    *_new_X.CBB
  ```
  to
  ```
      DATA (BINARY) 50    *_new_gvX.CBB   DIRECT 1.
  ```

  The ASCII version of these .CBB files was then created using the utility cbc2dos.exe, which reads direct access binary files.

2.2 Set-up of VKD function

In order to activate the VKD function in run C of each test as described in Section 2, the following parameters need to be assigned:

- LAYCON (4 for unconfined conditions, 5 for confined-unconfined conditions)
- For each model layer:
  - If the simulation is in time-variant mode
    - Sf1 (**Sbase**, base unconfined storage coefficient)
  - HY (**Kbase**, base hydraulic conductivity)
  - **BOT** (layer bottom elevation)
  - If not the bottom layer
    - VCONT
  - If LAYCON = 5 and the simulation is in time-variant mode
    - Sf2 (confined storage coefficient)
  - If LAYCON = 5
    - **TOP** (layer top elevation)
  - If the wetting capability is active
    - WETDRY (combination of wetting threshold and flag indicating which neighboring cells can cause a cell to become wet)
  - **VMID** (elevation of the point of inflection for hydraulic conductivity, where hydraulic conductivity changes from being constant to varying with depth)
  - **VKGRAD** (gradient factor applied to the zone where hydraulic conductivity is variable. It is multiplied by the base hydraulic conductivity to give the hydraulic conductivity gradient)
  - **VKMAX** (the maximum hydraulic conductivity factor applied to the zone where hydraulic conductivity is variable. It is multiplied by the base hydraulic conductivity to give the maximum hydraulic conductivity)

- If the simulation is in time-variant mode
  - **VSMID** (elevation of the point of inflection for the unconfined storage coefficient, where the unconfined storage coefficient changes from being constant to varying with depth)
  - **VSGRAD** (unconfined storage coefficient gradient factor applied to the zone where the unconfined storage coefficient is variable. It is multiplied by the base unconfined storage coefficient to give the unconfined storage coefficient gradient)
  - **VSMAX** (the maximum unconfined storage coefficient factor applied to the zone where the unconfined storage coefficient is variable. It is multiplied by the base unconfined storage coefficient to give the maximum unconfined storage coefficient).

The VKD parameters in bold are represented in Figures 1a and 1b, which show the conceptual model for the variation of hydraulic conductivity and storage coefficient with depth as illustrated in the MF-VKD User Guide, Figures 2.1 and 2.2 (EA, 2003).

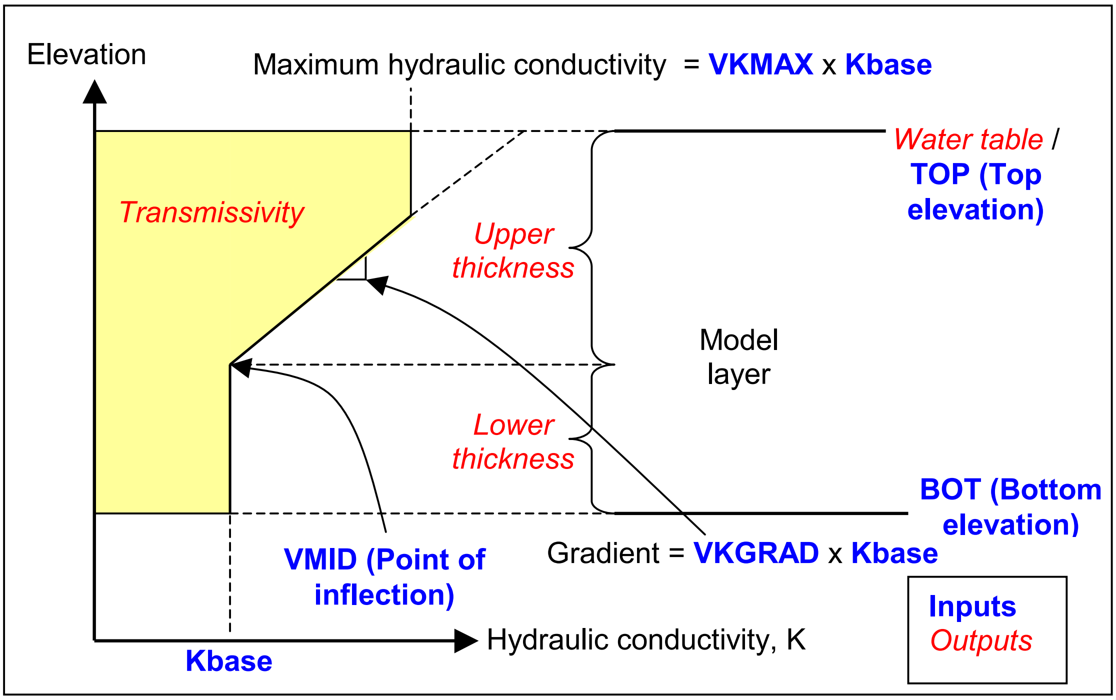


Figure 1a Conceptual model for the variation of hydraulic conductivity with depth

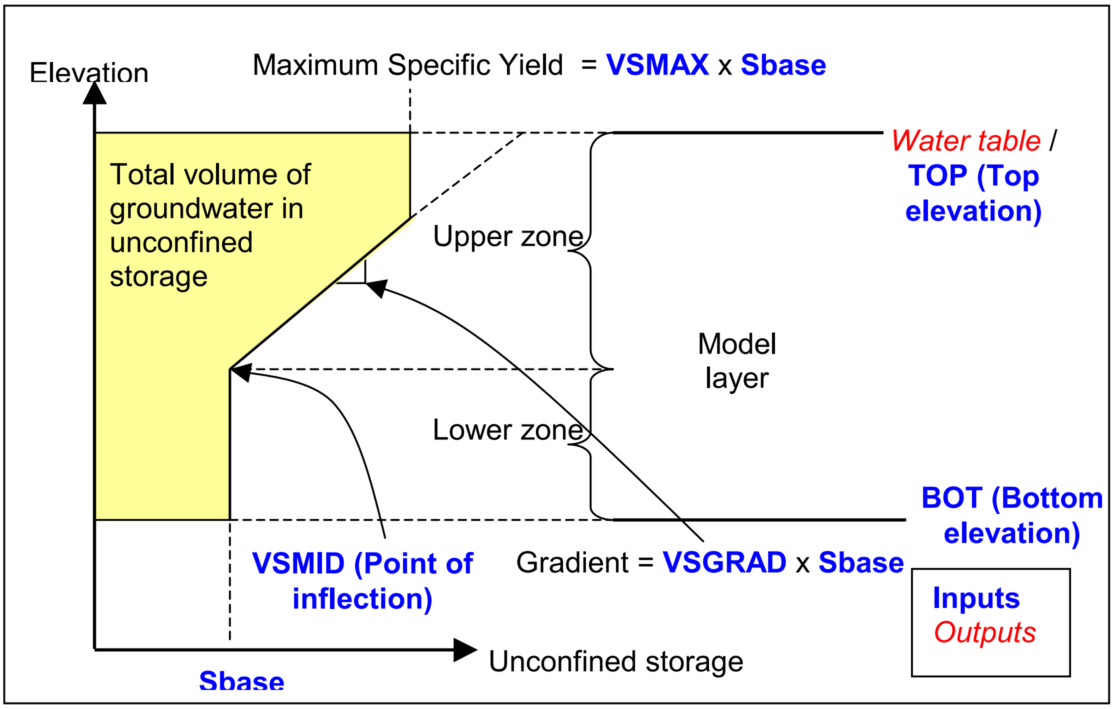


Figure 1b Conceptual model for the variation of storage with depth

The listed VKD parameters are assigned in the .BCF file according to the format described in the MF-VKD User Guide (EA, 2003).

In order to reproduce the hydraulic properties of the test models using the VKD parameters, the following settings have been applied:

- LAYCON value conversion. For the VKD function to be active, a value of 4 (unconfined conditions) or 5 (confined-unconfined conditions) must be assigned to LAYCON. Table 1 shows the conversion between the LAYCON value according to the unmodified MODFLOW and the VKD LAYCON value with associated comments.

Table 1 Conversion of LAYCON values from MF96 to MF-VKD

LAYCON according to unmodified MODFLOW	VKD LAYCON	Comments
0	5	Strictly confined conditions. These can be reproduced by setting values of TOP and BOT that are guaranteed to be always lower than the head range throughout the simulation. The difference between TOP and BOT is set to 1, so that values of transmissivity, T, used by unmodified MODFLOW are equal to the values of the base hydraulic conductivity, Kbase, used by MODFLOW-VKD.
1	4	Unconfined conditions.
2	N/A	Confined-unconfined conditions where T is constant and S varies over time. It is not possible to reproduce these conditions with LAYCON = 5, as both T and S vary over time.
3	5	Confined-unconfined conditions where T and S vary over time.

- VMID (and VSMID values in time-variant simulations) is set equal to BOT. This means that the inflection point is placed at the bottom of the layer.
- VKGRAD (and VSGRAD in time-variant simulations) is set equal to 1. This means that a linear variation of K (S) of 1 x Kbase (1 x Sbase) occurs for a unit change in depth within the layer.
- VKMAX (and VSMAX values in time-variant simulations) is set equal to 1. This means that K (S) is kept equal to Kbase (Sbase) within the layer, thus overriding the variation set with VKGRAD (VSGRAD).

The VKD parameter setting described in the above three bullet points is shown in Figure 1c.

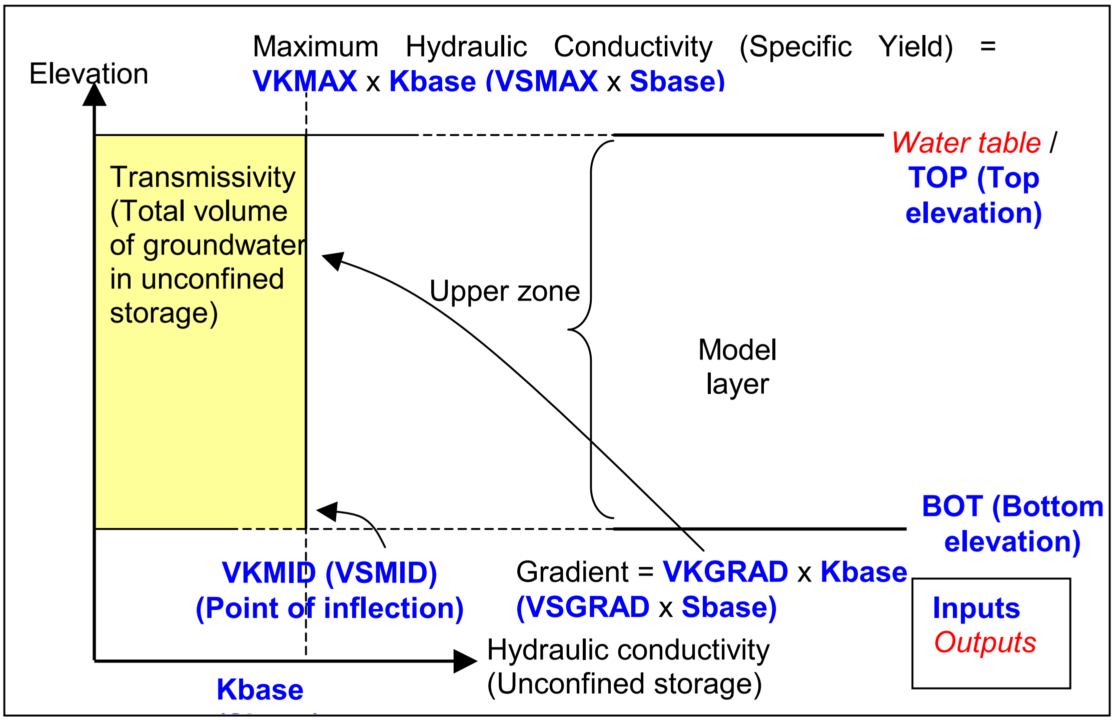


Figure 1c Representation of depth-uniform hydraulic conductivity (unconfined storage) using VKD parameters

3 EXECUTION OF TEST PROBLEMS

The test models employed to identify the differences between the results produced by the unmodified and VKD version of MODFLOW are detailed in the tables shown in Appendix A. For each test, the following information is provided:

- Model name and dataset root name.
- Reference documentation (USGS / US EPA / EA report where the model is described).
- Aim and brief description of test.
- Input files required in test.
- Changes required for execution with MF96 and MF-VKD (changes in output setting not included).
- Type of differences identified between the output files generated by MF96 and MF-VKD, with VKD function active and inactive.

The type of differences identified between the output files generated by the unmodified and VKD version of MODFLOW were classified as follows:

- Differences in headers and comments in the .OUT file. These are differences in the format of the output files that are not problem-specific, ie the same differences were identified in all tests. These differences are not related to a change in the basic MODFLOW functionality due to the presence of VKD components and are therefore not mentioned in the test summary tables.
- Differences in memory allocation printed in the .OUT file. The VKD version of MODFLOW increases the size of the X-array, a general array where MODFLOW stores most of its variable arrays, from 1,500,00 to 10,000,000 elements (an element being equal to 4 bytes). Different numbers of elements from the X-array are also allocated for the variable arrays required by the BCF and other packages employed in each test model. These differences do not compromise the basic MODFLOW functionality due to the presence of VKD components and are therefore not mentioned in the test summary tables.

- Differences in the values of head, drawdown, components of the volumetric budget of cell-by-cell flows in the .OUT and .DAT version of the .CBB file. “Type I” differences are defined as small differences in global flow between balance components (i.e. less than 0.1%), and in heads and drawdowns (less than 1%). These are due to round-off errors deriving from the use of different parameters and variables in the calculation of the coefficients of the flow equation in MF96 and MF-VKD. In particular, some of the parameters and variables employed by MF-VKD to calculate storage coefficients and conductances are declared in the source file BCF-VKD6.for (subroutines BCF5FM, SBCF5S, SBCF5H and bcf5vk) using double precision. Larger differences (defined as “Type II”) may be related to a change in the basic MODFLOW functionality due to the presence of VKD components and therefore need further investigation.

Whenever differences between the drawdowns, heads and maximum changes in heads during an iteration emerge in the .OUT files generated by the MF96 and MF-VKD (VKD function active and inactive), the associated cell-by cell flows are compared by subtraction within Excel, in order to identify whether significant discrepancies are present.

3.1 GPZ model testing

The unmodified and VKD version of MODFLOW were run in combination with MODPATH using the flow and particle-tracking model employed to define groundwater protection zones for the Tadcaster brewery boreholes. The model was developed for the Environment Agency, Northeast Region, by WMC (EA, 1997). The version of the model employed in the test is referred as Tad401. This model version was used to define the Outer and Total Catchment Zone for the source 0577 (EA, 1997). The MODPATH files for this model were created using the ‘Create dataset’ facility within GWVistas. MODPATH version 3.0 included in the GWVistas software was used to perform this test. In order to run MODPATH outside GWVistas, the MODPATH input files generated by GWVistas were renamed and modified as follows:

- Response file, mpath.rsp. This file was renamed mpath_Tad401.rsp and the name file listed in it was renamed.
- Name file, modpath.dat. This file was renamed mpath_Tad401.dat.
- The head and cell-by-cell flow file names were replaced by *Tad401_new_gvX.hds* and *Tad401_new_gvX.cbb*, respectively, where X indicates the run type (A, B, C, D). These are direct access binary files that are created from the binary files produced by MF96 and MF-VKD using the utilities hds2gv.exe and cbc2gv.exe. Direct access binary files are required by the version of MODPATH available within GWVistas. It is noted that direct access binary files can be created directly by MF96 and MF-VKD by adding the expression DIRECT 1 after the name of the head and cell-by-cell flow file listed in the .NAM file. The reason why this method was not employed to create the direct binary .CBB files required by MODPATH in this test is that the utilities hds2gv.exe and cbb2gv.exe were used to convert the .CBB files into direct access binary format as well into ASCII, .DAT format.
- The files *Tad401_X.ept* and *Tad401_X.ptl* are added to the file list. These files contain information about the particle final positions and the particle pathlines, respectively.

The results generated by the runs performed for this test were compared using the .OUT, the ASCII version of the .CBB and .HDS files.

Besides the runs A, B and C, an additional run, D, was performed in this test. The purpose of this run is to ascertain that the VKD function works correctly. In this run, hydraulic conductivity is set to vary with depth in such a way that the transmissivity resulting from integration over the steady-state saturated thickness produced by the unmodified MODFLOW is the same as the steady-state transmissivity based on the constant hydraulic conductivity field of the original model.

This is illustrated in Figure 2, where K_0 is the original hydraulic conductivity and the remaining parameters are the same as those shown in Figures 2.1 and 2.2 in the MF-VKD User Guide (EA, 2003).

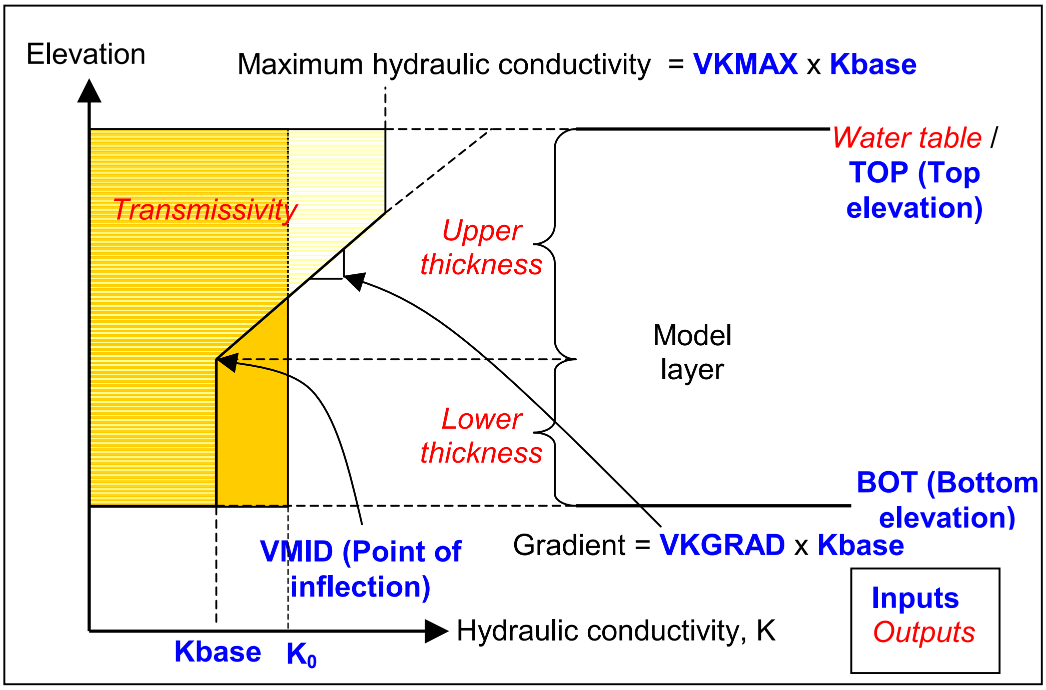


Figure 2 Equivalence between transmissivity associated with constant and depth-veriable hydraulic conductivity

The auto-conversion option available in MF-VKD was employed to define the depth-dependent hydraulic conductivity field. The use of the auto-conversion option is described in Sections 3.3 and 8.4.1 of the MF-VKD User Guide (EA, 2003). The following VKD parameters were set in the .BCF file (renamed tad401auto.bcf) to define the hydraulic conductivity field:

- ISS (steady-state) flag = 3.
- ITRANS = 1 (flag specifying that internodal transmissivities will be output to the .OUT file).
- Thickness of upper zone = 5 m (this value is automatically re-set equal to the saturated thickness whenever the saturated thickness value is less than 5 m).
- VKGRAD = 1.
- VKMAX = 3.

Run D was performed using variable K with depth as described above and employing the steady-state heads produced by MF96 as initial conditions. Steady-state heads are used as initial conditions to ensure that the simulation converges on a solution quickly. The correct functioning of VKD is verified in this test by checking that the steady-state heads produced by MF-VKD are identical to the initial conditions, i.e. the steady-state heads generated by MF96. The differences in heads calculated by runs A and D were checked and found to be less than 0.02 m throughout the model domain.

The pathlines generated by MODPATH coupled with MF96, Run A, and MF-VKD, Run D, after 10000 days are shown in Figure 3a in three dimensions for the entire modelled domain, whereas Fig 3b zooms on two of the pathlines generated by the two model versions in order to highlight their discrepancy.

Figure 3b shows that the pathlines generated by MODPATH using the MF96 and MF-VKD head and cell-by-cell flow files have non-negligible discrepancies, with a maximum distance between pathlines of 113 m (in three dimensions). The comparison between the .ptl files produced by MODPATH based on the heads and flows generated by MF96 and MF-VKD in runs A and D confirms the presence of differences in the particle positions, travel times and number of cells that particles cross along their pathline.

The reason for the differences in the positions of the pathlines calculated by runs A and D is due to small differences between the heads and flows calculated by these runs. The maximum head difference between the two models should not exceed twice the head convergence criterion of 0.01 m. The difference of 0.017 falls within the expected range. Due to large values of hydraulic conductivity (1000 m^2/d) and large contrasts in K from one cell to the next (of up to 100:1 at the main fissure locations), small head differences result in rather larger differences in flow. The maximum difference in flow is 15 m^3/d. This is less than twice the flow convergence criterion of 8 m^3/d.

The differences in head are due in part to the very slightly different values of transmissivity being used in simulations A and D. With VKD active very small rounding errors are introduced when calculating the base hydraulic conductivities in the auto-conversion simulation. This leads to the small differences in transmissivity values and consequently to slightly different calculations being used during the iterative process, resulting in a slightly different solution being found. Since the maximum differences are so slight, of the same order as the models convergence criteria, they do not in any way affect the validation of the VKD code. Whilst tighter convergence criteria would reduce head and flow differences, it was found that the large contrasts in hydraulic conductivity present in the model proved troublesome, and the model failed to converge.

The differences in head, and more particularly the differences in flow, affect the trajectory of the pathlines calculated by MODPATH. If the trajectories are changed enough that a flowline enters a different cell (in comparison to the previous simulation – this may be the result of only very small changes in trajectory), then the different flow conditions in that cell will have a greater influence in further diverting the trajectory of that flowline.

In the Tadcaster model runs A and D the maximum horizontal difference in pathline locations is around 30 m. The pathlines subsequently converge. The pathlines diverge at the locations of high contrast in hydraulic conductivity located around the fissures as described above. Around this area of the model the model cells have sides of 100 m. As the precision in the locations of the fissures is determined by the cell size, it can be seen that the uncertainty in the position of the pathline is less than a third of the uncertainty in the location of the fissure. It should also be noted that this 30 m discrepancy is small compared to the changes to pathline positions produced in sensitivity analysis (as used, for instance, in defining source protection zones).

So, whilst slight differences in pathlines have been found, they are due to very small rounding errors. The resulting head and flow discrepancies are dependant on the model convergence criteria and have no affect on the validity of the VKD code.

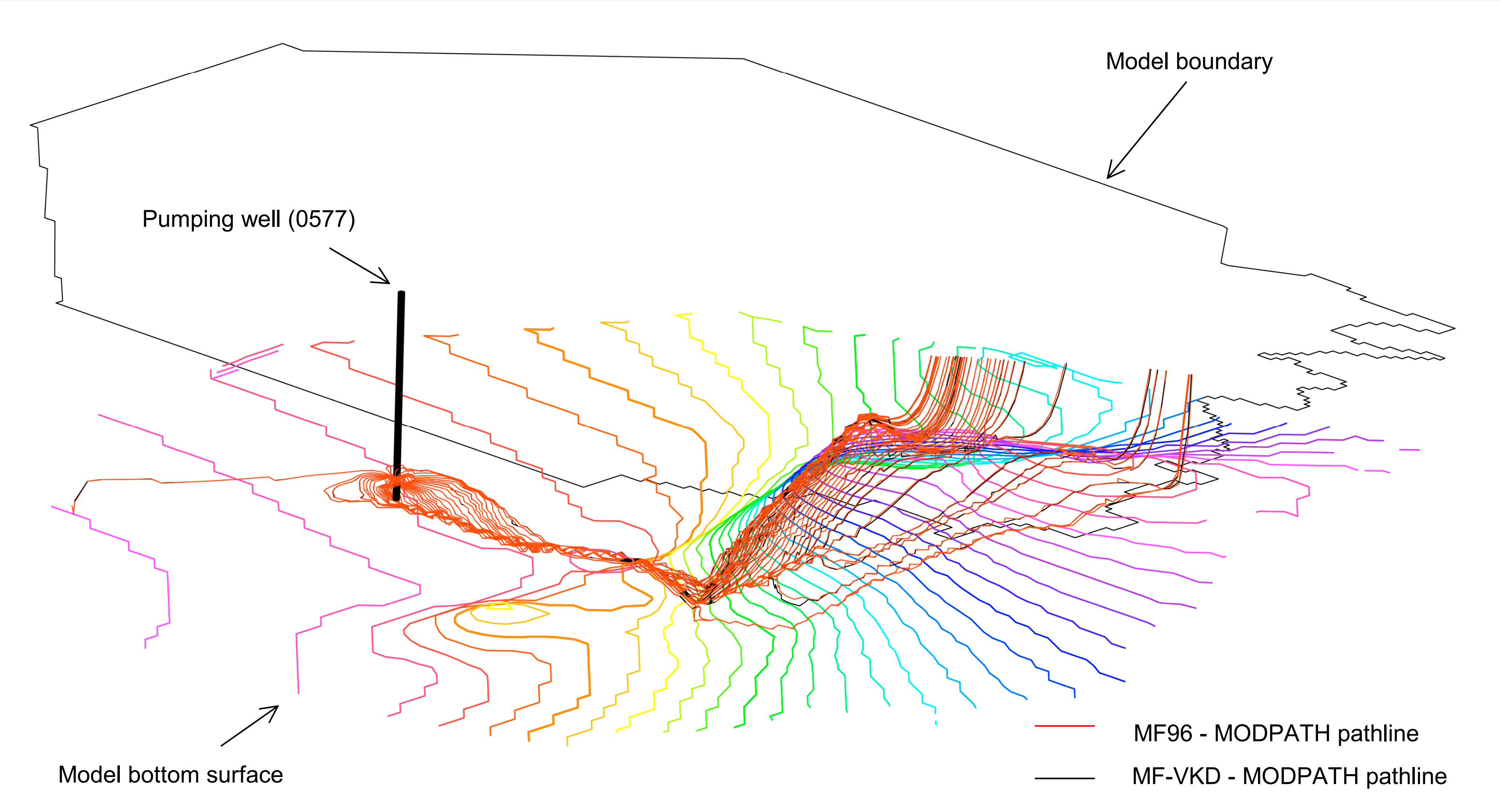


Figure 3a 3D view of pathlines generated by MF96 and by MF-VKD with depth-variable K yielding the same steady-state transmissivity – full model domain

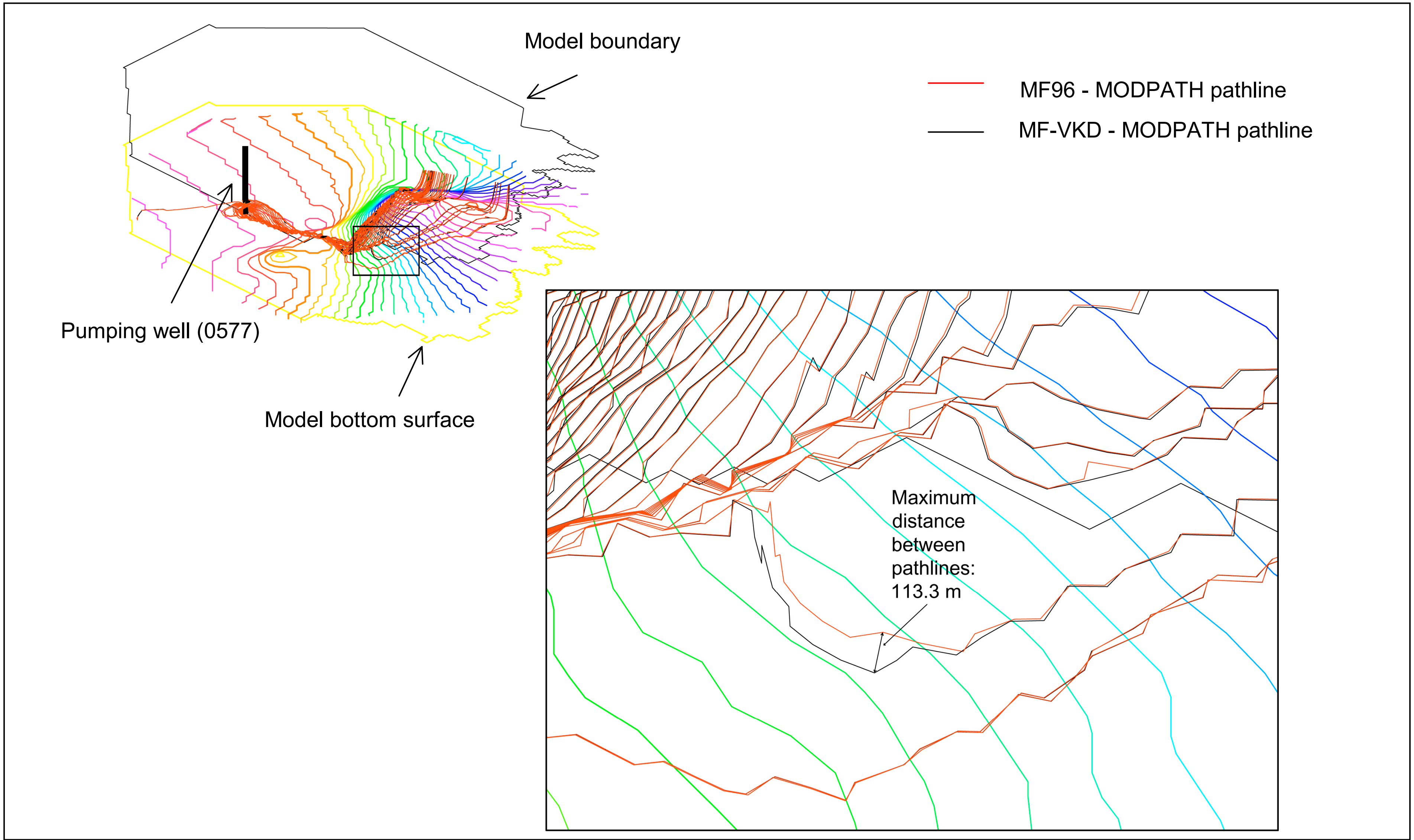


Figure 3b Example of discrepancy between the pathlines generated by MF96 and by MF-VKD with depth-variable K yielding the same steady-state transmissivity

3.2 Implications of performing particle tracking with VKD

When running MODPATH using the results of a model where VKD is used to set variable hydraulic conductivity with depth within a layer, attention must be drawn to how MODPATH computes the velocities used to define particle pathlines. In MODPATH average velocities across the faces of each cell are computed and are assumed to be constant within the cell face. These velocities are calculated by dividing the volume flow rates across the cell faces by the cell cross sectional areas and the cell porosity. The velocity of a particle located within a cell is then calculated by means of linear interpolation of the velocities defined at the cell faces.

The use of constant, depth-averaged velocities for each cell face means that MODPATH does not account correctly for the variation of hydraulic conductivity with depth within a model layer.

The implication of how MODPATH computes velocities is that when particle tracking is performed on a flow model where VKD is present, velocities can be under or over-estimated, according to the depth at which particles are released. In particular, the velocities for particles released in the low K zone will be over-estimated, whereas velocities for particles released in the high K zone will be under-estimated.

When particle tracking is applied with VKD to identify groundwater protection zones (GPZs) as in the Tadcaster model, run D, described in Section 3.1, the under-estimation of velocities for particles moving in the high K zone equates to the definition of smaller zones. Figure 4 compares particle tracks to a well for a standard MODPATH simulation to those that would be expected if MODPATH took VKD into account.

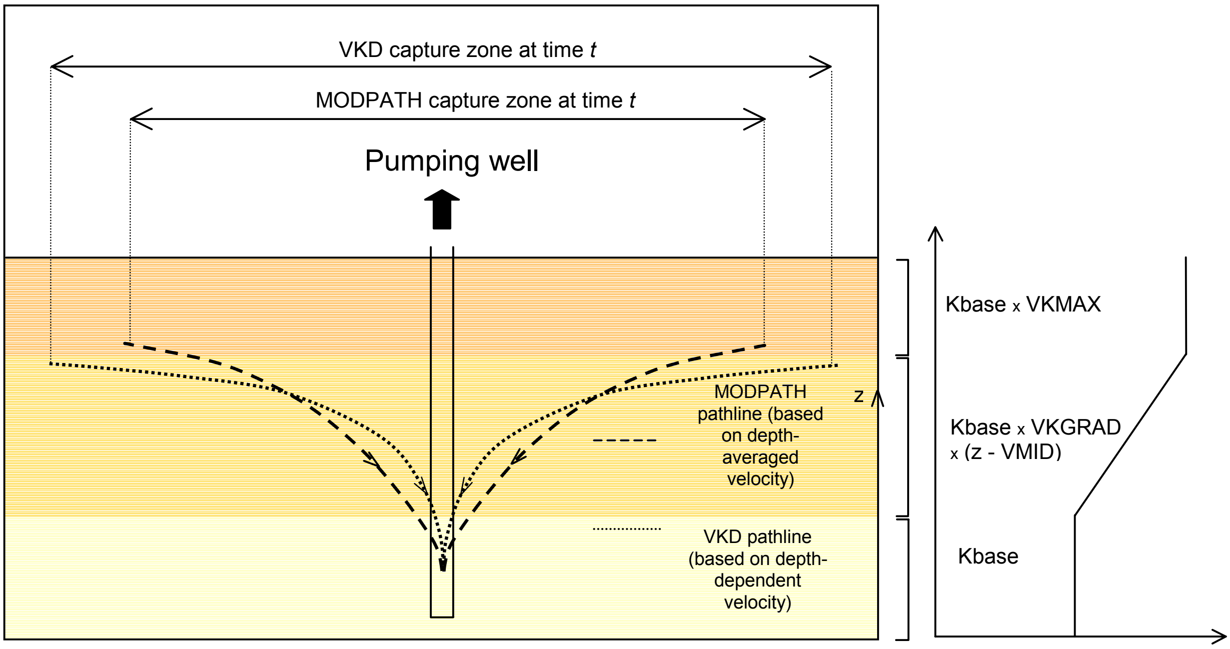


Figure 4 Effect of velocity computation method on pathlines

In order to reproduce fully the effect of variable K with depth, velocities across the faces of a cell should be computed in MODPATH as the ratio between the Darcy velocity and porosity, ie

$$v_i(z) = \frac{1}{\phi}\frac{k(z)\Delta h}{\Delta x_i}$$

where
$v_i(z)$: velocity in i direction at elevation z within the cell face
$k(z)$: hydraulic conductivity at elevation z
Δh: head difference across the cell face
Δx_i: distance between the centres of the cells separated by the cell face
ϕ: porosity

Using the above expression, velocities would no longer be constant but would vary with depth within a cell face according to K, and pathlines would be calculated accordingly.

3.3 Execution of test using UCODE

UCODE is a sensitivity analysis and parameter estimation program developed by the USGS (UGSG, 1998). In order to explore if MODFLOW-VKD can be made to work with UCODE, one of the test models used in the original testing program was applied. The aim of this test was to use UCODE to calculate the sensitivity of VKD parameters.

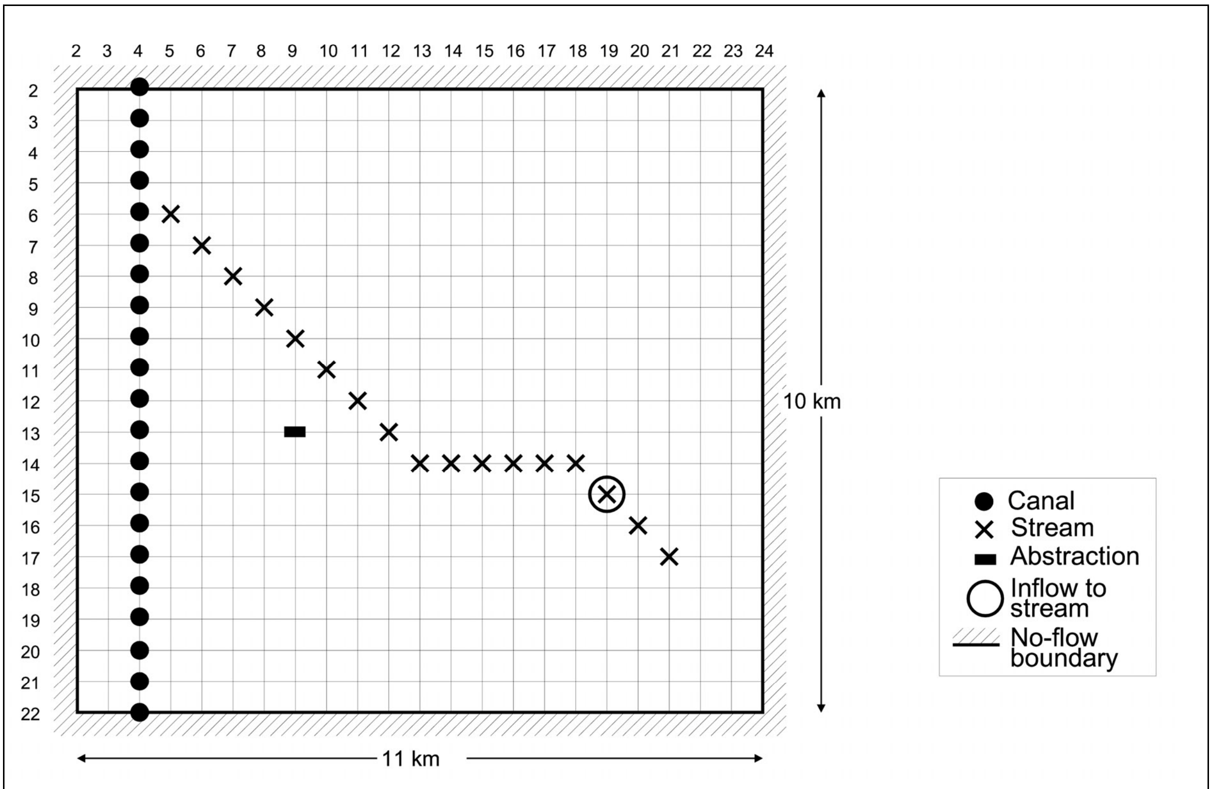


Figure 5 Simple model used for UCODE testing

Simple modifications were necessary to the VKD dataset with more detailed changes to UCODE input files. The testing model is a simple time-variant model of a hypothetical aquifer system with VKD, inflows from areal recharge and discharges to a well, a river and a stream (Figure 5). The model is described in more detail in the Enhancements to MODFLOW VKD report (EA, 2002a).

The model was run using MODFLOW-VKD in conjunction with UCODE. The following modifications were made to the MODFLOW input files:

- The name (*.nam) file was changed so that the binary files were written in direct access format (by adding 'DIRECT 1' after the relevant file names). This was done so that the Groundwater Vistas program 'targpest' could be used to extract simulated head values. These could then be used by UCODE to evaluate the sensitivity of the VKD parameters.
- The Block Centred Flow (BCF) package was modified so that the value for the hydraulic conductivity gradient factor (VKGRAD) was replaced by the label '!VKGRAD1,!'. The file was then saved as a template file, which would then be used by UCODE to create the .BCF file, the label being replaced by different values to evaluate the sensitivity of this parameter.

The UCODE input files were then created as follows:

- The universal input file (*.uni) was set up to do a sensitivity simulation (PHASE=2) and contained details of the batch file required to run MODFLOW-VKD and 'targpest'. It also contained details of dummy observed values at a point in the centre of the model, varying over time.
- The prepare file (*.pre) gave details of the filenames of the template and .BCF files, along with details relating to the parameter to be varied, VKGRAD.
- The extract file (*.ext) gave instructions to UCODE on how to extract the simulated head values from the output file created by 'targpest'.
- A file called 'targpest.dat' was created to allow 'targpest' to extract head values from the MODFLOW heads file (*.hds) to the output file, 'targpest.out'.
- An application model batch file containing the commands to run MODFLOW-VKD ('mf-vkd1 <mfvkd.in') and 'targpest'.
- The input file ('mfvkd.in') containing the filename of the MODFLOW name file (*.nam).
- The UCODE batch file containing the command required to run UCODE ('perl ucode ucode1 0')

By looking at the main output file produced by UCODE, ucode1._ot, it was checked that UCODE ran successfully, executing MODFLOW-VKD and 'targpest' three times to evaluate the sensitivity of the simulated heads to changes in the hydraulic conductivity gradient factor. It is concluded that UCODE can be successfully applied to MF-VKD models once necessary alterations are made to data files.

4 SUMMARY, CONCLUSIONS AND RECOMMENDATIONS

This section summarises the results of the testing, draws conclusions about whether the VKD code modifications have changed the basic function of MODFLOW and makes some limited recommendations concerning the use of MF-VKD. Full results of each test problem are presented in Appendix A.

4.1 Summary of results

A series of tests were performed using the MF-96 and MF-VKD versions of MODFLOW to ensure that the VKD modifications do not affect the basic function of MODFLOW. The tests include:

- A suite of instructional models developed by the US EPA.
- Examples illustrating specific MODFLOW packages developed by the USGS.
- A resource assessment model for the Wirral peninsula.
- A model developed to define groundwater protection zones for the Tadcaster brewery boreholes.
- A simple VKD test model used to test UCODE.

A summary of the tests and the differences in results produced by MF96 and MF-VKD is shown Table 2.

For the EPA tests there are no significant differences between MF96 and MF-VKD. For the USGS datasets the only significant difference occurs in stream flow calculations where known changes have been deliberately introduced to improve the MF-VKD code.

The GPZ model for the Tadcaster brewery boreholes was run four times:

1. MF-96 to produce full output files for comparison.
2. MF-VKD inactive.
3. MF-VKD active but with no variation in hydraulic conductivity with depth.
4. MF-VKD active and with hydraulic conductivity varying with depth.

The fourth simulation with VKD active and hydraulic conductivity varying with depth was performed using the auto-conversion option available in MF-VKD to introduce a variation in K with depth. In steady state this yields the same transmissivity values as those obtained with the original constant hydraulic conductivity model. The heads and flows predicted when hydraulic conductivity varies with depth were the same as those predicted using MF96 and MF-VKD without VKD active.

Particle tracking was performed on all four Tadcaster tests. The only significant difference in results is found when comparing MF-VKD with VKD active to the results of MF-96. Maximum spatial differences in particle tracks of 113.3 m were seen. These differences are due to the sensitivity of particle tracking to small rounding / precision errors which can put particles into different model cells and then transport them with different gradients and velocities due to slightly different steady state solutions being produced by the two versions of MODFLOW.

Two tests were used to explore whether differences in FORTRAN compiler cause discrepancies in model results. The Compaq and Salford FORTRAN compilers were used in the comparison to create the MF-VKD executable. The use of different compilers does not affect MF-VKD / MF-96 results.

One of the test models used in the original testing program for MF-VKD was employed to check whether MF-VKD can work with the optimisation and sensitivity analysis program UCODE. The test dataset and UCODE templates were modified and UCODE successfully calculated the sensitivity of VKD parameters. UCODE can be applied with MF-VKD and if necessary can explore the sensitivity of VKD parameters.

Table 2 List of tests and discrepancies identified in the results produced by MF96 and MF-VKD

Test Number	Test Name	Differences in results
1	US EPA problem 1	Insignificant (Type I)
2	US EPA problem 2	Insignificant (Type I)
3	US EPA problem 3	Insignificant (Type I)
4	US EPA problem 5	Insignificant (Type I)
5	US EPA problem 8	Insignificant (Type I)
6	US EPA problem 9	Insignificant (Type I)
7	US EPA problem 12	Insignificant (Type I)
8	US EPA problem 13	Insignificant (Type I)
9	US EPA problem 14	Insignificant (Type I)
10	US EPA problem 16	Insignificant (Type I)
11	US EPA problem 18	Insignificant (Type I)
12	US EPA problem 20	Insignificant (Type I)
13	USGS problem 1	Insignificant (Type I)
14	USGS problem 2	Significant differences in stream outflows due to correction in the accreted streamflow calculated in MF-VKD
15	USGS problem 3	Insignificant (Type I)
16	USGS problem 4	Insignificant (Type I)
17	USGS problem 5	Insignificant (Type I)
18	USGS problem 6	Insignificant (Type I)
19	USGS problem 7	Insignificant (Type I)
20	Wirral model (EA, 2002)	Insignificant (Type I)
21	Tadcaster model (EA, 1997) with MODPATH	Significant differences in pathlines (Type II) due to small differences in head and flow (Type I).
22	VKD test model (EA,2002) with UCODE	Insignificant (Type I)

4.2 Conclusions

The results produced for the series of tests performed using the unmodified and VKD version of MODFLOW show that the introduction of VKD does not alter the basic functionality of MODFLOW. The output files generated by MF96 and MF-VKD show insignificant numerical differences, which are due to the different output formats, precision settings and round-off errors associated with the calculations performed in the two programs.

The only significant differences emerge from:

- Test problem 14 (problem 2 of the USGS series) of the USGS suite, where the stream, STR, package is employed. The results reveal differences in the accretion profiles computed by MF96 and MF-VKD that are due to a small correction in the calculation of streamflows that was introduced in MF-VKD.
- Test problem 21 based on the GPZ model for the Tadcaster brewery boreholes. The .ptl files produced by MODPATH show differences in the particle positions, travel times and the number of cells particles cross along their pathline. These differences arise as a result of small discrepancies between the heads and flows calculated by the two different versions of MODFLOW. Whilst these discrepancies are not significant in themselves, the effect on particle locations can be significant. However, the differences are less than those associated with other modelling uncertainties, and do not indicate any problems with the validity of the VKD code.

Other tests performed with UCODE and different compilers show that MF-VKD can be compiled with different compilers without problems and that UCODE can be used with MF-VKD.

During a modelling project involving the use of MODFLOW-VKD that the Environment Agency undertook at the time of the further testing described in this report, a problem was found in the MODFLOW-VKD code. This problem occurs when both confined and unconfined storage are used in the model domain. Upon close analysis of this problem, WMC discovered that if the point of inflection on the VSD (variable unconfined storage with depth) profile is set above the top elevation of the MODFLOW layer, then the unconfined storage value is used when the confined S should be used. This error was corrected and WMC provided the Environment Agency with the revised code.

4.3 Recommendations

The use of MF-VKD rather than MF96 for simple non-VKD models may require small modifications to the input data files. For the tests described in this report, such modifications consist only in removing comments in the first line of the .BCF file. Similarly, small modifications to the format of input files may be necessary when running versions of MF-VKD created with different FORTRAN compilers. These modifications can be viewed in the test data files included in the CD accompanying the revised version of the User Guide for MODFLOW-VKD.

The magnitude of the discrepancies in the pathlines and travel times produced by MODPATH when used in combination with MF96 and MF-VKD are the result of small differences in cell-by-cell flows due to numerical round-off errors.

Particle tracking with MODPATH on a model with the VKD function active must be interpreted with great care. The velocities used to define pathlines are depth-averages within a layer and are not a full reflection of the VKD function. Since the pathlines from particle tracking with VKD active using MODPATH are inaccurate, the results should be view with much care. However, given the magnitude of other uncertainties in assumptions underlying particle tracking, such as a continuous porous medium assumption in a Chalk aquifer, the additional uncertainty that VKD introduces to MODPATH pathlines may be relatively small. The Agency does not at present have plans to develop a VKD version of MODPATH. It does plan to issue further guidance on Groundwater Protection Zone delineation where MODFLOW VKD regional models exist.

REFERENCES

Andersen, P.F, 1993, *A manual of instructional problems for the USGS MODFLOW model.* US Environmental Protection Agency Report EPA/600/R-93/010.

Environment Agency, 1997, *Tadcaster Groundwater Protection Zones*. WMC Report 1345/R1 prepared for the Environment Agency, Northeast Region.

Environment Agency, 2002a, *Enhancements to MODFLOW – Variation in hydraulic conductivity and storage with depth*. WMC Report 1621/R1 prepared for the Environment Agency, National Groundwater and Contaminated Land Centre, Project NC/00/23.

Environment Agency, 2003, *Enhancements to MODFLOW – User guide for MODFLOW-VKD – A modified version of MODFLOW-96 to include variations in hydraulic properties with depth*. WMC Report 1621/R3 prepared for the Environment Agency, National Groundwater and Contaminated Land Centre, Project NC/00/23.

Fenske, J.P, Leake, S.A., Prudic, D.E., 1996, *Documentation of a computer program (RES1) to simulate leakage from reservoirs using the modular finite-difference ground-water flow model (MODFLOW)*, U.S. Geological Survey Open-File Report 96-364, 58 p. Available at http://water.usgs.gov/software/modflow-96.html.

Harbaugh, A.W., and McDonald, M.G., 1996, *User's documentation for MODFLOW-96, an update to the U.S. Geological Survey modular finite-difference ground-water flow model: U.S. Geological Survey* Open-File Report 96-485, 56 p. Available at http://water.usgs.gov/software/modflow-96.html.

Leake, S.A, Leahy, P.P., Navoy, A.S., 1994, *Documentation of a computer program to simulate transient leakage from confining units using the modular finite-difference ground-water flow model*, U.S. Geological Survey Open-File Report 94-59, 77 p. Available at http://water.usgs.gov/software/modflow-96.html.

Leake, S.A., and Lilly, M.R., 1997, *Documentation of a computer program (FHB1) for assignment of transient specified-flow and specified-head boundaries in applications of the modular finite-difference ground-water flow model (MODFLOW),* U.S. Geological Survey Open-File Report 97-571, 56 p. Available at http://water.usgs.gov/software/modflow-96.html.

Leake, S.A., and Prudic, D.E., 1997, *Documentation of a computer program to simulate aquifer-system compaction using the modular finite-difference ground-water flow model,* U.S. Geological Survey Techniques of water resources investigation, Book 6, Chapter A2, 74 p. Available at http://water.usgs.gov/pubs/twri/twri6a2/pdf/TWRI_6-A2.pdf.

McDonald, M.G., Harbaugh, A.W., Orr, B.R., Ackerman, D.J., 1991, Open-File Report 91-536, 103 p.

Osiensky, J.L. and Williams, R.E., 1997, *Potential Inaccuracies in MODFLOW Simulations Involving the SIP and SSOR Methods for Matrix Solution*. Groundwater, Volume 35, No. 2.

Poeter, E.P. and Hill, M.C., 1998, *Documentation of UCODE, a computer code for universal inverse modelling.* U.S. Geological Survey Water-Resources Investigations Report 98-4080, 116 p.

Pollock, D.W., 1994*, User's Guide for MODPATH/MODPATH-PLOT, Version 3: A particle tracking post-processing package for MODFLOW, the U.S. Geological Survey finite-difference ground-water flow model.* U.S. Geological Survey Open-File Report 94-464, 6 ch. Available at http://water.usgs.gov/software/modpath.html

Prudic, D.E., 1988*, Documentation of a computer program to simulate stream-aquifer relations using a modular finite-difference ground-water flow model,* Open-File Report 88-729, 119 p.

Taylor, A.B., Hayes. P.J., Hulme, P.J., 2002*, Examination of Groundwater Flow Model Sensitivity Analysis and Parameter Optimisation using UCODE, prepared for the Environment Agency, National Groundwater and Contaminated Land Centre, Project NC/01/60.*

APPENDIX A
Results of the test problems

List of test problems

Test number	Test name	Description
1	US EPA problem 1	The Theis solution
2	US EPA problem 2	Anisotropy
3	US EPA problem 3	Artesian-water table conversion
4	US EPA problem 5	Mass balance
5	US EPA problem 8	Grid and time stepping considerations
6	US EPA problem 9	Calibration and prediction
7	US EPA problem 12	Leaky aquifers
8	US EPA problem 13	Solution techniques and convergence
9	US EPA problem 14	Head dependent boundary conditions
10	US EPA problem 16	Evapotranspiration
11	US EPA problem 18	Cross-sectional simulations
12	US EPA problem 20	Application to a hazardous waste site
13	USGS problem 1	Storage depletion (use of inter-bed storage, IBS1, package, which simulates storage changes and compaction in fine-grained beds and confining beds. Used to reproduce aquifer compaction and land subsidence)
14	USGS problem 2	Example problem in OFR 88-729 (use of stream, STR, package)
15	USGS problem 3	Example problem in TWRI 6-A1
16	USGS problem 4	Example problem in OFR 91-536
17	USGS problem 5	Example problem 1 in OFR 94-59 (use of transient leakage, TLK1, package to simulate storage depletion in confining beds)
18	USGS problem 6	Example problem in OFR 96-364 (use of reservoir, RES1, package to simulate leakage from a reservoir)
19	USGS problem 7	Example problem in OFR 97-571 (use of transient specified flow and specified head boundary, FHB1, package)
20	Wirral117tv	Wirral regional model
21	Tad401	Tadcaster model (EA, 1997) with MODPATH
22	VKD19-tv	VKD test model (EA, 2002a) with UCODE

Test Problem 1				
Reference	**Name**	**Description**	**Dataset Root Name**	**MF optional files employed**
US EPA problem 1 [modflmn.pdf, 1-1]	The Theis solution (Part a)	Confined model, 1 layer, 19 rows, 19 cols with mesh spacing reduction around pumping well in centre of grid. One stress period of 1 day and 20 time steps.	pr1a	.WEL .SIP .OPC

Change to input data files		
MODFLOW 96 (A)	**MODFLOW-VKD VKD inactive (B)**	**MODFLOW-VKD VKD active (C)**
None	None	Set BOT = –100 m and TOP = –99 m in order to reproduce strictly confined conditions. Kbase=T, VMID=BOT, VKGRAD=1, VKMAX=1, Sbase=Sf1=Sf2, VSMID=BOT, VSGRAD=1, VSMAX=1.

Comparison of output files – Difference types					
AB		**AC**		**BC**	
.OUT files	**.DAT files**	**.OUT files**	**.DAT files**	**.OUT files**	**.DAT files**
Balance residuals in volumetric budget (Type I)	Type I	Balance residuals, storage changes and total inflows in volumetric budget (Type I)	Type I	Balance residuals, storage changes and total inflows in volumetric budget (Type I)	Type I

Test Problem 2				
Reference	**Name**	**Description**	**Dataset Root Name**	**MF optional files employed**
US EPA problem 2 [modflmn.pdf, 2-1]	Anisotropy (Part b)	Same as problem 1, with one quadrant of model grid (pumping well in node [1,1]) and anisotropy ratio Ty/Tx of 0.1	pr2b	.WEL .SIP .OPC

Change to input data files		
MODFLOW 96 (A)	**MODFLOW-VKD VKD inactive (B)**	**MODFLOW-VKD VKD active (C)**
None	Eliminate comment on 1st line of .BCF file.	Set BOT = –100 m and TOP = –99 m in order to reproduce strictly confined conditions. Kbase=T, VMID=BOT, VKGRAD=1, VKMAX=1, Sbase=Sf1=Sf2, VSMID=BOT, VSGRAD=1, VSMAX=1.

Comparison of output files – Difference types					
AB		**AC**		**BC**	
.OUT files	**.DAT files**	**.OUT files**	**.DAT files**	**.OUT files**	**.DAT files**
Balance residuals in volumetric budget (Type I)	Type I	Maximum head changes (Type I)	Type I	Balance residuals in volumetric budget and maximum head changes (Type I)	Type I

<table>
<tr><th colspan="6">Test Problem 3</th></tr>
<tr><th>Reference</th><th>Name</th><th>Description</th><th>Dataset Root Name</th><th colspan="2">MF optional files employed</th></tr>
<tr><td>US EPA problem 3 [modflmn.pdf, 3-1]</td><td>Artesian-water table conversion</td><td>Confined-unconfined model with constant T and variable S (LAYCON=2), 1 layer, 26 rows, 26 cols with mesh spacing reduction towards pumping well in node [1,1]. One stress period of 100 days and 25 time steps.</td><td>pr3</td><td colspan="2">.WEL
.SIP
.OPC</td></tr>
<tr><th colspan="6">Change to input data files</th></tr>
<tr><th colspan="2">MODFLOW 96 (A)</th><th colspan="2">MODFLOW-VKD VKD inactive (B)</th><th colspan="2">MODFLOW-VKD VKD active (C)</th></tr>
<tr><td colspan="2">None</td><td colspan="2">None</td><td colspan="2">Not possible to reproduce LAYCON conditions with LAYCON 4 or 5. Run not executed.</td></tr>
<tr><th colspan="6">Comparison of output files – Difference types</th></tr>
<tr><th colspan="2">AB</th><th colspan="2">AC</th><th colspan="2">BC</th></tr>
<tr><th>.OUT files</th><th>.DAT files</th><th>.OUT files</th><th>.DAT files</th><th>.OUT files</th><th>.DAT files</th></tr>
<tr><td>Flow terms in volumetric budget and elapsed times (Type I)</td><td>Type I</td><td>N/A</td><td>N/A</td><td>N/A</td><td>N/A</td></tr>
</table>

Test Problem 4					
Reference	Name	Description	Dataset Root Name	MF optional files employed	
US EPA problem 5 [modflmn.pdf, 5-1]	Mass balance	Confined model, 1 layer, 7 rows, 7 cols, specified heads along row 1 and column 7. One stress period of 365 days and 5 time steps.	pr5	.RCH .SIP .OPC	
Change to input data files					
MODFLOW 96 (A)		**MODFLOW-VKD VKD inactive (B)**		**MODFLOW-VKD VKD active (C)**	
None		Eliminate comment on 1st line of .BCF file.		Set BOT = –100 m and TOP = –99 m in order to reproduce strictly confined conditions. Kbase=T, VMID=BOT, VKGRAD=1, VKMAX=1, Sbase=Sf1=Sf2, VSMID=BOT, VSGRAD=1, VSMAX=1.	
Comparison of output files – Difference types					
AB		**AC**		**BC**	
.OUT files	**.DAT files**	**.OUT files**	**.DAT files**	**.OUT files**	**.DAT files**
Flow terms in volumetric budget (Type I)	Type I	Flow terms in volumetric budget and drawdowns (Type I)	Type I	Flow terms in volumetric budget and drawdowns (Type I)	Type I

<table>
<tr><th colspan="6">Test Problem 5</th></tr>
<tr><th>Reference</th><th>Name</th><th>Description</th><th>Dataset Root Name</th><th colspan="2">MF optional files employed</th></tr>
<tr><td>US EPA problem 8 [modflmn.pdf, 8-1]</td><td>Grid and time stepping considerations (Part a)</td><td>Confined model, 1 layer, 4 rows, 4 cols, specified heads (set with GHB) along edge of one side. One stress period of 20 days and 10 time steps.</td><td>p8a</td><td colspan="2">.WEL
.GHB
.SIP</td></tr>
<tr><th colspan="6">Change to input data files</th></tr>
<tr><th colspan="2">MODFLOW 96
(A)</th><th colspan="2">MODFLOW-VKD
VKD inactive (B)</th><th colspan="2">MODFLOW-VKD
VKD active (C)</th></tr>
<tr><td colspan="2">None</td><td colspan="2">Eliminate comment on 1st line of .BCF file.</td><td colspan="2">Set BOT = –100 m and TOP = –99 m in order to reproduce strictly confined conditions. Kbase=T, VMID=BOT, VKGRAD=1, VKMAX=1, Sbase=Sf1=Sf2, VSMID=BOT, VSGRAD=1, VSMAX=1.</td></tr>
<tr><th colspan="6">Comparison of output files – Difference types</th></tr>
<tr><th colspan="2">AB</th><th colspan="2">AC</th><th colspan="2">BC</th></tr>
<tr><th>.OUT files</th><th>.DAT files</th><th>.OUT files</th><th>.DAT files</th><th>.OUT files</th><th>.DAT files</th></tr>
<tr><td>Flow terms in volumetric budget (Type I)</td><td>Type I</td><td>Flow terms in volumetric budget (Type I)</td><td>Type I</td><td>Flow terms in volumetric budget (Type I)</td><td>Type I</td></tr>
</table>

Test Problem 6				
Reference	**Name**	**Description**	**Dataset Root Name**	**MF optional files employed**
US EPA problem 9 [modflmn.pdf, 9-1]	Calibration and prediction (Part a)	Confined model, 1 layer, 15 rows, 15 cols, with river crossing along west-east direction. Steady state simulation.	p9a,b	.WEL .RIV .RCH .SIP

Change to input data files		
MODFLOW 96 (A)	**MODFLOW-VKD VKD inactive (B)**	**MODFLOW-VKD VKD active (C)**
None	Eliminate comment on 1st line of .BCF file.	Set BOT = –100 m and TOP = –99 m in order to reproduce strictly confined conditions. Kbase=T, VMID=BOT, VKGRAD=1, VKMAX=1.

Comparison of output files – Difference types					
AB		**AC**		**BC**	
.OUT files	**.DAT files**	**.OUT files**	**.DAT files**	**.OUT files**	**.DAT files**
None	Type I	None	Type I	None	None

Test Problem 7

Reference	Name	Description	Dataset Root Name	MF optional files employed
US EPA problem 12 [modflmn.pdf, 12-1]	Leaky aquifers (Part a)	10 confined layers (1 layer for upper aquifer, 7 layers for aquitard, 2 layers for lower aquifer), 25 rows, 25 cols with mesh spacing reduction towards pumping well in nodes [9,1,1] and [10,1,1]. One stress period of 787900 s and 30 time steps.	p12a	.WEL .SIP .OPC

Change to input data files

MODFLOW 96 (A)	MODFLOW-VKD VKD inactive (B)	MODFLOW-VKD VKD active (C)
None	None	Set BOT of 10 layers from -500 to –509 m and TOP from –499 to –508 m in order to reproduce strictly confined conditions. Kbase=T, VMID=BOT, VKGRAD=1, VKMAX=1, Sbase=Sf1=Sf2, VSMID=BOT, VSGRAD=1, VSMAX=1 in all layers.

Comparison of output files – Difference types

AB		AC		BC	
.OUT files	**.DAT files**	**.OUT files**	**.DAT files**	**.OUT files**	**.DAT files**
Flow terms in volumetric budget and elapsed times (Type I)	Type I	Drawdowns, flow terms in volumetric budget and elapsed times (Type I)	Type I	Drawdowns and flow terms in volumetric budget (Type I)	Type I

Test Problem 8

Reference	Name	Description	Dataset Root Name	MF optional files employed
US EPA problem 13 [modflmn.pdf, 13-1]	Solution techniques and convergence (Part a and b)	3 layers (layer 1 unconfined, layer 2 and 3 confined), 10 rows, 15 cols with specified head along east side in layer 1. Steady-state simulation.	p13a,b,c	.RCH .SIP .SOR .OC (created) .PCG (created)

Change to input data files

MODFLOW 96 (A)	MODFLOW-VKD VKD inactive (B)	MODFLOW-VKD VKD active (C)
Increase MXITER in .SIP for sip solver to achieve convergence.	Increase MXITER in .SIP for sip solver to achieve convergence. Eliminate comment on 1st line of .BCF file.	Increase MXITER in .SIP for sip solver to achieve convergence. Set BOT of layer 2 and 3 from -501 to –502 m and TOP of layer 2 and 3 from –500 to –501 m in order to reproduce strictly confined conditions. Kbase=K in layer 1, Kbase=T in layer 2 and 3, VMID=BOT, VKGRAD=1, VKMAX=1 in all layers.

Comparison of output files – Difference types

AB		AC		BC	
.OUT files	**.DAT files**	**.OUT files**	**.DAT files**	**.OUT files**	**.DAT files**
None	Type I	None	Type I	None	Type I

Note:
This problem was run using both the SIP and PCG2 solvers with essentially identical results being obtained

<table>
<tr><th colspan="6">Test Problem 9</th></tr>
<tr><th>Reference</th><th>Name</th><th>Description</th><th>Dataset Root Name</th><th colspan="2">MF optional files employed</th></tr>
<tr><td>US EPA problem 14 [modflmn.pdf, 14-1]</td><td>Head dependent boundary conditions (Part a, c and d considered at once)</td><td>Unconfined model, 1 layer, 7 rows, 7 cols with uniform mesh spacing, pumping well in node [1,1]. Set a river, drains and evt along col 4. One stress period of 365 days and 20 time steps.</td><td>p14</td><td colspan="2">.WEL
.DRN
.RIV
.ET
.SIP
.OPC</td></tr>
<tr><th colspan="6">Change to input data files</th></tr>
<tr><th colspan="2">MODFLOW 96
(A)</th><th colspan="2">MODFLOW-VKD
VKD inactive (B)</th><th colspan="2">MODFLOW-VKD
VKD active (C)</th></tr>
<tr><td colspan="2">None</td><td colspan="2">Eliminate comment on 1st line of .BCF file.</td><td colspan="2">Set Kbase=K, VMID=BOT, VKGRAD=1, VKMAX=1, Sbase=Sf1, VSMID=BOT, VSGRAD=1, VSMAX=1.</td></tr>
<tr><th colspan="6">Comparison of output files – Difference types</th></tr>
<tr><th colspan="2">AB</th><th colspan="2">AC</th><th colspan="2">BC</th></tr>
<tr><th>.OUT files</th><th>.DAT files</th><th>.OUT files</th><th>.DAT files</th><th>.OUT files</th><th>.DAT files</th></tr>
<tr><td>Flow terms in volumetric budget (Type I)</td><td>Type I</td><td>Flow terms in volumetric budget (Type I)</td><td>Type I</td><td>Flow terms in volumetric budget, maximum head changes and heads (Type I)</td><td>Type I</td></tr>
</table>

Test Problem 10

Reference	Name	Description	Dataset Root Name	MF optional files employed
US EPA problem 16 [modflmn.pdf, 16-1]	Evapotranspiration (Part a)	Unconfined model, 1 layer, 20 rows, 18 cols with specified heads in zone to the east. One stress period of 1 day and 1 time step.	p16a	.ET .RCH .SIP .OPC (created)

Change to input data files

MODFLOW 96 (A)	MODFLOW-VKD VKD inactive (B)	MODFLOW-VKD VKD active (C)
None	Eliminate comment on 1st line of .BCF file.	Set Kbase=K, VMID=BOT, VKGRAD=1, VKMAX=1, Sbase=Sf1, VSMID=BOT, VSGRAD=1, VSMAX=1.

Comparison of output files – Difference types

AB		AC		BC	
.OUT files	.DAT files	.OUT files	.DAT files	.OUT files	.DAT files
None	Type I	None	Type I	None	None

<table>
<tr><th colspan="9">Test Problem 11</th></tr>
<tr><th colspan="2">Reference</th><th colspan="2">Name</th><th colspan="2">Description</th><th colspan="2">Dataset Root Name</th><th>MF optional files employed</th></tr>
<tr><td colspan="2">US EPA problem 18 [modflmn.pdf, 18-1]</td><td colspan="2">Cross-sectional simulations (Part b)</td><td colspan="2">Vertical section (1 row), 6 layers (layer 1 is unconfined, other layers are convertible), 27 cols, specified head in layer 1, 2 and 3 reproduce river, spec. heads in layer 6 reproduce leakage. Steady-state simulation.</td><td colspan="2">p18</td><td>.RCH
.SOR
.OPC</td></tr>
<tr><th colspan="9">Change to input data files</th></tr>
<tr><th colspan="3">MODFLOW 96
(A)</th><th colspan="2">MODFLOW-VKD
VKD inactive (B)</th><th colspan="4">MODFLOW-VKD
VKD active (C)</th></tr>
<tr><td colspan="3">None</td><td colspan="2">Eliminate comment on 1st line of .BCF file.</td><td colspan="4">Set Kbase=K, VMID=BOT,VKGRAD=1,VKMAX=1 in all layers.</td></tr>
<tr><th colspan="9">Comparison of output files – Difference types</th></tr>
<tr><th colspan="3">AB</th><th colspan="2">AC</th><th colspan="4">BC</th></tr>
<tr><th>.OUT files</th><th colspan="2">.DAT files</th><th>.OUT files</th><th>.DAT files</th><th colspan="2">.OUT files</th><th colspan="2">.DAT files</th></tr>
<tr><td>None</td><td colspan="2">None</td><td>Constant head flows in volumetric budget (Type I)</td><td>None</td><td colspan="2">None</td><td colspan="2">None</td></tr>
</table>

Test Problem 12				
Reference	**Name**	**Description**	**Dataset Root Name**	**MF optional files employed**
US EPA problem 20 [modflmn.pdf, 20-1]	Application to a hazardous waste site (Part a, b and d)	Unconfined model, 1 layer, 39 rows, 37 cols with mesh spacing reduction in landfill site, a drain is present. Steady-state simulation to produce the initial heads for a time-variant simulation with one stress period of 30.8 years and 20 time steps.	p20a,b,d,ba,bb,b,bda,bdb	.DRN .RCH .SIP .PCG (created) .OPC

Change to input data files		
MODFLOW 96 (A)	**MODFLOW-VKD VKD inactive (B)**	**MODFLOW-VKD VKD active (C)**
Copy head matrix generated in the steady-state simulation into the .BAS file used in the time-variant run.	Copy head matrix generated in the steady-state simulation into the .BAS file used in the time-variant run.	Copy head matrix generated in the steady-state simulation into the .BAS file used in the time-variant run. Set Kbase=K, VMID=BOT, VKGRAD=1,VKMAX=1 and Sbase=Sf1, VSMID=BOT, VSGRAD=1, VSMAX=1 in time-variant run.

Comparison of output files – Difference types											
AB				**AC**				**BC**			
.OUT files		**.DAT files**		**.OUT files**		**.DAT files**		**.OUT files**		**.DAT files**	
SS	TV	SS	TV	SS	TV	SS	TV	SS	TV	SS	TV
None	Flow terms in volumetric budget (Type I)	Type I	Type I	None	Flow terms in volumetric budget, maximum head changes, drawdowns and heads (Type I)	Type I	Type I	None	Flow terms in volumetric budget, maximum head changes, drawdowns and heads (Type I)	None	Type I

Note:
This problem was run using both the SIP and PCG2 solvers with essentially identical results being obtained

Test Problem 13

Reference	Name	Description	Dataset Root Name	MF optional files employed
USGS problem 1 [TWRI Book 6, Chapter A2]	Storage depletion (use of inter-bed storage, IBS1, package, which simulates storage changes and compaction in fine-grained beds and confining beds. Used to reproduce aquifer compaction and land subsidence)	2 confined layers (layer 1 has interbed storage), 10 rows, 12 cols. Three stress periods of 1000 days and 10 time steps.	IBS	.SIP .OC .IBS

Change to input data files

MODFLOW 96 (A)	MODFLOW-VKD VKD inactive (B)	MODFLOW-VKD VKD active (C)	MODFLOW-VKD (Compaq compiled) VKD active (C1)
Added comments in .IBS file	None	Set BOT of layer 1 and 2 from -501 to –502 m and TOP of layer 2 and 3 from –500 to –501 m in order to reproduce strictly confined conditions. Kbase=T, VMID=BOT, VKGRAD=1, VKMAX=1, Sbase=Sf1=Sf2, VSMID=BOT, VSGRAD=1, VSMAX=1 in both layers.	Set BOT of layer 1 and 2 from -501 to –502 m and TOP of layer 2 and 3 from –500 to –501 m in order to reproduce strictly confined conditions. Kbase=T, VMID=BOT, VKGRAD=1, VKMAX=1, Sbase=Sf1=Sf2, VSMID=BOT, VSGRAD=1, VSMAX=1 in both layers. Replace tabs with spaces in IBS.IBS file

Comparison of output files – Difference types

AB		AC		BC		CC1	
.OUT files	.DAT files	.OUT files	.DAT files	.OUT files	.DAT files	.OUT files	.DAT files
Flow terms in volumetric budget (Type I)	Type I	Flow terms in volumetric budget and maximum head changes (Type I)	Type I	Flow terms in volumetric budget and maximum head changes (Type I)	Type I	None	Type I

Test Problem 14				
Reference	**Name**	**Description**	**Dataset Root Name**	**MF optional files employed**
USGS problem 2 [OFR 88-729]	Example problem in OFR 88-729 (use of stream, STR, package)	Confined model, 1 layer, 6 rows, 6 cols, a stream crosses the modelled area. One stress period of 15 days and 3 time steps.	STR	.SIP .OC .STR

Change to input data files			
MODFLOW 96 (A)	**MODFLOW-VKD VKD inactive (B)**	**MODFLOW-VKD VKD active (C)**	**MODFLOW-VKD (Compaq compiled) VKD active (C1)**
None	None	Set BOT = -501 m and TOP = –500 m, in order to reproduce strictly confined conditions. Kbase=T, VMID=BOT, VKGRAD=1, VKMAX=1, Sbase=Sf1=Sf2, VSMID=BOT, VSGRAD=1, VSMAX=1.	Set BOT = -501 m and TOP = –500 m, in order to reproduce strictly confined conditions. Kbase=T, VMID=BOT, VKGRAD=1, VKMAX=1, Sbase=Sf1=Sf2, VSMID=BOT, VSGRAD=1, VSMAX=1.

Comparison of output files – Difference types							
AB		**AC**		**BC**		**CC1**	
.OUT files	**.DAT files**	**.OUT files**	**.DAT files**	**.OUT files**	**.DAT files**	**.OUT files**	**.DAT files**
Flow terms in volumetric budget, heads and drawdowns (Type I)	Type II At time step 3, stream outflow [m^3/day] at nodes: [2,3]. A=5.11, B=2.27, %diff=77 [5,3]. A=1.79, B=0.99, %diff=58 [5,4]. A=9.20, B=5.25, % diff=55. See Note	Flow terms in volumetric budget, heads, drawdowns and maximum head changes (Type I)	Type II* At time step 3, stream outflow [m^3/day] at nodes: [2,3]. A=5.11, B=2.27, %diff=77 [5,3]. A=1.79, B=0.99, % diff=58 [5,4]. A=9.20, B=5.25, %diff=55. See Note	None	None	None	Type I
Note: A discrepancy in stream outflows is expected in cells where more than one reach is defined, as a correction in the calculation of the accreted streamflow was introduced in the MF-VKD code. This is explained in detail in Section 3.2.3 of MF-VKD User Guide (EA, 2003).							

<table>
<tr><th colspan="12">Test Problem 15</th></tr>
<tr><th colspan="3">Reference</th><th colspan="3">Name</th><th colspan="3">Description</th><th colspan="2">Dataset Root Name</th><th>MF optional files employed</th></tr>
<tr><td colspan="3">USGS problem 3 [TWRI Book 6, Chapter A1, Appendix D]</td><td colspan="3">Example problem in TWRI 6-A1</td><td colspan="3">3 layers (layer 1 unconfined, layer 2 and 3 confined), 15 rows, 15 cols, specified heads along one side in layer 1 and 2, drains and pumping wells are present. Steady-state simulation.</td><td colspan="2">TWRI</td><td>.WEL
.DRN
.RCH
.SIP
.OC (created)</td></tr>
<tr><th colspan="12">Change to input data files</th></tr>
<tr><th colspan="4">MODFLOW 96
(A)</th><th colspan="4">MODFLOW-VKD
VKD inactive (B)</th><th colspan="4">MODFLOW-VKD
VKD active (C)</th></tr>
<tr><td colspan="4">None</td><td colspan="4">Eliminate comment on 1st line of .BCF file.</td><td colspan="4">Set BOT of layer 2 and 3 from -901 to –902 m and TOP of layer 2 and 3 from –900 to –901 m in order to reproduce strictly confined conditions. Kbase=K in layer 1, Kbase=T in layer 2 and 3, VMID=BOT, VKGRAD=1, VKMAX=1 in all layers.</td></tr>
<tr><th colspan="12">Comparison of output files – Difference types</th></tr>
<tr><th colspan="4">AB</th><th colspan="4">AC</th><th colspan="4">BC</th></tr>
<tr><th colspan="2">.OUT files</th><th colspan="2">.DAT files</th><th colspan="2">.OUT files</th><th colspan="2">.DAT files</th><th colspan="2">.OUT files</th><th colspan="2">.DAT files</th></tr>
<tr><td colspan="2">None</td><td colspan="2">Type I</td><td colspan="2">None</td><td colspan="2">Type I</td><td colspan="2">None</td><td colspan="2">Type I</td></tr>
</table>

<table>
<tr><th colspan="6">Test Problem 16</th></tr>
<tr><th>Reference</th><th>Name</th><th>Description</th><th>Dataset Root Name</th><th colspan="2">MF optional files employed</th></tr>
<tr><td>USGS problem 4 [OFR 91-536, not available on-line]</td><td>Example problem in OFR 91-536</td><td>2 layers (layer 1 unconfined, layer 2 confined), 10 rows, 15 cols, a river and two pumping wells are present. Two stress periods of 1 day and 1 time step each.</td><td>BCF2SS</td><td colspan="2">.WEL
.RIV
.RCH
.PCG
.OC</td></tr>
<tr><th colspan="6">Change to input data files</th></tr>
<tr><th colspan="2">MODFLOW 96
(A)</th><th colspan="2">MODFLOW-VKD
VKD inactive (B)</th><th colspan="2">MODFLOW-VKD
VKD active (C)</th></tr>
<tr><td colspan="2">None</td><td colspan="2">None</td><td colspan="2">Set BOT of layer 2 = -901 m and TOP of layer 2 = –900 m in order to reproduce strictly confined conditions. Kbase=K in layer 1, Kbase=T in layer 2, VMID=BOT, VKGRAD=1, VKMAX=1, Sbase=Sf1 in layer 1, Sbase=Sf1=Sf2 in layer 2, VSMID=BOT, VSGRAD=1, VSMAX=1 in both layers.</td></tr>
<tr><th colspan="6">Comparison of output files – Difference types</th></tr>
<tr><th colspan="2">AB</th><th colspan="2">AC</th><th colspan="2">BC</th></tr>
<tr><th>.OUT files</th><th>.DAT files</th><th>.OUT files</th><th>.DAT files</th><th>.OUT files</th><th>.DAT files</th></tr>
<tr><td>None</td><td>Type I</td><td>None</td><td>Type I</td><td>None</td><td>None</td></tr>
</table>

Test Problem 17				
Reference	**Name**	**Description**	**Dataset Root Name**	**MF optional files employed**
USGS problem 5 [OFR 94-59]	Example problem 1 in OFR 94-59 (use of transient leakage, TLK1, package to simulate storage depletion in confining beds)	Vertical section (1 row), 4 layers (all layers confined), 13 cols, specified heads along model edges, storage of confining layers is reproduced. One stress period of 100 days and 40 time steps.	TLKP1	.TLK .SIP .OC

Change to input data files		
MODFLOW 96 (A)	**MODFLOW-VKD VKD inactive (B)**	**MODFLOW-VKD VKD active (C)**
None	None	Set BOT of layers from –901 m to –904 m and TOP from –900 m to –903 m in order to reproduce strictly confined conditions. Kbase=T, VMID=BOT, VKGRAD=1, VKMAX=1, Sbase=Sf1=Sf2, VSMID=BOT, VSGRAD=1, VSMAX=1 in all layers.

Comparison of output files – Difference types					
AB		**AC**		**BC**	
.OUT files	**.DAT files**	**.OUT files**	**.DAT files**	**.OUT files**	**.DAT files**
Flow terms in volumetric budget, (Type I)	Type I	Flow terms in volumetric budget and maximum head changes (Type I)	Type I	Flow terms in volumetric budget and maximum head changes (Type I)	Type I

Test Problem 18				
Reference	**Name**	**Description**	**Dataset Root Name**	**MF optional files employed**
USGS problem 6 [OFR 96-364]	Example problem in OFR 96-364 (use of reservoir, RES1, package to simulate leakage from a reservoir)	Confined model, 1 layer, 12 rows, 12 cols, general-head boundaries along the model edges, a reservoir is present. Three stress periods of 5 days and 5 time steps each.	RESTEST	.GHB .SIP. .OC .RES

Change to input data files		
MODFLOW 96 (A)	**MODFLOW-VKD VKD inactive (B)**	**MODFLOW-VKD VKD active (C)**
None	None	Set BOT = –901 m and TOP = –900 m in order to reproduce strictly confined conditions. Kbase=T, VMID=BOT, VKGRAD=1, VKMAX=1, Sbase=Sf1=Sf2, VSMID=BOT, VSGRAD=1, VSMAX=1.

Comparison of output files – Difference types					
AB		**AC**		**BC**	
.OUT files	**.DAT files**	**.OUT files**	**.DAT files**	**.OUT files**	**.DAT files**
Storage flow terms in volumetric budget, (Type I)	Type I	Storage flow terms in volumetric budget, (Type I)	Type I	Flow terms in volumetric budget, (Type I)	Type I

<table>
<tr><th colspan="6">Test Problem 19</th></tr>
<tr><th>Reference</th><th>Name</th><th>Description</th><th>Dataset Root Name</th><th colspan="2">MF optional files employed</th></tr>
<tr><td>USGS problem 7 [OFR 97-571]</td><td>Example problem in OFR 97-571 (use of transient specified flow and specified head boundary, FHB1, package)</td><td>1 layer, 3 rows, 10 cols. Three stress periods of 400, 200, 400 days and 10, 4, 6 time steps respectively.</td><td>FHB</td><td colspan="2">.SIP
.OC
.FHB</td></tr>
<tr><th colspan="6">Change to input data files</th></tr>
<tr><th colspan="2">MODFLOW 96 (A)</th><th colspan="2">MODFLOW-VKD VKD inactive (B)</th><th colspan="2">MODFLOW-VKD VKD active (C)</th></tr>
<tr><td colspan="2">None</td><td colspan="2">None</td><td colspan="2">Set BOT = –901 m and TOP = –900 m in order to reproduce strictly confined conditions. Kbase=T, VMID=BOT, VKGRAD=1, VKMAX=1, Sbase=Sf1=Sf2, VSMID=BOT, VSGRAD=1, VSMAX=1.</td></tr>
<tr><th colspan="6">Comparison of output files – Difference types</th></tr>
<tr><th colspan="2">AB</th><th colspan="2">AC</th><th colspan="2">BC</th></tr>
<tr><th>.OUT files</th><th>.DAT files</th><th>.OUT files</th><th>.DAT files</th><th>.OUT files</th><th>.DAT files</th></tr>
<tr><td>Flow terms in volumetric budget (Type I)</td><td>Type I</td><td>Flow terms in volumetric budget (Type I)</td><td>Type I</td><td>Flow terms in volumetric budget (Type I)</td><td>Type I</td></tr>
</table>

Test Problem 20				
Reference	**Name**	**Description**	**Dataset Root Name**	**MF optional files employed**
Examination of Groundwater Flow Model Sensitivity Analysis and Parameter Optimisation using UCODE, NGWCLC Project NC/01/60, Chapter 4	Wirral regional model	1 layer (convertible with variable T), 122 rows, 90 cols, river cells located along the coast to limit inflow, pumping wells with large abstraction. 348 stress periods of one-month and 1 time step each.	wirral117tv	.RCH .WEL .RIV .STR (dummy) .OC .PCG

Change to input data files		
MODFLOW 96 (A)	**MODFLOW-VKD VKD inactive (B)**	**MODFLOW-VKD VKD active (C)**
None	None	Set Kbase=K, VMID=BOT, VKGRAD=1, VKMAX=1, Sbase=Sf1=Sf2, VSMID=BOT, VSGRAD=1, VSMAX=1.

Comparison of output files – Difference types					
AB		**AC**		**BC**	
.OUT files	**.DAT files**	**.OUT files**	**.DAT files**	**.OUT files**	**.DAT files**
Flow terms in volumetric budget, heads, drawdowns and maximum head changes (Type I)	Type I	Flow terms in volumetric budget, heads, drawdowns and maximum head changes (Type I)	TypeI	Flow terms in volumetric budget and maximum head changes (Type I)	Type I

Note: Cell-by-cell flows, heads and drawdowns are printed only in 1st and last stress period in order to reduce the size of the .OUT and .CBB files and allow comparison with UltraEdit-32©.

<table>
<tr><th colspan="5">Test Problem 21</th></tr>
<tr><th>Reference</th><th>Name</th><th>Description</th><th>Dataset Root Name</th><th>MF optional files employed</th></tr>
<tr><td>Tadcaster Groundwater Protection zones. WMC Report 1345/R1.</td><td>Tadcaster GPZ flow and particle tracking model</td><td>1 unconfined layer, 148 rows, 194 cols, general-head boundaries along Northeast edge representing lateral continuation of aquifer, river cells representing the River Wharfe, Bramham Beck and Cock Beck, pumping wells with large abstraction. Steady-state simulation.</td><td>Tad401 (Modflow files)
Tad401p (Modpath files)</td><td>.WEL
.RCH
.RIV
.GHB
.OC
.PCG</td></tr>
</table>

<table>
<tr><th colspan="4">Change to input data files</th></tr>
<tr><th>MODFLOW 96 (A)</th><th>MODFLOW-VKD VKD inactive (B)</th><th>MODFLOW-VKD VKD active (C)</th><th>MODFLOW-VKD VKD active, variable K (D)</th></tr>
<tr><td>None</td><td>None</td><td>Set Kbase=K, VMID=BOT, VKGRAD=1, VKMAX=1.</td><td>Using the auto-conversion option, set Kbase=K, thickness of upper zone = 5, VKGRAD=1, VKMAX=3.</td></tr>
</table>

<table>
<tr><th colspan="3">Comparison of output files – Difference types</th></tr>
<tr><td rowspan="3">AB</td><td>.OUT files</td><td>None</td></tr>
<tr><td>.DAT files</td><td>Type I</td></tr>
<tr><td>.PTL files</td><td>Type I</td></tr>
<tr><td rowspan="3">AC</td><td>.OUT files</td><td>None</td></tr>
<tr><td>.DAT files</td><td>Type I</td></tr>
<tr><td>.PTL files</td><td>Type I</td></tr>
<tr><td rowspan="3">AD</td><td>.OUT files</td><td>Flow terms in volumetric budget, heads and drawdowns (Type I)</td></tr>
<tr><td>.DAT files</td><td>Max head diff = -0.018 m, Max cbc diff = -0.0367 m^3/day (Type I)</td></tr>
<tr><td>.PTL files</td><td>Max travel time diff = 5.4 %, Max diff in X, Y and Z coord = 92.11, 37.74 and 3.3 m respectively.</td></tr>
<tr><td rowspan="3">BC</td><td>.OUT files</td><td>None</td></tr>
<tr><td>.DAT files</td><td>None</td></tr>
<tr><td>.PTL files</td><td>None</td></tr>
<tr><td rowspan="3">BD</td><td>.OUT files</td><td>Flow terms in volumetric budget, heads and drawdowns (Type I)</td></tr>
<tr><td>.DAT files</td><td>Max head diff = -0.018 m, Max cbc diff = -0.0367 m^3/day (Type I)</td></tr>
<tr><td>.PTL files</td><td>Type I</td></tr>
<tr><td rowspan="3">CD</td><td>.OUT files</td><td>Flow terms in volumetric budget, heads and drawdowns (Type I)</td></tr>
<tr><td>.DAT files</td><td>Max head diff = -0.018 m, Max cbc diff = -0.0367 m^3/day (Type I)</td></tr>
<tr><td>.PTL files</td><td>Type I</td></tr>
</table>

APPENDIX B

Contents of the CD included in the revised version of the User Guide for MODFLOW-VKD

The directory structure of the deliverables related to the project VKD further testing - Phase 1 (Groundwater Technical Services Contract #11589) is shown below. This structure is also detailed in the file `readme_deliverables.txt`.

> Reports & Documentation: directory containing the updated User Guide and all MODFLOW-VKD reports in PDF format

> Software: directory containing the programs MF96.exe, MF-VKD6.exe (including source code), MF-VKD6-Compaq.exe (compiled with the Compaq FOTRAN compiler) and the utilities heads2gv.exe, cbc2gv.exe, cbc2dos.exe employed to compare the .HDS and .CBB files produced in the tests

> Example datasets from further testing:
> USEPA_Tests: directory containing the EPA instructional problems for MODFLOW used for VKD testing. The I/O files for each problem are stored in separate sub-directories. Each sub-directory is named according to the problem number and includes the following:

- comparison files (.TXT) for the .OUT files and the ASCII version (.DAT) of the .CBB files
- comparison spreadsheets (.XLS) for the ASCII version (.DAT) of the .CBB files, for the tests where comparison files reveal several discrepancies (problems 1,2,12,14,20TV)

> Mf96(A): directory containing the I/O files of the test performed with MF96
> vkd_inact(B): directory containing the I/O files of the test performed with MF-VKD (VKD function inactive)
> vkd_act(C): directory containing the I/O files of the test performed with MF-VKD (VKD function active)

> USGS_Tests: directory containing the USGS MODFLOW tests used for VKD testing. The I/O files for each problem are stored in separate sub-directories. Each sub-directory is named according to the problem number and includes the following:

- comparison files (.TXT) for the .OUT files and the ASCII version (.DAT) of the .CBB files
- comparison spreadsheets (.XLS) for the ASCII version (.DAT) of the .CBB files, for the tests where comparison files reveal several discrepancies (problems 1,2,5)

> Mf96(A): directory containing the I/O files of the test performed with MF96
> vkd_inact(B): directory containing the I/O files of the test performed with MF-VKD (VKD function inactive)
> vkd_act(C): directory containing the I/O files of the test performed with MF-VKD (VKD function active)
> vkd_act_Compaq(C1) [in problems 1 and 2]: directory containing the I/O files of the test performed with MF-VKD compiled with the FORTRAN Compaq compiler (VKD function active)

> Wirral_model: directory containing I/O files for the Wirral MODFLOW model ('wirral117tv' root name). This directory contains the following:

- comparison files (.TXT) for the .OUT files and the ASCII version (.DAT) of the .CBB files
- comparison spreadsheets (.XLS) for the ASCII version (.DAT) of the .HDS and .CBB files

> Mf96(A): directory containing the I/O files of the test performed with MF96
> vkd_inact(B): directory containing the I/O files of the test performed with MF-VKD (VKD function inactive)
> vkd_act(C): directory containing the I/O files of the test performed with MF-VKD (VKD function active)

> Tadcaster_model: directory containing I/O files for the MODFLOW and MODPATH Tadcaster GPZ model ('tad401' root name). This directory contains the following:

- comparison files (.TXT) for the .OUT files and the ASCII version (.DAT) of the .CBB files
- comparison spreadsheets (.XLS) for the ASCII version (.DAT) of the .HDS, .CBB files and for the .PTL files

> Mf96(A): directory containing the I/O files of the test performed with MF96 and MODPATH

> vkd_inact(B): directory containing the I/O files of the test performed with MF-VKD (VKD function inactive) and MODPATH

> vkd_act(C): directory containing the I/O files of the test performed with MF-VKD (VKD function active) and MODPATH

> vkd_act_Teq(D): directory containing the I/O files of the test performed with MF-VKD (VKD function active and depth-dependent K) and MODPATH

> Additional documentation: directory containing the reports that describe some of the tests

> Spreadsheets and utilities developed by various consultants on different projects where MODFLOW-VKD has been used

APPENDIX G
Independent review of the MODFLOW VKD code

INDEPENDENT REVIEW OF THE MODFLOW VKD CODE

David Lerner, Groundwater Protection and Restoration Group, University of Sheffield, July 2003

SUMMARY

The code development and testing have been well carried out, and the code should be released for use in flow modelling. I have three recommendations:

1) The results of the use of the code in further field situations are reviewed.

2) Further attempts are made to understand the causes of the difficulties with the particle tracking when VKD is present but not operational. However efforts to improve MODPATH should be concentrated on issues of high significance to the Environment Agency, such as fractured rocks.

3) Contact is made with the code custodians, as it would be useful to have VKD recognised as an official variant on MODFLOW, and achieve forward compatibility.

INTRODUCTION

This is my response to The Environment Agency's request (email from Sarah Evers 28th Mar 03 and EA order 032029353 of 16th June 03) to review several reports on the above project. I have the following documents:

a. Representation of the variation of hydraulic conductivity with saturated thickness in MODFLOW. Stages I & II - Code changes and testing against Birmingham University code (NC/99/67, August 1999, ISBN 0 857 051947). Paper and electronic copies of report, electronic copy of appendices.

b. Representation of the variation of hydraulic conductivity with saturated thickness in MODFLOW. Stage III - Investigation using South West Chilterns model. (NC/99/67, July 2000, ISBN 1 857 05360 5). Paper copy received late June.

c. Enhancements to MODFLOW - Variations in hydraulic conductivity and storage with depth. (NC/00/23, March 2002, ISBN 1 857 05765 1). Part copy received late June, full copy received late July.

d. Enhancements to MODFLOW - User guide for MODFLOW-VKD - A modified version of MODFLOW-96 to include variations in hydraulic properties with depth . (NC/00/23, March 2002, ISBN 1 857 05766 X). Paper copy received late June.

e. Further Testing of the MODFLOW VKD – Draft version (NC/XXXXX, April 2003, in preparation). Electronic copy.

f. Responses to a draft review from Sarah Evers (EA) and Phil Hayes (WMC).

Objectives of the review

The objective of the review, as specified in the tender documents, was "To provide independent review of the MODFLOW VKD code and of the further testing of the MODFLOW VKD code". (A second objective "To test VKD functionality in a way which is appropriate" was excluded from the tender.) I have considered the following questions in reviewing the documents:

- Has the approach adopted been logical and stepwise?
- Has sufficient care been taken in checking the results?
- Does the evidence show that the code performs as required?
- Were the test cases suitable?
- Should any further test be conducted?
- Any other comments.

REVIEW

Has the approach adopted been logical and stepwise?

Yes. The changes to the code were made in two major batches (Reports A and C). Within each batch, the changes were made in increments, tested individually, and the changes and tests were well documented.

Has sufficient care been taken in checking the results?

Yes. The checking was sufficiently detailed to identify errors in the original BHM code, and to spot oddities in the data in the SW Chilterns model. The consultants pursued the detail of any differences between models or unexpected results until they either had an explanation or made corrections.

Does the evidence show that the code performs as required?

Yes. The VKD code reproduces the results of the BHM code, which was the initial objective. It also is compatible with the unchanged MODFLOW.

The VSD option was found to make the code less stable, and there were difficulties in finding solutions with the complex multi-layer case. These are not surprising, all groundwater codes have some scenarios which are very difficult to find solutions for, and they need to be nursed through. The issues have been clearly identified in the documentation.

Problems were found with matching particle tracking results. I agree this is probably due to rounding errors, but this has not been proven. If true, it shows that particle tracking is very sensitive to small changes in parameter values, especially for time-of-travel-zones, and the resulting uncertainties should be recognised.

Were the test cases suitable?

The basic tests in A were OK, that is fairly simple but reasonable first steps. The tests against the USGS and field-based examples in E were sensible choices for checking conformity with the original MODFLOW. The results were adequately summarised.

Only two examples of the code's use with real field problems were presented, the SW Chilterns and Tadcaster models. The Chilterns case was used to demonstrate the VKD capability, and no attempt was made to re-calibrate the model. It could be argued that the VKD model was a poorer fit than the original, particularly on streamflow. The example demonstrates that the code works, but does not make a case that it was useful; only the former was intended. The Tadcaster case was principally used to test MODPATH, and was suitable for the purpose.

Four other field cases were modelled (Minram, Itchen, Bourne, Kennet). These will be reviewed separately, based on presentations to be made at a seminar in October 2003.

Overall, a sufficient and suitable range of test cases has been used.

Should any further tests be conducted?

No. I can see no obvious issues to tackle, except for the particle tracking problem. The next stage is to use the code in earnest on real field problems, demonstrate that the underlying conceptual model works well, and that the code is stable. The modellers will need to aware, as always, that unexpected errors could arise, and should build in some resources for careful checking of results. See below on the particle tracking issue.

Any other comments.

There are a few typos and missing references in the reports, but no major errors were found. Report E was still in draft.

Some problems outside the remit of the project (e.g. stream package) were detected and corrected. Some useful diagnostic outputs were added.

There is no evidence of contact with the code custodians, as recommended in Report A. I recommend this done, as it would be useful to have this recognised as an official variant on MODFLOW, and help to achieve forward compatibility.

Chapter 8 of the User Guide is important, as it gives help in how to approach a modelling problem rather than just dealing with the mechanics of data entry.

The differences encountered with particle tracking require some investigation, although there may not be any solution. If they are due to rounding errors, then remember that there are approximations in the way MODPATH handles velocity interpolation, which are at least as significant and no further action is needed. If changes are needed to the code to handle vertical variations in velocity, then the proposed approach in Report E is too simple, and work will need to start from a clear conceptual model of the hydrogeology and it's representation in the code. For example, how is velocity handled in a cell with a well, where the vertical elevation of the exit point at the well is unknown? However, it is the EA's intention to concentrate resources on other issues related to SPZs (fractured rock, recharge. These are significant issues that should take precedence.

APPENDIX H
Summary of MODFLOW VKD version history

MF-VKD1.exe (Oct 2001)	Previous release of software
MF-VKD2.exe (Jan 2003)	Fixed bug so that confined storage values were used when heads rise above the top (when LAYCON = 5)
MF-VKD3.exe (May 2003)	Allow VKGRAD and VKMAX to be input as absolute values (rather than just as multiples of Kbase)
MF-VKD4.exe (May 2003)	Allow output of calculated internodal transmissivity values to a binary output file (rather than just as text to the listing file)
MF-VKD5.exe (May 2003)	Allow VSGRAD and VSMAX to be input as absolute values (rather than just as multiples of Kbase)
MF-VKD6.exe (May 2003)	Change to use double precision variables (including binary output) (makes VSD calculations more likely to converge)
MF-VKD7.exe (Jun 2003)	Fixed bug so that zero values entered for K or S zones would not produce a divide by zero error when specifying absolute values for VKGRAD, etc.