British
Geological Survey
NATURAL ENVIRONMENT RESEARCH COUNCIL

User's manual for ZETUP, the set up program for the groundwater flow model ZOOMQ3D

Groundwater Systems & Water Quality Programme

Internal Report IR/04/139

BRITISH GEOLOGICAL SURVEY

GROUNDWATER SYSTEMS & WATER QUALITY PROGRAMME

INTERNAL REPORT IR/04/139

User's manual for ZETUP, the set up program for the groundwater flow model ZOOMQ3D

C.R. Jackson[1] and A.E.F. Spink[2]

Keywords

Groundwater flow; ZETUP; ZOOMQ3D.

Bibliographical reference

JACKSON, C.R. AND SPINK, A.E.F. 2004. User's manual for ZETUP, the set up program for the groundwater flow model ZOOMQ3D. *British Geological Survey Internal Report*, IR/04/139. 34pp.

1 British Geological Survey

2 University of Birmingham

Keyworth, Nottingham British Geological Survey 2004

BRITISH GEOLOGICAL SURVEY

The full range of Survey publications is available from the BGS Sales Desks at Nottingham, Edinburgh and London; see contact details below or shop online at www.geologyshop.com

The London Information Office also maintains a reference collection of BGS publications including maps for consultation.

The Survey publishes an annual catalogue of its maps and other publications; this catalogue is available from any of the BGS Sales Desks.

The British Geological Survey carries out the geological survey of Great Britain and Northern Ireland (the latter as an agency service for the government of Northern Ireland), and of the surrounding continental shelf, as well as its basic research projects. It also undertakes programmes of British technical aid in geology in developing countries as arranged by the Department for International Development and other agencies.

The British Geological Survey is a component body of the Natural Environment Research Council.

British Geological Survey offices

Keyworth, Nottingham NG12 5GG

☎ 0115-936 3241 Fax 0115-936 3488
e-mail: sales@bgs.ac.uk
www.bgs.ac.uk
Shop online at: www.geologyshop.com

Murchison House, West Mains Road, Edinburgh EH9 3LA

☎ 0131-667 1000 Fax 0131-668 2683
e-mail: scotsales@bgs.ac.uk

London Information Office at the Natural History Museum (Earth Galleries), Exhibition Road, South Kensington, London SW7 2DE

☎ 020-7589 4090 Fax 020-7584 8270
☎ 020-7942 5344/45 email: bgslondon@bgs.ac.uk

Forde House, Park Five Business Centre, Harrier Way, Sowton, Exeter, Devon EX2 7HU

☎ 01392-445271 Fax 01392-445371

Geological Survey of Northern Ireland, 20 College Gardens, Belfast BT9 6BS

☎ 028-9066 6595 Fax 028-9066 2835

Maclean Building, Crowmarsh Gifford, Wallingford, Oxfordshire OX10 8BB

☎ 01491-838800 Fax 01491-692345

Sophia House, 28 Cathedral Road, Cardiff, CF11 9LJ

☎ 029–2066 0147 Fax 029–2066 0159

Parent Body

Natural Environment Research Council, Polaris House, North Star Avenue, Swindon, Wiltshire SN2 1EU

☎ 01793-411500 Fax 01793-411501
www.nerc.ac.uk

Foreword

This development of the modelling software within the ZOOM family, of which ZETUP is a part, has been undertaken through a continuing tripartite collaboration between the University of Birmingham, the Environment Agency and the British Geological Survey. The development of ZETUP and ZOOMQ3D was initially undertaken at the University of Birmingham between 1998 and 2001 but continued after this time as a collaborative project between the three partner organisations. Since the inception of the collaborative project, the development of the software has been directed by the ZOOM steering committee, the members of which are:

University of Birmingham

Dr Andrew Spink

Environment Agency

Steve Fletcher

Paul Hulme

British Geological Survey

Dr Denis Peach

Dr Andrew Hughes

Dr Chris Jackson

Acknowledgements

The authors would like to acknowledge the assistance of A.G. Hughes and M.M. Mansour of the British Geological Survey and P.J. Hulme of the Environment Agency for their help in reviewing the ZETUP software. Additionally, the authors would like to acknowledge the assistance of the following colleagues at the British Geological Survey for reviewing this document: E. Cullis, A.G. Hughes and D.G. Kinniburgh.

Preface to the second edition

The production of the second edition of the ZETUP manual coincides with the release of version 1.03 of the code. This version of the code incorporates one only change to version 1.02.

DIFFERENCES BETWEEN VERSION 1.02 AND 1.03 OF ZETUP

- All executables should now be placed in a suitable directory e.g. 'c:\Program Files\ZOOM' and this folder should be added to the Windows system PATH variable. ZETUP can then be run from any working directory by typing the name of the executable followed by the path of the working directory e.g. 'ZETUP c:\myDirectory'. Alternatively the name of the executable can be followed by an input directory name (e.g. where an existing model is located) and an output directory name (e.g. where the files can be written after modification), for example, 'ZETUP c:\myDirectory c:\myDirectory\newFiles'. These strings could be placed in a batch file and the batch file run from the command line.

Contents

FIGURES

TABLES

1 Introduction

ZETUP is the pre-processor for the finite difference groundwater flow model ZOOMQ3D. It is used to construct the model grid, which may contain multiple areas of local grid refinement, and to create rivers within these complex meshes. ZETUP produces the input files required by ZOOMQ3D that define the structure of the model mesh and the structure of rivers (Table 1). It also produces templates of all of the other files required by ZOOMQ3D (Table 2) in the correct format. These can subsequently be modified using a text editor to complete the model specification. Finally, it produces a set of files that enable the visualisation and checking of the model structure (Table 3). If modifications to the structure of the model are required, it can be reloaded into ZETUP for alteration. Only the checking files listed in Table 3 are described in detail within this manual. Detailed descriptions of the files listed in Table 1 and 2, which form input to ZOOMQ3D are presented in the ZOOMQ3D manual (Jackson & Spink, 2004).

1.1 TERMINOLOGY

ZETUP is written using an object-oriented programming language. Whilst the users do not need to concern themselves with what this means, the term *object* is used a number of times within this manual and consequently, a brief explanation is required.

The user can think of an object in abstract terms as any distinct entity that stores data and perform tasks. In ZETUP and ZOOMQ3D objects are defined to represent real world features. For example, a pumped well is represented by an object. Pumped wells are described by data such as a depth and radius, and have the capability to pump water out of an aquifer. References are made in this manual to objects, which represent finite difference grids and rivers.

1.2 UNIT CONVENTION

All lengths in ZETUP must be specified in metres. The unit of time is specified as days. ZETUP reads hydraulic conductivity data for the beds of rivers, which is specified in m day^{-1}.

Table 1 List of all main ZOOMQ3D input files produced by ZETUP

1	aquifer.map	3	grids.out
2	boundary.out	4	rivers.out

NB. ‘boundary.out, ‘grids.out’, and ‘rivers.out’ must be renamed ‘boundary.dat’, ‘grids.dat’ and ‘rivers.dat’ for input into ZOOMQ3D. However, ‘boundary.dat’ is produced if ZETUP is being used to construct a new model. If an existing model is re-loaded into ZETUP for modification, ‘boundary.out’ will then be created.

Table 2 List of all other ZOOMQ3D input files produced by ZETUP as templates

1	anisotropy##.map & anisotropy##.cod	&	anisotropy##.cod	per layer
2	clock.dat			
3	contour_times.dat			
4	entry_method.dat			
5	fixedheads.dat			
6	gauging_stations.dat			
7	hydcond##.map & hydcond##.cod	&	hydcond##.dat	per layer
8	initialflow.dat			
9	initialh##.map & initialh##.cod			per layer
10	initialh.dat			
11	leakage.dat			
12	noflow##.map			per layer
13	obsleak.dat			
14	obswells.dat			
15	pumping.dat			
16	recharge.dat			
17	recharge.cod & recharge.map	&	recharge_rates.dat	
18	river_inputs.dat			
19	sor.dat			
20	specstor##.map & specstor##.cod		specstor##.dat	per layer
21	springs.dat			
22	syield##.map & syield##.cod		syield##.dat	per layer
23	vcond##.map & vcond##.cod	&	vcond##.dat	per layer
24	vkd.cod & vkd.map			
25	vkd.dat			
26	vkdkx01.map & vkdkx01.cod	&	vkdkx##.dat	
27	vkdky01.map & vkdky01.cod	&	vkdky##.dat	
28	vkdzp01.map & vkdzp01.cod	&	vkdzp##.dat	
29	vkdgrad01.map & vkdgrad01.cod	&	vkdgrad##.dat	
30	wetflag##.map			per layer
31	wethead##.map & wethead##.cod		wethead##.dat	
32	wetthresh##.map & wetthresh##.cod		wetthresh##.dat	
33	zbase##.map & zbase##.cod	&	zbase##.dat	per layer
34	zone_balance.dat			
35	zoomq3d.dat			
36	ztop##.map & ztop##.cod	&	ztop##.dat	per layer

NB. ## denotes the two digit model layer number starting at 01 at the top of the model.

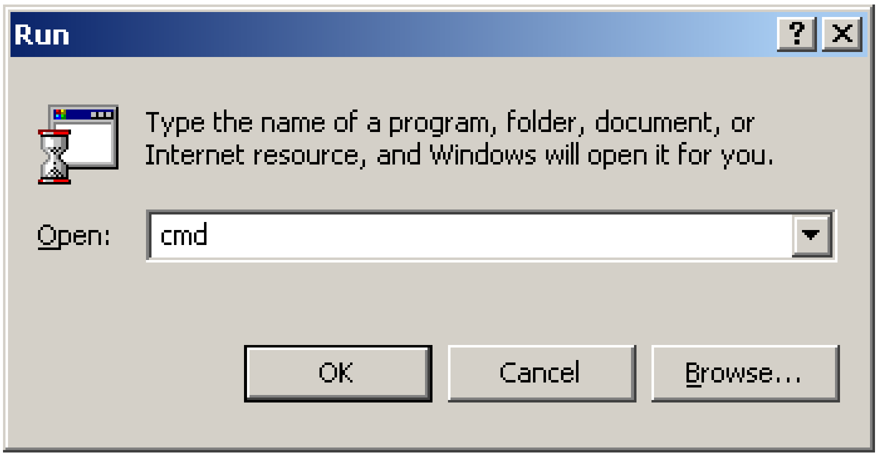
Table 3 List of files produced by ZETUP for model visualisation and checking

1	boundary.bln	5	river.dxf
2	boundary.dxf	6	river_check.ztp
3	boundary_map.ztp	7	rivernum.dxf
4	grid.dxf	8	spline.dxf

2 Running ZETUP

To install ZETUP on a Windows PC copy the executable ‘zetup.exe’ into suitable directory such as ‘c:\Program Files\ZOOM’. Then add this directory to the Windows system PATH variable (Control Panel→System→Advanced Tab→Environment Variables). No installation procedure is run in which ZETUP program files are added to the system registry. All the input files required by ZETUP to produce a specific model must be located in a single directory. The output files produced by ZETUP can be created in the same directory or a different directory. **It is strongly recommended that ZETUP and ZOOMQ3D are not run in the same directory and that their files are kept within separate folders**.

ZETUP should be run from the command line in a console window and not started from Windows Explorer. To start a console window select ‘Run’ from the Windows start menu and type ‘cmd’ in the drop down list box (Figure 1). The user should then change directory to that of the working directory. For help on the commands used to change directory type ‘help cd’ within the console window (Figure 2). To run the code type ‘zetup’ followed by the path to the working directory on the command line e.g. ‘**zetup c:\myZOOMQ3D_project**’. Alternatively the name of the executable can be followed by an input directory name (e.g. where an existing model is located) and an output directory name (e.g. where the files can be written after modification), such as ‘**ZETUP c:\myDirectory c:\myDirectory\newFiles**’. These strings could be placed in a batch file (a text file with a .bat extension e.g. ‘runzetup.bat’) and the name of this batch file typed on the command line (omit the extension when doing this e.g. type ‘runzetup’).

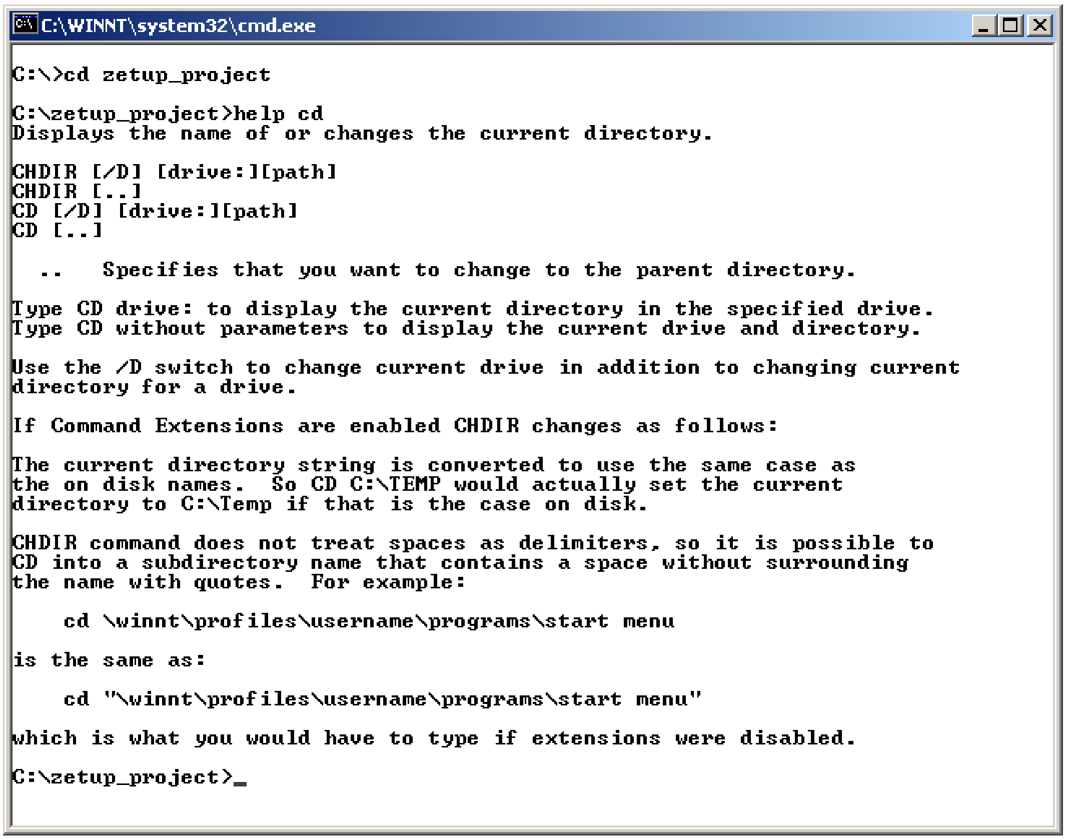


Type the name of a program, folder, document, or Internet resource, and Windows will open it for you.

Open: cmd

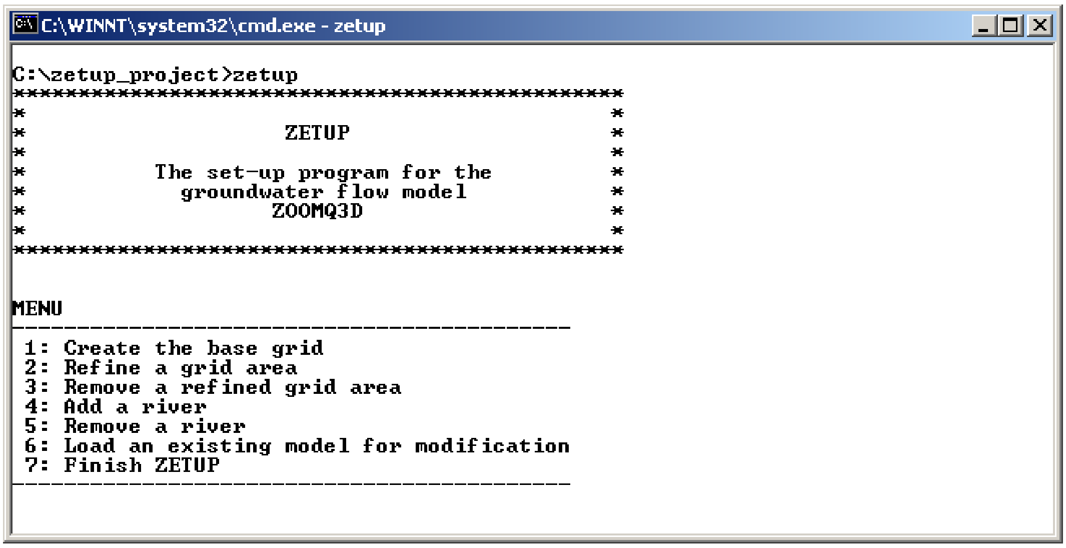
Figure 1 Starting a command line window from the Windows start menu

In the event that an error occurs, messages are written to the screen. If ZETUP is run from Explorer it may terminate before the user is able to read the error messages. The program is menu driven and accepts direct input from the keyboard. In addition to input from the keyboard, some procedures read data from files. The format of all the input data files is described in detail in the relevant section of this manual. On execution of the program the user is presented with the menu shown in Figure 3.



```
C:\>cd zetup_project

C:\zetup_project>help cd
Displays the name of or changes the current directory.

CHDIR [/D] [drive:][path]
CHDIR [..]
CD [/D] [drive:][path]
CD [..]

  ..   Specifies that you want to change to the parent directory.

Type CD drive: to display the current directory in the specified drive.
Type CD without parameters to display the current drive and directory.

Use the /D switch to change current drive in addition to changing current
directory for a drive.

If Command Extensions are enabled CHDIR changes as follows:

The current directory string is converted to use the same case as
the on disk names.  So CD C:\TEMP would actually set the current
directory to C:\Temp if that is the case on disk.

CHDIR command does not treat spaces as delimiters, so it is possible to
CD into a subdirectory name that contains a space without surrounding
the name with quotes.  For example:

    cd \winnt\profiles\username\programs\start menu

is the same as:

    cd "\winnt\profiles\username\programs\start menu"

which is what you would have to type if extensions were disabled.

C:\zetup_project>_
```

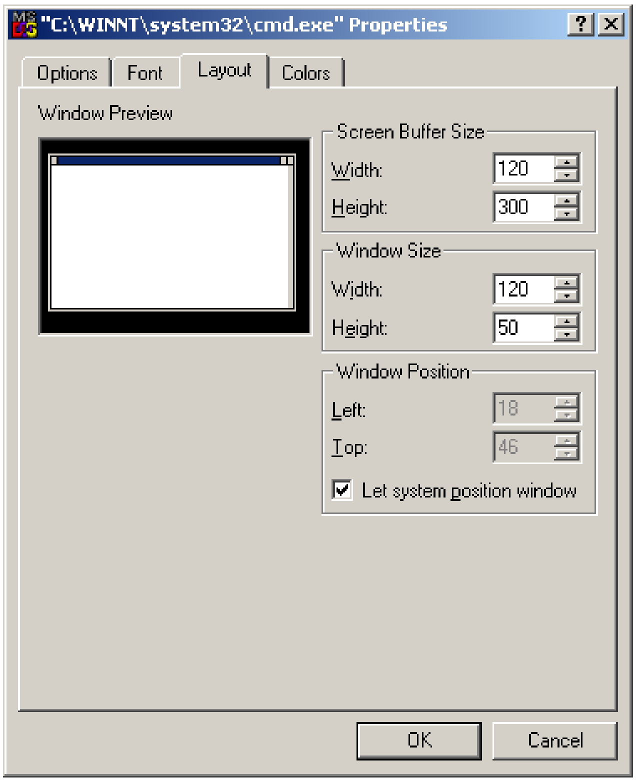
Figure 2   Example of changing the working directory within a console window



```
C:\zetup_project>zetup
***********************************************
*                                             *
*                    ZETUP                    *
*                                             *
*          The set-up program for the         *
*            groundwater flow model           *
*                   ZOOMQ3D                   *
*                                             *
***********************************************


MENU
-------------------------------------------
 1: Create the base grid
 2: Refine a grid area
 3: Remove a refined grid area
 4: Add a river
 5: Remove a river
 6: Load an existing model for modification
 7: Finish ZETUP
-------------------------------------------
```

Figure 3   Initial ZETUP screen

The size of the console box can be adjusted by clicking on the icon in the top left hand corner of its window and selecting ‘Properties’ from the menu list. Suitable values for the width and height of the window and its associated screen buffer are shown in Figure 3.



Screen Buffer Size
Width: 120
Height: 300

Window Size
Width: 120
Height: 50

Window Position
Left: 18
Top: 46
☑ Let system position window

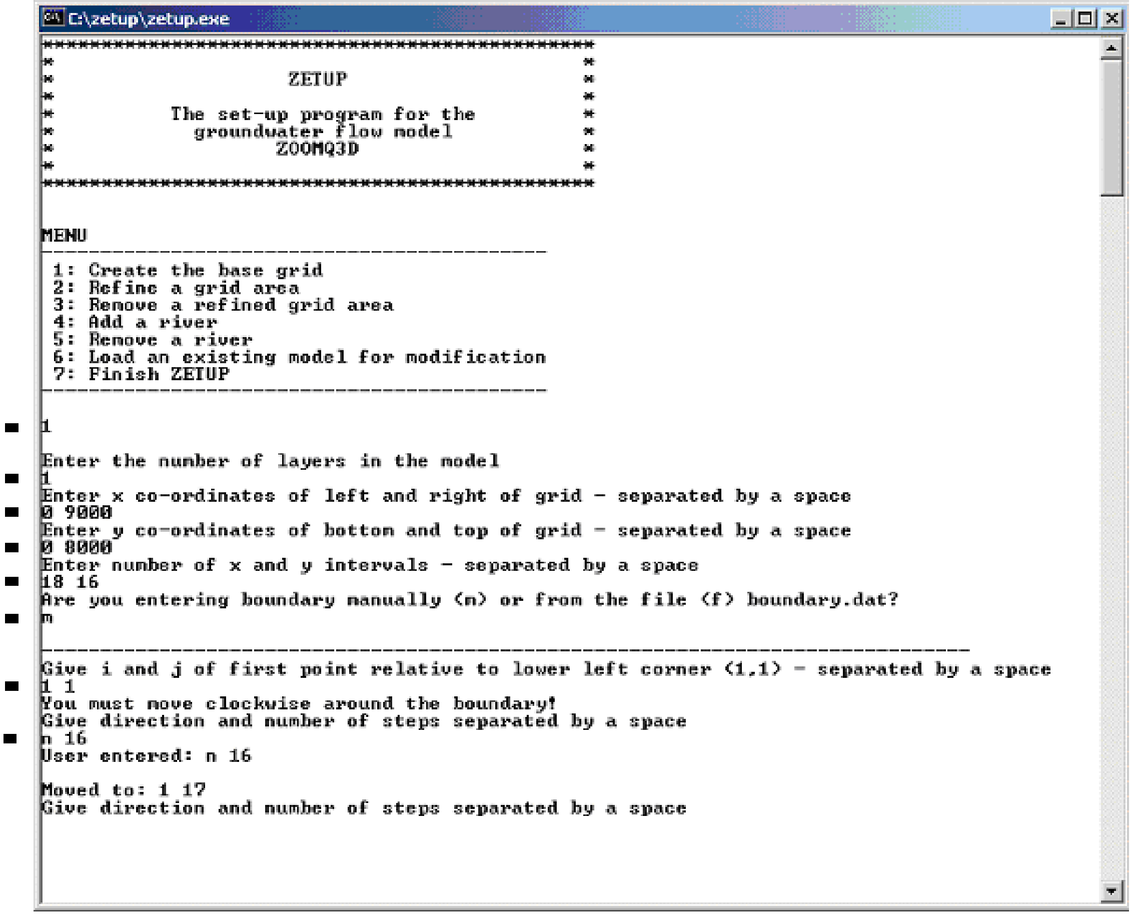
Figure 4 Changing the properties of the console window

3 Creating the base grid

To create the base grid, option 1 is selected from the menu. The user is then prompted for grid information as shown in Figure 5. The sequence of data entry is:

1. The number of layers in the model.
2. The co-ordinates of the left and right, and the bottom and top limits of the rectangular area containing the model. The lower left of the model does not have to be defined at (0, 0), and it could for example be a field co-ordinate.
3. The number of mesh intervals in the x and y co-ordinate directions.

In the example shown in Figure 5 the base grid is 9000 m wide from left to right and 8000 m wide from bottom to top. The data input by the user is highlighted by the dashes on the left hand side of the figure. The base grid is composed of a 500 m square mesh because 18 and 16 intervals are specified. Within each ZETUP / ZOOMQ3D grid the mesh is regular in the x and y directions. However, the mesh width may differ between the x and y directions. The model may therefore be composed of regular rectangular cells. Because local grid refinement is used to increase resolution, mesh grading, the gradual reduction or increase in cell size, is not supported. Regular meshes reduce the truncation error in the finite difference equations. Once the structure of the base grid has been specified, the shape of the boundary must be defined.



```
C:\zetup\zetup.exe
***********************************************
*                                             *
*                   ZETUP                     *
*                                             *
*          The set-up program for the         *
*            groundwater flow model           *
*                  ZOOMQ3D                    *
*                                             *
***********************************************


MENU
-------------------------------------------
 1: Create the base grid
 2: Refine a grid area
 3: Remove a refined grid area
 4: Add a river
 5: Remove a river
 6: Load an existing model for modification
 7: Finish ZETUP
-------------------------------------------

1

Enter the number of layers in the model
1
Enter x co-ordinates of left and right of grid - separated by a space
0 9000
Enter y co-ordinates of bottom and top of grid - separated by a space
0 8000
Enter number of x and y intervals - separated by a space
18 16
Are you entering boundary manually (m) or from the file (f) boundary.dat?
m

-----------------------------------------------------------------------------
Give i and j of first point relative to lower left corner (1,1) - separated by a space
1 1
You must move clockwise around the boundary!
Give direction and number of steps separated by a space
n 16
User entered: n 16

Moved to: 1 17
Give direction and number of steps separated by a space
```

Figure 5 Construction of the base grid using ZETUP

3.1 DEFINING THE MODEL BOUNDARY

The shape of the boundary is defined by issuing commands from the keyboard or through an input data file. *Manual* (m) or keyboard data entry is selected in Figure 5. If *file entry* is specified (f), the program reads boundary information from the file 'boundary_steps.dat', which must exist in the same directory as the ZETUP executable file.

The boundary definition procedure is simple. First, the column and row number (i, j) of a starting point on the boundary are entered. The origin of the (i, j) co-ordinate system is in the lower left of each grid at point (1,1). The boundary is then traced by defining steps around it. The direction as the cardinal point of a compass and the number of mesh interval steps to be taken are entered. **Note that the boundary must be traced in a clockwise direction**. To create a rectangular model, four pairs of commands are entered if the starting position on the boundary is the origin. To create the base grid shown in Figure 6a these are:

```
n 16
e 18
s 16
w 18
```

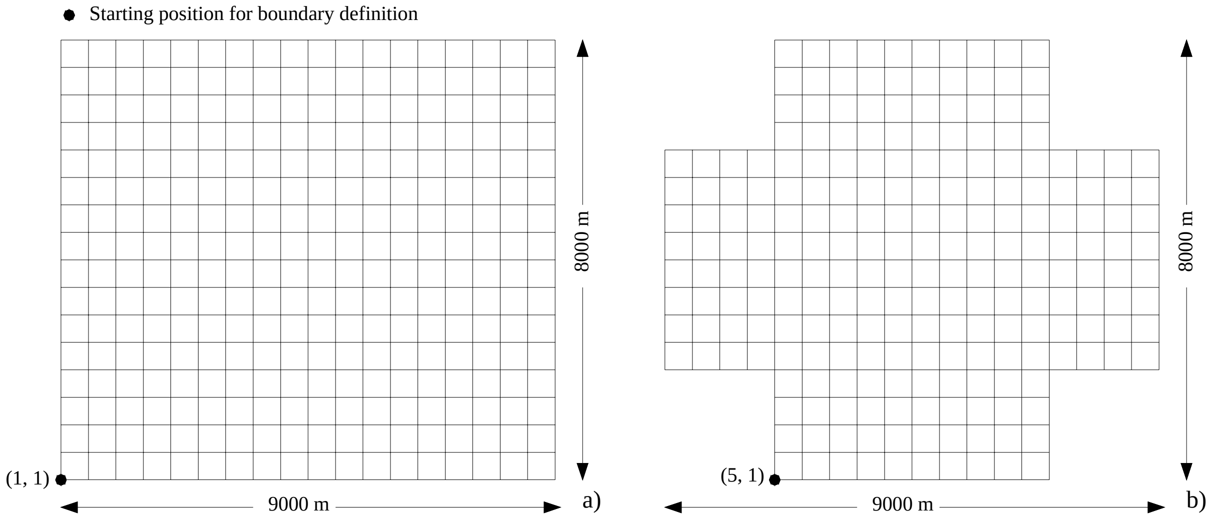


Figure 6 Example model boundaries

To define the boundary using a data file, the ASCII text file 'boundary_steps.dat' must be created in the same directory as 'zetup.exe'. This space or tab delimited file contains the i and j co-ordinates of the starting boundary position on its first line. Data corresponding to each move around the boundary are written on subsequent lines. For the examples shown in Figure 6, 'boundary_steps.dat' contains the lines shown in Table 4.

Table 4 Format of boundary definition file ‘boundary_steps.dat’

Format of ‘boundary_steps.dat’ for Figure 6a	Format of ‘boundary_steps.dat’ for Figure 6b
1 1 n 16 e 18 s 16 w 18	5 1 n 4 w 4 n 8 e 4 n 4 e 8 s 4 e 4 s 8 w 4 s 4 w 8

4 Refining the grid

After the base grid has been constructed the mesh can be refined. When an area of mesh is locally refined an additional grid is created. The new grid is referred to as either the *refined grid*, the *child grid* or the *subgrid*. The model mesh can be refined successively. This means that child grids can be refined, as can their children to produce a hierarchy of grid objects. The hierarchy for the mesh shown in Figure 7 is shown in Figure 8. Grids exist on a level within the hierarchy. The top level, level 1, can only contain one grid, the base grid. Lower levels can contain multiple grids. The order that grids are referenced within a level depends on their order of construction and destruction. This order affects the order in which they are listed within certain output files produced by ZETUP, which form input to ZOOMQ3D.

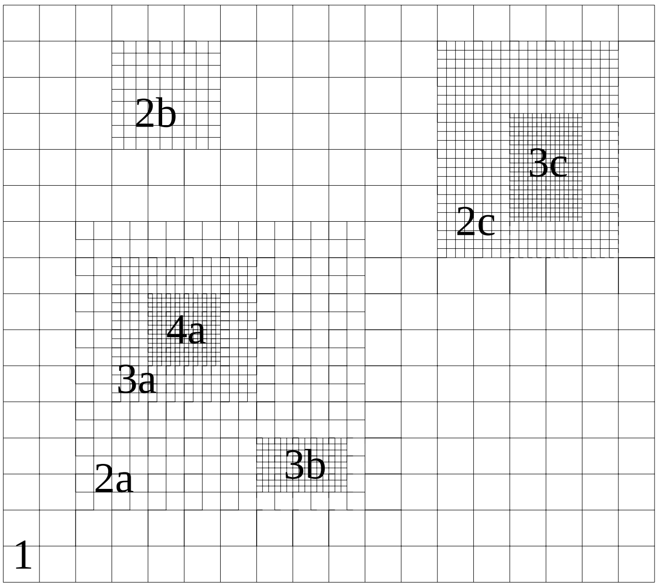


Figure 7 Example mesh with successive areas of grid refinement and level numbers

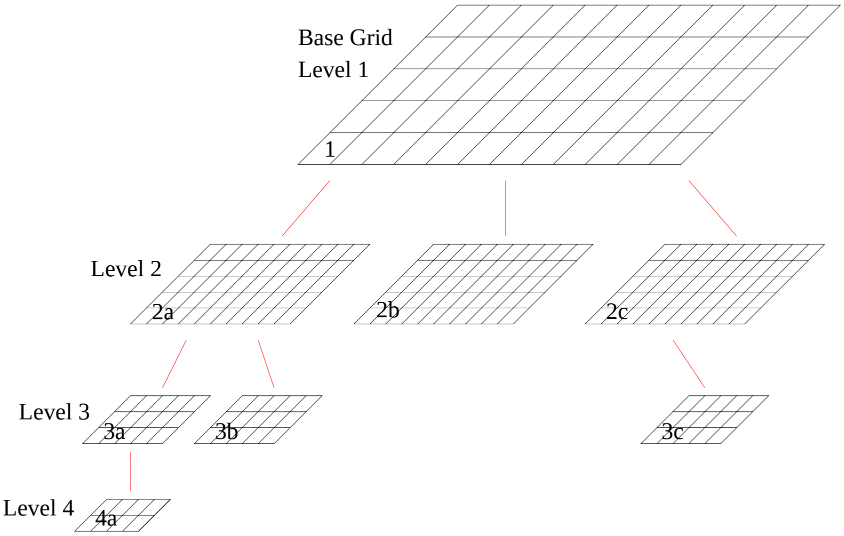
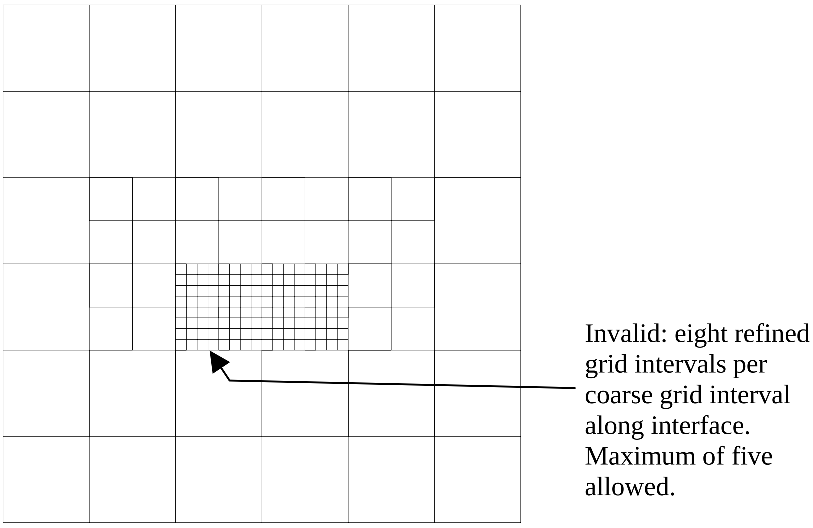


Figure 8 Hierarchy of grids for example mesh shown in Figure 7

The following rules apply to the local grid refinement procedure:

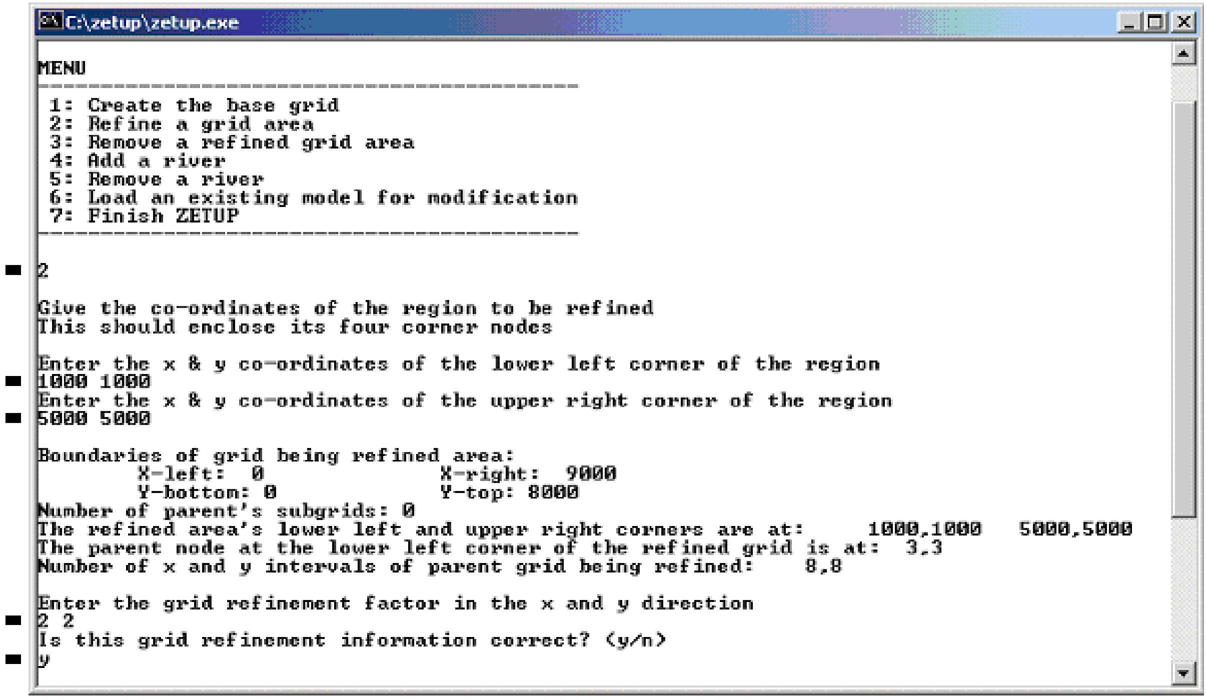
1. ZETUP can only produce rectangular areas of refinement.
2. The whole of the base grid cannot be refined.
3. The whole of a subgrid cannot be refined.
4. Refined grids on the same grid level must be separated by at least one mesh interval of their parent.
5. To maintain accuracy the mesh width cannot be reduced by more than five times at the interface between any two grids. For example, whilst ZETUP will allow the construction of the mesh shown in Figure 9 it is not valid.
6. It is permissible to refine across the model boundary.



Invalid: eight refined grid intervals per coarse grid interval along interface. Maximum of five allowed.

Figure 9 Excessive decrease in cell size at the interface of two grids

Refining the mesh is a simple procedure. As shown in Figure 10, it begins by selecting option 2 from the ZETUP menu. The user is prompted for the co-ordinates of the lower left and upper right corners of the rectangular region to be refined. This rectangle should enclose the parent grid nodes on the edge of the new refined grid. The grid refinement factors in the x and y directions are then entered. These integers represent the number of intervals into which each parent mesh interval is divided. They do not have to be identical but must be between 2 and 5. In the example shown in Figure 10, the mesh widths are halved. The data input by the user is highlighted by the dashes on the left hand side of the figure.

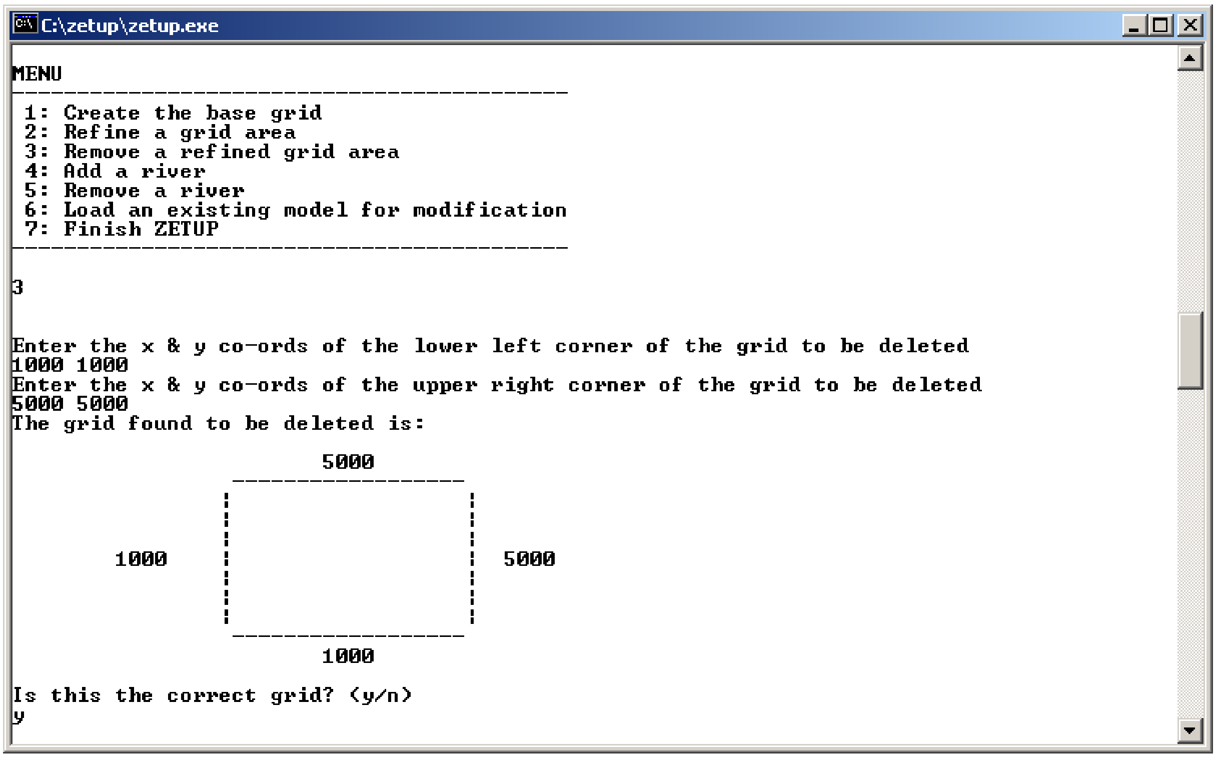


```
MENU
--------------------------------------------
 1: Create the base grid
 2: Refine a grid area
 3: Remove a refined grid area
 4: Add a river
 5: Remove a river
 6: Load an existing model for modification
 7: Finish ZETUP
--------------------------------------------

2

Give the co-ordinates of the region to be refined
This should enclose its four corner nodes

Enter the x & y co-ordinates of the lower left corner of the region
1000 1000
Enter the x & y co-ordinates of the upper right corner of the region
5000 5000

Boundaries of grid being refined area:
        X-left:  0              X-right:  9000
        Y-bottom: 0             Y-top: 8000
Number of parent's subgrids: 0
The refined area's lower left and upper right corners are at:     1000,1000   5000,5000
The parent node at the lower left corner of the refined grid is at:  3,3
Number of x and y intervals of parent grid being refined:    8,8

Enter the grid refinement factor in the x and y direction
2 2
Is this grid refinement information correct? (y/n)
y
```

Figure 10 Local grid refinement procedure in ZETUP

4.1 DELETING A REFINED GRID

Refined grids are removed from a model by selecting option 3 from ZETUP's menu. The procedure is shown in Figure 11. The user is prompted for the co-ordinates of the bottom left and top right corners of a rectangle enclosing the child grid. An illustration of the grid found by ZETUP for deletion is drawn on the screen and the user is asked if this is the correct grid. A grid cannot be deleted if it contains further areas of refinement. In this case, the areas of refinement must be deleted first.



```
MENU
-------------------------------------------
 1: Create the base grid
 2: Refine a grid area
 3: Remove a refined grid area
 4: Add a river
 5: Remove a river
 6: Load an existing model for modification
 7: Finish ZETUP
-------------------------------------------

3


Enter the x & y co-ords of the lower left corner of the grid to be deleted
1000 1000
Enter the x & y co-ords of the upper right corner of the grid to be deleted
5000 5000
The grid found to be deleted is:

                        5000
                  ------------------
                 ¦                  ¦
                 ¦                  ¦
        1000     ¦                  ¦    5000
                 ¦                  ¦
                 ¦                  ¦
                  ------------------
                        1000

Is this the correct grid? (y/n)
y
```

Figure 11 Deleting a refined grid using ZETUP

5 Constructing rivers

Significant changes can be made to the structure of a ZOOMQ3D model with little effort using the local grid refinement facility. Whilst the technique allows the user to focus attention on different areas within an aquifer, grid refinement may affect other model features, for example rivers. Re-designing the mesh in many conventional regional groundwater models requires the user to redefine the model's rivers and their parameters manually. This can be a time consuming task. A technique is implemented in ZOOMQ3D, which automates much of the work involved in changing a model grid and its associated rivers. This technique is based on the differentiation between model rivers and data that describes the real geometry and characteristics of rivers. Objects are defined in ZETUP to represent the real structure of rivers. These are considered to be 'maps' of the river channels. When the mesh is refined, the river map objects examine the new grid structure and use it to reconstruct the model's rivers. This process is described in this section. First the representation of rivers as 'maps' is described.

5.1 APPROXIMATING REALITY: RIVERS AND SPLINES

This sub-section illustrates how rivers are represented in ZETUP, how they are translated into the form required by ZOOMQ3D and how rivers are automatically modified during grid refinement. Rivers are represented within ZETUP by objects that aim to represent their real structure closely. Three types of objects are defined to represent real rivers. These are termed *catchment maps*, *river maps* and *river branches.* Catchment maps store and manage a number of river maps, which are composed of one or more river branches. The hierarchy is shown in Figure 12. The objects are described as follows:

> *Catchment map*: A catchment map is a collection of river maps.
>
> *River map*: A river map is composed of a series of polylines that describe the geometry of the channels within a dendritic river basin i.e. the main river channel and all its tributaries. In addition to describing the geometry of the basin, river maps store data relating to the physical and hydraulic characteristics of the channels at specific points along their length. **A river map cannot represent two separate river basins i.e. all the channels within a river map must be connected**. River maps are referenced using a unique integer number.
>
> *River branch*: Either the main channel of a river basin or one of its tributaries. A numbering scheme must be adopted in which the upstream branches have a higher value. The main channel is assigned number 1. A river branch is one of the polylines forming the river map.

Each river branch is represented by a series of mathematically generated curves that are 'fitted' to the real shape of the river branch. These curves, which are based on a mixture of cubic splines and linear interpolants, are constructed after the user specifies a series of points, referred to as *spline points*, along a branch. River parameter information is defined at each of these spline points. For example, Figure 13 shows a dendritic river system, each branch of which is defined by a series of spline points. The user specifies in an input file the following information at each spline point:

1. x and y co-ordinates of the point,
2. river stage,

3. river-bed elevation,
4. river width,
5. vertical hydraulic conductivity of the river bed under effluent conditions,
6. vertical hydraulic conductivity of the river bed under influent conditions and,
7. river bed thickness.

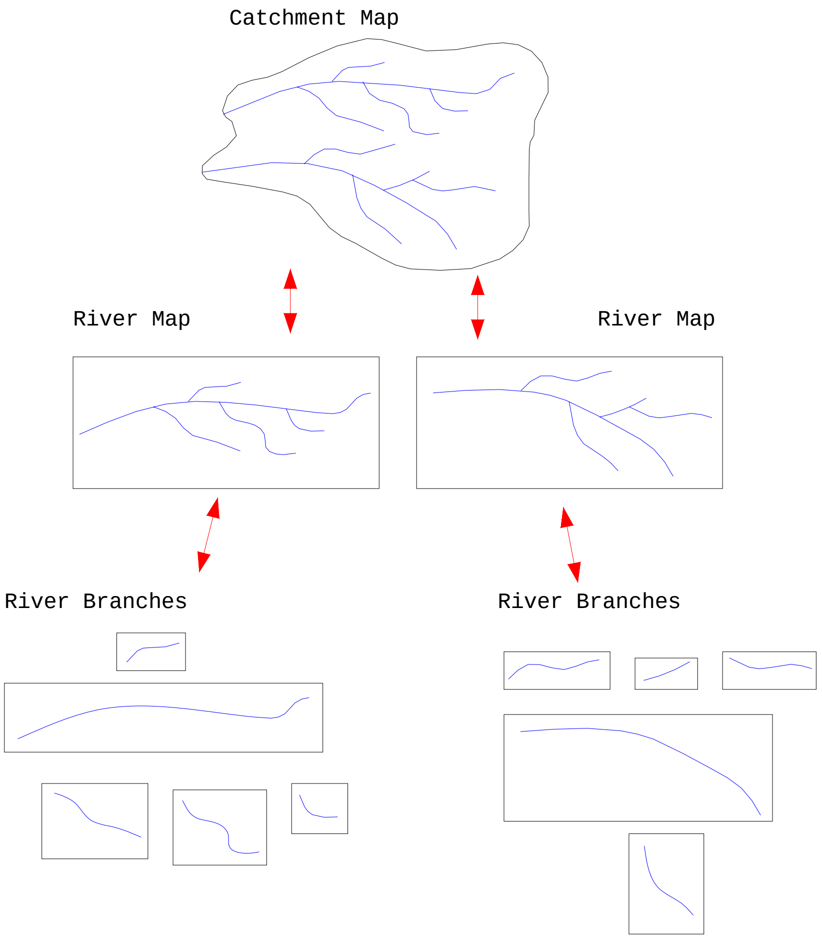


Figure 12 Representation of real rivers as catchment map, river map and river branch objects

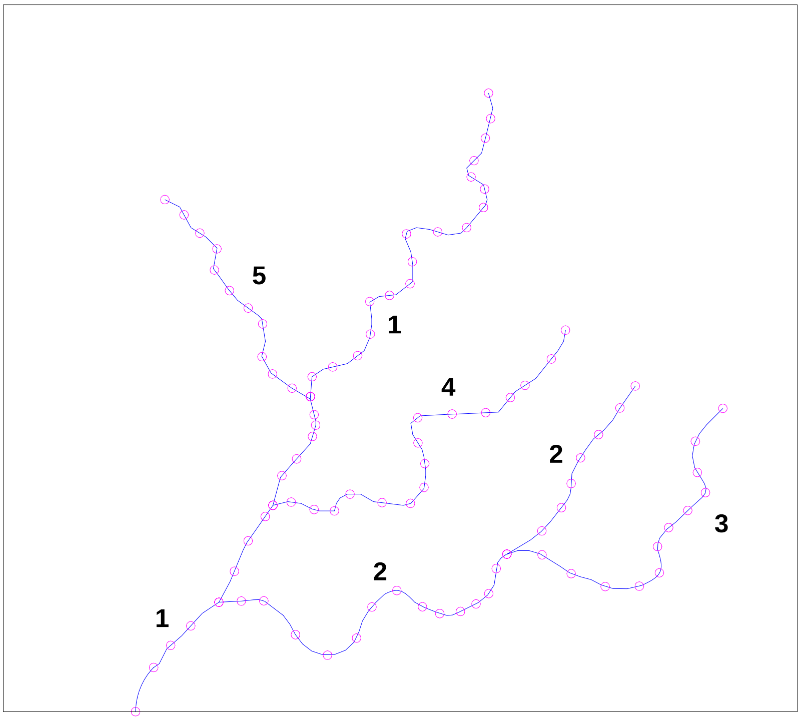


Figure 13 Definition of spline points along a river and branch numbers

Each river branch is defined by its associated spline points and the data at these locations. In Figure 13 there are five river branches, numbered as shown. Branch 1 is the main river channel. **A numbering scheme must be adopted in which the upstream branches have a higher value. Similarly, for each branch, spline points are specified from downstream to upstream. Spline points must be defined at the ends of the branches and at the confluence of two or more channels. A spline point at a confluence exists on all the tributaries flowing to this point (and has to be defined on each branch within the river map input file).**

ZETUP fits curves through the spline points. These curves represent the real river and are distinct from the model rivers. The *splined* river is shown in Figure 14. It should be noted that splines approximate the true shape of a curve through a set of points and thus will not represent the river geometry exactly. Where greater accuracy is required a larger number of spline points should be defined. It may be beneficial to define extra points in the region of particular interest or where a river bends tightly or meanders irregularly. In some instances, if an insufficient number of splines points are placed along the river, the fitted curve will oscillate wildly as shown in Figure 15. However, ZETUP checks for irregular behaviour of the spline by comparing:

- the direction of the spline at each spline point with that of the straight line between its two adjacent spline points,
- the length of the spline between two adjacent spline points with the length of the straight line between the same two points.

Where either of these comparisons indicates that there is poor agreement, and consequently that the spline is oscillating between two points, the appropriate sections are replaced by straight lines. If straight lines are inadequate the user can improve the representation of the shape of the river by adding more spline points. These oscillations occur in the upper reaches of branch 1 in Figure 15 and are rectified using the lines shown in Figure 16. A detailed description of cubic splines and the splining process is given in Appendix 1.

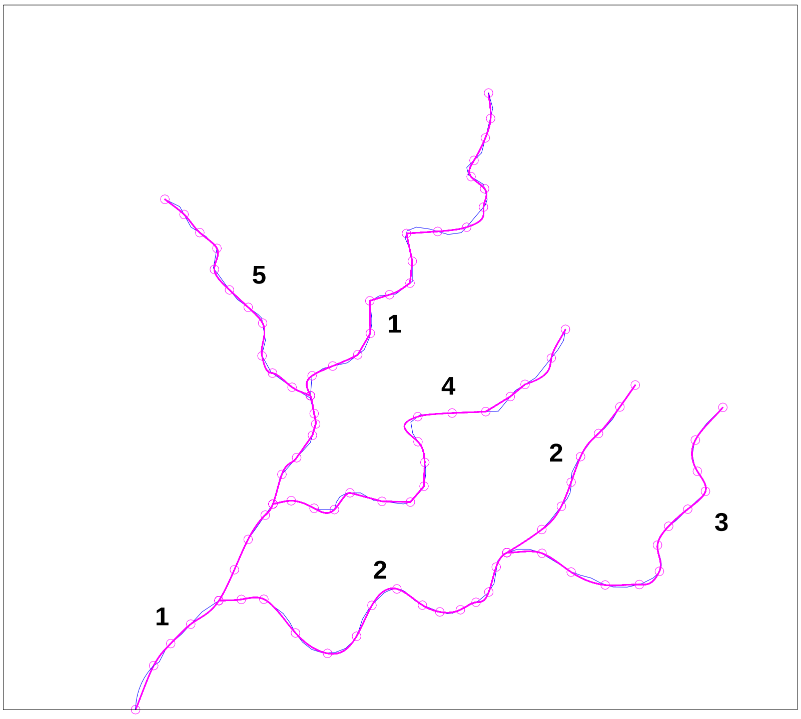


Figure 14 Representation of real river as a series of splines and lines in river map objects

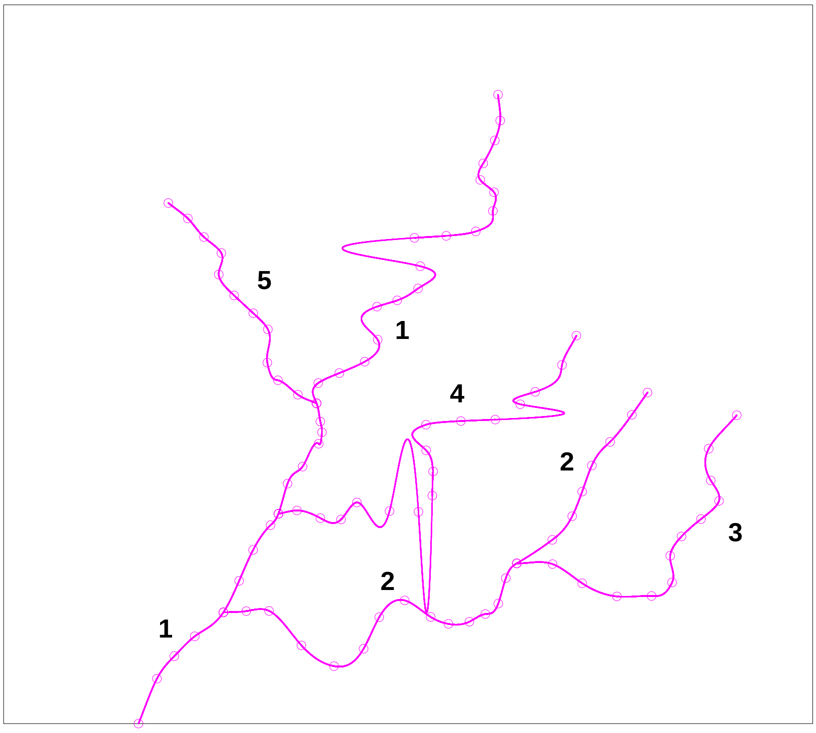


Figure 15 Oscillation of splines caused by a low number of spline points

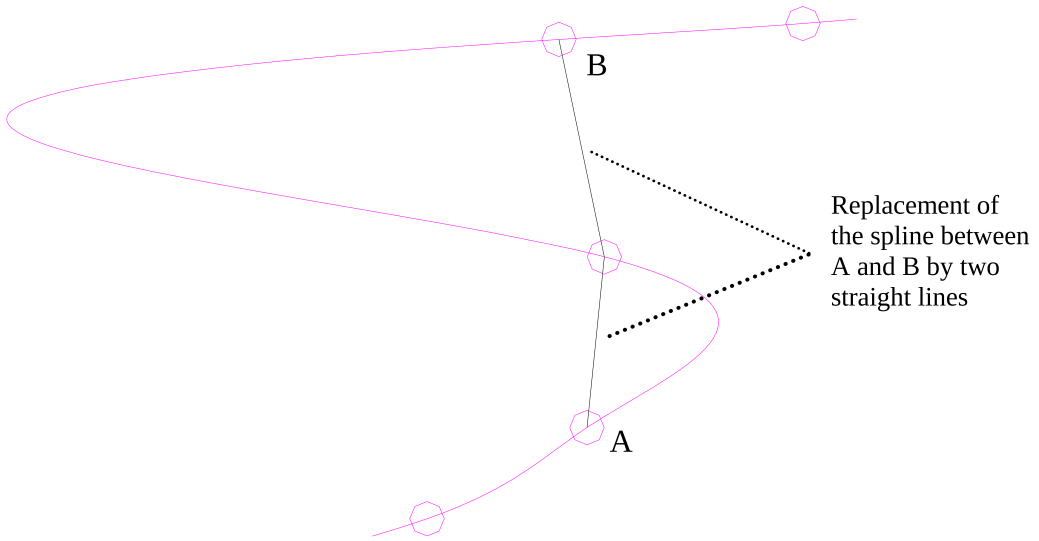

B
A
Replacement of the spline between A and B by two straight lines

Figure 16 Replacement of splines by straight lines

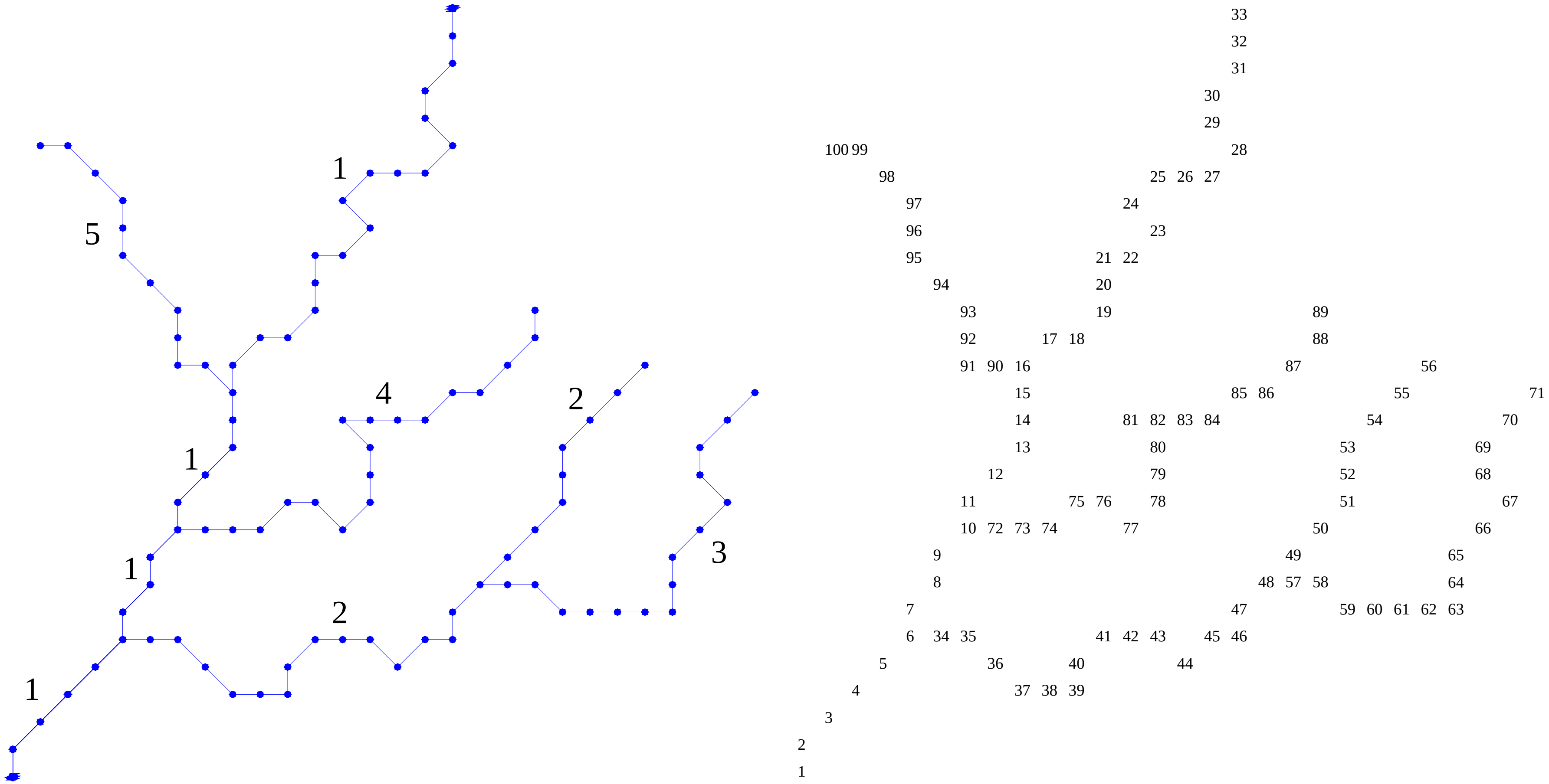
5.2 TRANSLATING ZETUP'S RIVER MAPS INTO ZOOMQ3D RIVERS

After the river maps have been created and the real rivers approximated by splines, the maps are used to generate the rivers required by the *simulation* model, ZOOMQ3D.

ZOOMQ3D can simulate the baseflow within dendritic river basins. Both the baseflow along the river and the interaction between the river and the aquifer are modelled. The structure of the numerical model rivers within ZOOMQ3D is described using three terms: rivers, branches and river nodes, which have the following definition:

> *Rivers*: a river is composed of both an interconnected series of river branches and a interconnected series of river nodes. **A river cannot be composed of two sets of interconnected river nodes that are not themselves connected, i.e. a river cannot represent multiple separate catchments**. ZOOMQ3D rivers represent the translation of a ZETUP river map onto the finite difference grid.
>
> *Branch*: a branch represents the main channel in a catchment or a tributary within the river (catchment). Only dendritic river catchments can be modelled and consequently, each branch has one node at its upstream end and one node at its downstream end.
>
> *River node*: a river node represents a point on the river that coincides with a horizontal point of the finite difference mesh. The characteristics of the river are defined at each river node. A river node can have up to five upstream connections to river nodes but only one downstream connection.

Branches and river nodes are organised in a specific order described by a numbering scheme. **For river branches a numbering scheme is implemented in which the upstream branches have a higher integer value (Figure 17a)**. This numbering scheme is defined by the river map file containing the spline point data. **River node numbers also increase upstream and from low branch numbers to high branch numbers (Figure 17b)**. The river node at the bottom of the catchment is number 1.



(a) (b)

Figure 17 Numbering schemes in ZOOMQ3D for (a) river branches and (b) river nodes for an example model river

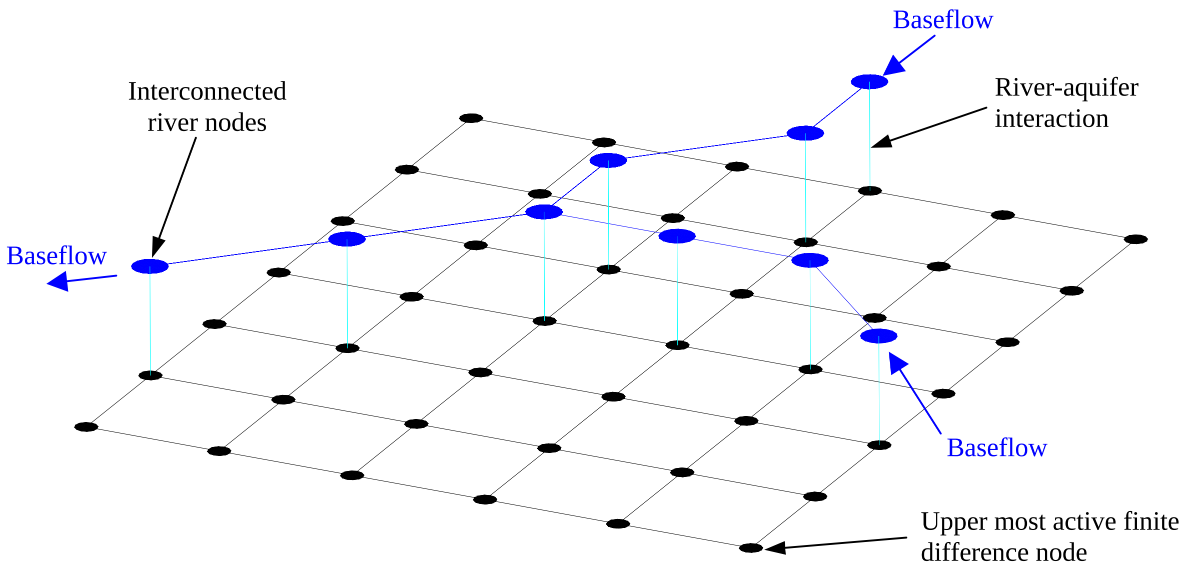

Baseflow
Interconnected river nodes
River-aquifer interaction
Baseflow
Baseflow
Upper most active finite difference node

Figure 18 Schematic representation of connections between river nodes and grid nodes

As stated above, model rivers are represented by a series of linked river nodes, which are also connected with upper most active aquifer node as illustrated in Figure 18. Each river node is located at the centre of the reach that it represents. The limits of a reach are the mid-points between the central river node and its adjacent river nodes. River nodes are characterised by the following set of parameters:

- Location specified by x and y co-ordinates.
- Width (m).
- Reach length (m). Equivalent to the half the distance between the adjacent two river nodes.
- River stage (m).
- River-bed elevation (m).
- River-bed thickness (m).
- Vertical hydraulic permeability under effluent conditions (m day^{-1}).
- Vertical hydraulic permeability under influent conditions (m day^{-1}).

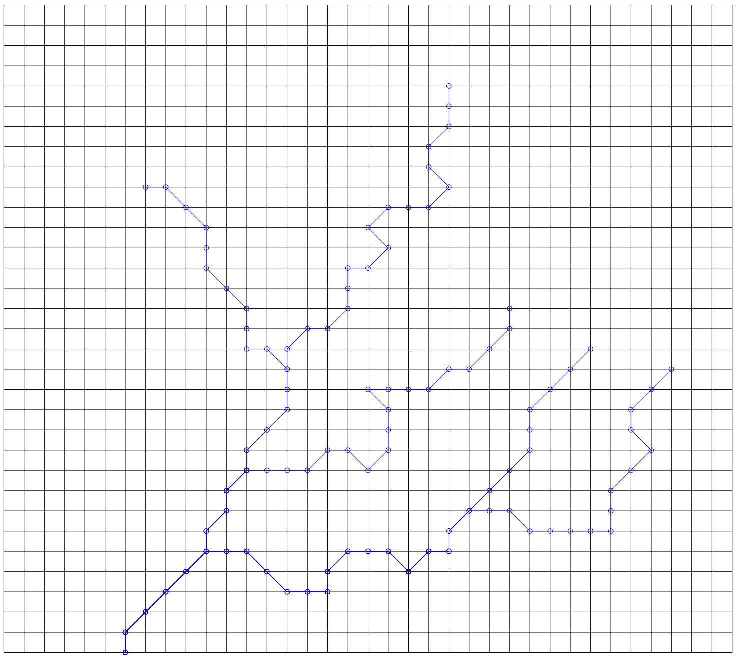


Figure 19 Model river generated by river map object on base grid

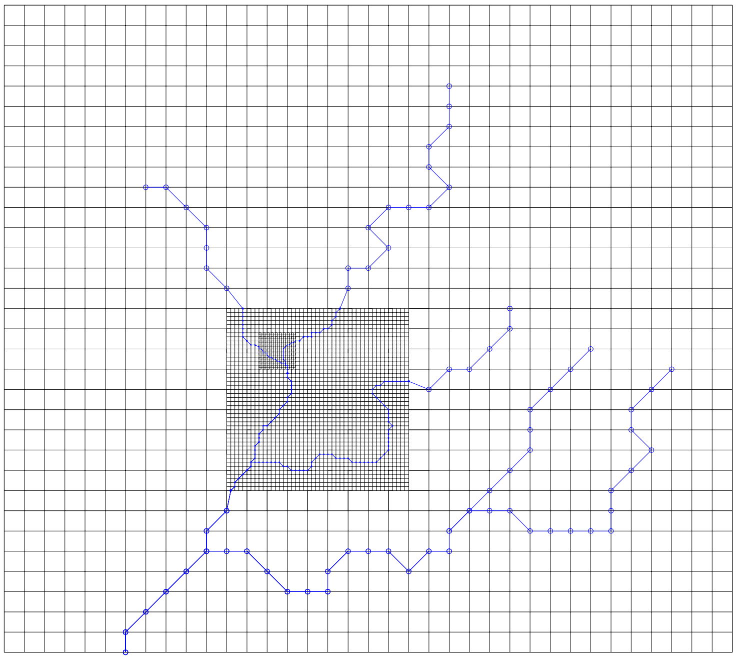


Figure 20 Model river generated by river map object on refined grid

In ZETUP river maps, or river splines, create the model's rivers by interrogating the finite difference grid. If there is only a base grid, a relatively coarse model river will be created as shown in Figure 19. This model river is an approximation of the splined river as it is mapped on the base grid. If the grid is subsequently refined the model river is automatically deleted and a new river created. The river map object interrogates the newly refined mesh and creates a more detailed representation on the finite difference grid. An example of the simultaneous refinement of both the model mesh and the river is shown in Figure 20.

Hydraulic parameters must be defined at the river nodes, which are created on the grid. This process uses the values of the hydraulic parameters specified at the spline points and involves interpolation between adjacent spline points. The procedure is defined by the following steps and is illustrated in Figure 21.

1. After a river has been created on the finite difference grid, the model river nodes are scanned.
2. The closest point on the spline to a river node is located.
3. The distance, ΔL1, between the two spline points straddling the nearest point is calculated.
4. The distance, ΔL2, between the point closest to the river node and one of the adjacent spline points is calculated.
5. The ratio ΔL2:ΔL1 is used to interpolate hydraulic parameters at the river node, for example channel width, bed thickness or bed permeability. Because ΔL1 and ΔL2 are measured along the spline this is in effect linear interpolation along the spline.

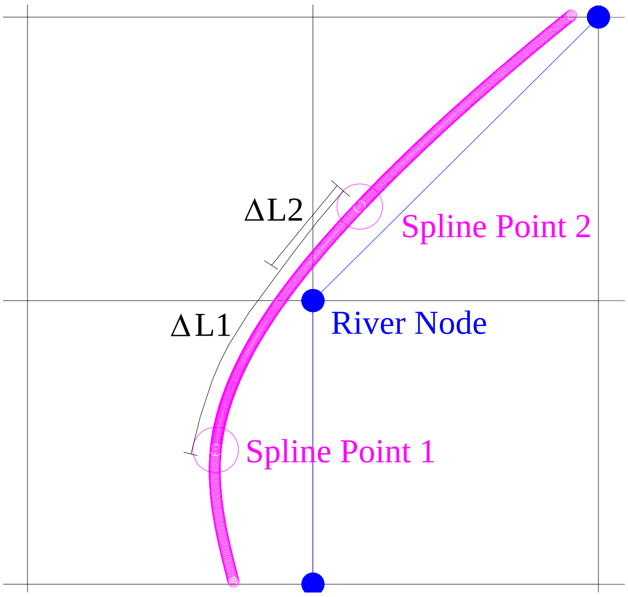
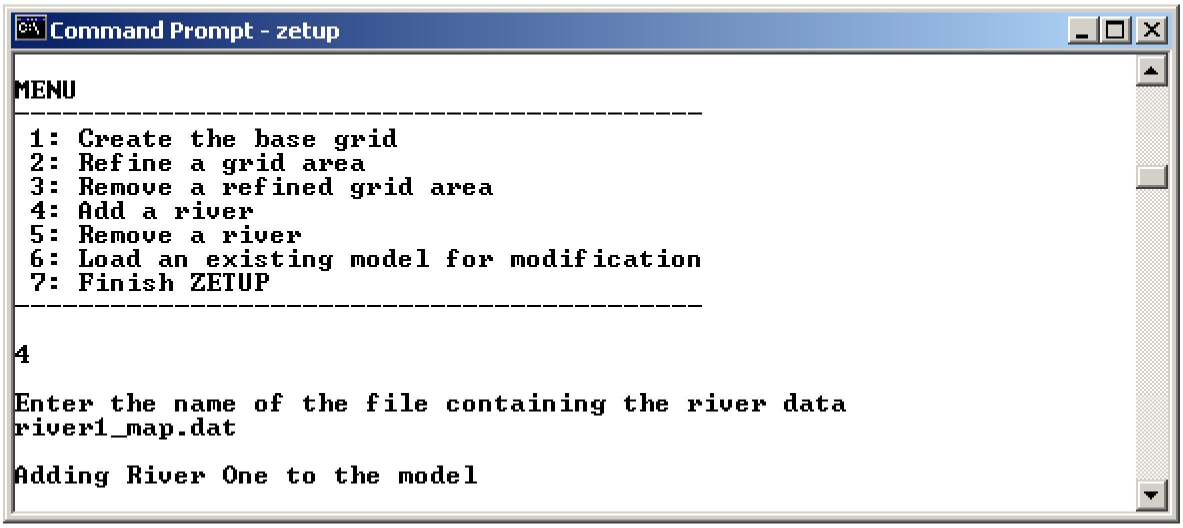


Figure 21 Interpolation of river hydraulic parameters along river spline

5.3 USING ZETUP TO ADD A RIVER TO THE MODEL

This section describes the input files required to construct the representation of rivers as maps, or splines, and the construction of the model rivers. Rivers can only be added to the model after the base grid has been constructed. Thereafter, they can be added both before and after the grid is refined. Rivers are added by selecting option 4 from the ZETUP menu as shown in Figure 22. ZETUP requests the name of the data file that is used to create the river map. The file contains data associated with only one river (i.e. one set of connected splines). An appropriate name for the river data file may be the name of the river itself.



```
MENU
-------------------------------------------
 1: Create the base grid
 2: Refine a grid area
 3: Remove a refined grid area
 4: Add a river
 5: Remove a river
 6: Load an existing model for modification
 7: Finish ZETUP
-------------------------------------------

4

Enter the name of the file containing the river data
river1_map.dat

Adding River One to the model
```

Figure 22 Adding a river to the model using ZETUP

The input data file contains the river parameter information at the spline points specified by the user. The format of the data file is shown in Figure 23. This file, named 'river1_map.dat' and as shown in Figure 22, is used to create the river maps and model rivers shown in Figures 14, 17, 19 and 20. River map files are space or tab delimited ASCII text files. The second column in Figure 23 gives the line numbers, which are not contained in the data file and are used for reference purposes only. The format of the data is described next.

The first line contains the name of the river and is a character string. On line 2 the river number and the number of branches in the river are entered. The first number acts as a unique identifier for the river. It is used for example, when a river is being deleted by the user. **Each river must be given a unique number**. After the first two lines of the input file, blocks of data define each river branch. The first block of data is for branch 1, the second block for branch 2 etc. A numbering scheme must be adopted in which the upstream branches have a higher value.

The first line of each block of branch data is a character string or comment line. On the second line of the block, the number of its downstream branch and the number of spline points on the branch are input. A downstream branch number is read from the file for branch 1 but not used by the program.

The data associated with each spline point is specified on the subsequent lines of the block. Each line represents one spline point. The data at each spline point is described in Table 5 where the line numbers in the third column refer to the file 'river1_map.dat'. The spline point data within each block must be listed in sequence from the downstream to upstream end of the branch. A spline point should be defined at the upper and lower limits of the branch and at the confluence of any channels. **Spline points at a river confluence will be defined more than once in the data file because they exist on more than one branch.** Care must be taken at these points to replicate the data associated with the point exactly. If different data is defined at the same spline point on different branches ZETUP produces a warning.

```
‘river1_map.dat’                                                                          Line
River One                                                                                    1
1 5                                                                                          2
Branch One                                                                                   3
0 33                                                                                         4
1500.00     0.00          1.0       0.0       5.0     0.00005     0.0001        1.0          5
1705.19     500.00        1.0       0.0       5.0     0.00005     0.0001        1.0          6
1897.98     750.26        2.0       1.0       5.0     0.00005     0.0001        1.0          7
2125.33     971.29        3.0       2.0       5.0     0.00005     0.0001        1.0          8
2444.21     1238.28       4.0       3.0       5.0     0.00005     0.0001        1.0          9
2619.65     1588.58       5.0       4.0       5.0     0.00005     0.0001        1.0         10
2775.24     1933.28       6.0       5.0       5.0     0.00005     0.0001        1.0         11
2968.92     2208.70       7.0       6.0       5.0     0.00005     0.0001        1.0         12
3056.55     2335.16       8.0       7.0       5.0     0.00005     0.0001        1.0         13
3157.95     2670.90       9.0       8.0       5.0     0.00005     0.0001        1.0         14
3325.22     2860.72       10.0      9.0       5.0     0.00005     0.0001        1.0         15
3503.97     3114.91       11.0      10.0      5.0     0.00005     0.0001        1.0         16
3540.83     3242.75       12.0      11.0      5.0     0.00005     0.0001        1.0         17
3522.90     3361.46       13.0      12.0      5.0     0.00005     0.0001        1.0         18
3481.09     3564.39       14.0      13.0      5.0     0.00005     0.0001        1.0         19
3499.96     3790.63       15.0      14.0      5.0     0.00005     0.0001        1.0         20
3733.69     3901.11       16.0      15.0      5.0     0.00005     0.0001        1.0         21
4016.90     4028.22       17.0      16.0      5.0     0.00005     0.0001        1.0         22
4160.20     4273.27       18.0      17.0      5.0     0.00005     0.0001        1.0         23
4153.78     4641.10       19.0      18.0      5.0     0.00005     0.0001        1.0         24
4377.86     4710.73       20.0      19.0      5.0     0.00005     0.0001        1.0         25
4610.02     4842.98       21.0      20.0      5.0     0.00005     0.0001        1.0         26
4635.26     5090.59       22.0      21.0      5.0     0.00005     0.0001        1.0         27
4569.34     5406.03       23.0      22.0      5.0     0.00005     0.0001        1.0         28
4923.48     5429.08       24.0      23.0      5.0     0.00005     0.0001        1.0         29
5251.89     5477.51       25.0      24.0      5.0     0.00005     0.0001        1.0         30
5442.41     5705.78       26.0      25.0      5.0     0.00005     0.0001        1.0         31
5454.84     5914.11       27.0      26.0      5.0     0.00005     0.0001        1.0         32
5301.88     6051.11       28.0      27.0      5.0     0.00005     0.0001        1.0         33
5335.00     6235.54       29.0      28.0      5.0     0.00005     0.0001        1.0         34
5463.65     6490.63       30.0      29.0      5.0     0.00005     0.0001        1.0         35
5523.75     6711.31       31.0      30.0      5.0     0.00005     0.0001        1.0         36
5500        7000          32.0      31.0      5.0     0.00005     0.0001        1.0         37
Branch 2                                                                                    38
1 22                                                                                        39
2444.21     1238.28       4.0       3.0       5.0     0.00005     0.0001        1.0         40
2698.28     1251.72       5.0       4.0       5.0     0.00005     0.0001        1.0         41
2953.01     1254.21       6.0       5.0       5.0     0.00005     0.0001        1.0         42
3312.10     870.89        7.0       6.0       5.0     0.00005     0.0001        1.0         43
3674.81     639.72        8.0       7.0       5.0     0.00005     0.0001        1.0         44
4004.29     833.77        9.0       8.0       5.0     0.00005     0.0001        1.0         45
4178.39     1184.79       10.0      9.0       5.0     0.00005     0.0001        1.0         46
4460.88     1370.84       11.0      10.0      5.0     0.00005     0.0001        1.0         47
4750.55     1186.89       12.0      11.0      5.0     0.00005     0.0001        1.0         48
4947.23     1109.77       13.0      12.0      5.0     0.00005     0.0001        1.0         49
5180.97     1134.26       14.0      13.0      5.0     0.00005     0.0001        1.0         50
5358.22     1219.54       15.0      14.0      5.0     0.00005     0.0001        1.0         51
5502.13     1335.76       16.0      15.0      5.0     0.00005     0.0001        1.0         52
5586.31     1619.23       17.0      16.0      5.0     0.00005     0.0001        1.0         53
5707.80     1783.15       18.0      17.0      5.0     0.00005     0.0001        1.0         54
6103.53     2045.39       19.0      18.0      5.0     0.00005     0.0001        1.0         55
6324.95     2308.62       20.0      19.0      5.0     0.00005     0.0001        1.0         56
6435.67     2582.08       21.0      20.0      5.0     0.00005     0.0001        1.0         57
6542.99     2872.33       22.0      21.0      5.0     0.00005     0.0001        1.0         58
6746.35     3135.54       23.0      22.0      5.0     0.00005     0.0001        1.0         59
6988.48     3436.85       24.0      23.0      5.0     0.00005     0.0001        1.0         60
7162.76     3685.23       25.0      24.0      5.0     0.00005     0.0001        1.0         61
Branch 3                                                                                    62
2 13                                                                                        63
5707.80     1783.15       18.0      17.0      5.0     0.00005     0.0001        1.0         64
6106.05     1775.10       19.0      18.0      5.0     0.00005     0.0001        1.0         65
6435.90     1562.03       20.0      19.0      5.0     0.00005     0.0001        1.0         66
6821.69     1415.43       21.0      20.0      5.0     0.00005     0.0001        1.0         67
7208.79     1420.47       22.0      21.0      5.0     0.00005     0.0001        1.0         68
7435.78     1571.15       23.0      22.0      5.0     0.00005     0.0001        1.0         69
7414.83     1868.39       24.0      23.0      5.0     0.00005     0.0001        1.0         70
7541.03     2082.92       25.0      24.0      5.0     0.00005     0.0001        1.0         71
7757.48     2276.53       26.0      25.0      5.0     0.00005     0.0001        1.0         72
7958.14     2478.73       27.0      26.0      5.0     0.00005     0.0001        1.0         73
7866.23     2706.72       28.0      27.0      5.0     0.00005     0.0001        1.0         74
7843.54     3060.34       29.0      28.0      5.0     0.00005     0.0001        1.0         75
8155.23     3432.09       30.0      29.0      5.0     0.00005     0.0001        1.0         76
Branch 4                                                                                    77
1 17                                                                                        78
3056.55     2335.16       8.0       7.0       5.0     0.00005     0.0001        1.0         79
3263.34     2371.94       9.0       8.0       5.0     0.00005     0.0001        1.0         80
3522.79     2287.27       10.0      9.0       5.0     0.00005     0.0001        1.0         81
3753.34     2271.88       11.0      10.0      5.0     0.00005     0.0001        1.0         82
3929.52     2461.72       12.0      11.0      5.0     0.00005     0.0001        1.0         83
4292.99     2365.33       13.0      12.0      5.0     0.00005     0.0001        1.0         84
4614.36     2357.29       14.0      13.0      5.0     0.00005     0.0001        1.0         85
4768.63     2537.37       15.0      14.0      5.0     0.00005     0.0001        1.0         86
4777.79     2807.18       16.0      15.0      5.0     0.00005     0.0001        1.0         87
4699.86     3041.30       17.0      16.0      5.0     0.00005     0.0001        1.0         88
4698.03     3326.40       18.0      17.0      5.0     0.00005     0.0001        1.0         89
5087.27     3367.78       19.0      18.0      5.0     0.00005     0.0001        1.0         90
5468.60     3383.36       20.0      19.0      5.0     0.00005     0.0001        1.0         91
5746.49     3554.72       21.0      20.0      5.0     0.00005     0.0001        1.0         92
5913.19     3692.88       22.0      21.0      5.0     0.00005     0.0001        1.0         93
6211.52     3991.51       23.0      22.0      5.0     0.00005     0.0001        1.0         94
6371.60     4317.97       24.0      23.0      5.0     0.00005     0.0001        1.0         95
Branch 5                                                                                    96
1 12                                                                                        97
3481.09     3564.39       14.0      13.0      5.0     0.00005     0.0001        1.0         98
3273.14     3660.80       15.0      14.0      5.0     0.00005     0.0001        1.0         99
3051.39     3821.22       16.0      15.0      5.0     0.00005     0.0001        1.0        100
2933.03     4016.95       17.0      16.0      5.0     0.00005     0.0001        1.0        101
2939.26     4388.19       18.0      17.0      5.0     0.00005     0.0001        1.0        102
2776.45     4567.50       19.0      18.0      5.0     0.00005     0.0001        1.0        103
2564.00     4764.99       20.0      19.0      5.0     0.00005     0.0001        1.0        104
2392.81     4998.15       21.0      20.0      5.0     0.00005     0.0001        1.0        105
2421.49     5237.23       22.0      21.0      5.0     0.00005     0.0001        1.0        106
2227.79     5415.50       23.0      22.0      5.0     0.00005     0.0001        1.0        107
2048.87     5622.12       24.0      23.0      5.0     0.00005     0.0001        1.0        108
1831.88     5794.53       25.0      24.0      5.0     0.00005     0.0001        1.0        109
```

Figure 23 Format of the river data file ‘river1.dat’

Table 5 General format of a river map data file

	File format	**Line**
	River name (character string no longer than one line)	1
	River number, Number of branches (integer, integer)	2
Branch 1	Comment line (character string no longer than one line)	3
	Downstream branch number, Number of spline points in branch (integer, integer)	4
	X, Y, River stage (m), River-bed elevation (m), Channel width (m), Kz (Effluent) (m d^{-1}), Kz (Influent) (m d^{-1}), River-bed thickness(m)	5-37
Branch 2	Comment line (character string no longer than one line)	38
	Downstream branch number, Number of spline points in branch (integer, integer)	39
	X, Y, River stage (m), River-bed elevation (m), Channel width (m), Kz (Effluent) (m d^{-1}), Kz (Influent) (m d^{-1}), River-bed thickness(m)	40-61
Branch 3	Comment line (character string no longer than one line)	62
	Downstream branch number, Number of spline points in branch (integer, integer)	63
	X, Y, River stage (m), River-bed elevation (m), Channel width (m), Kz (Effluent) (m d^{-1}), Kz (Influent) (m d^{-1}), River-bed thickness(m)	64-76
Branch 4	Comment line (character string no longer than one line)	77
	Downstream branch number, Number of spline points in branch (integer, integer)	78
	X, Y, River stage (m), River-bed elevation (m), Channel width (m), Kz (Effluent) (m d^{-1}), Kz (Influent) (m d^{-1}), River-bed thickness(m)	79-95
Branch 5	Comment line (character string no longer than one line)	96
	Downstream branch number, Number of spline points in branch (integer, integer)	97
	X, Y, River stage (m), River-bed elevation (m), Channel width (m), Kz (Effluent) (m d^{-1}), Kz (Influent) (m d^{-1}), River-bed thickness(m)	98-109

5.3.1 Condition when more than two branches join at a single spline point

A problem can occur with the creation of model rivers using ZETUP if downstream branch numbers are defined in a particular manner when more than two river spline branches join at the same spline point. In this case the downstream branch of branch n must be defined as branch n-1 in the river map / spline file. This situation is encountered in Figure 24.

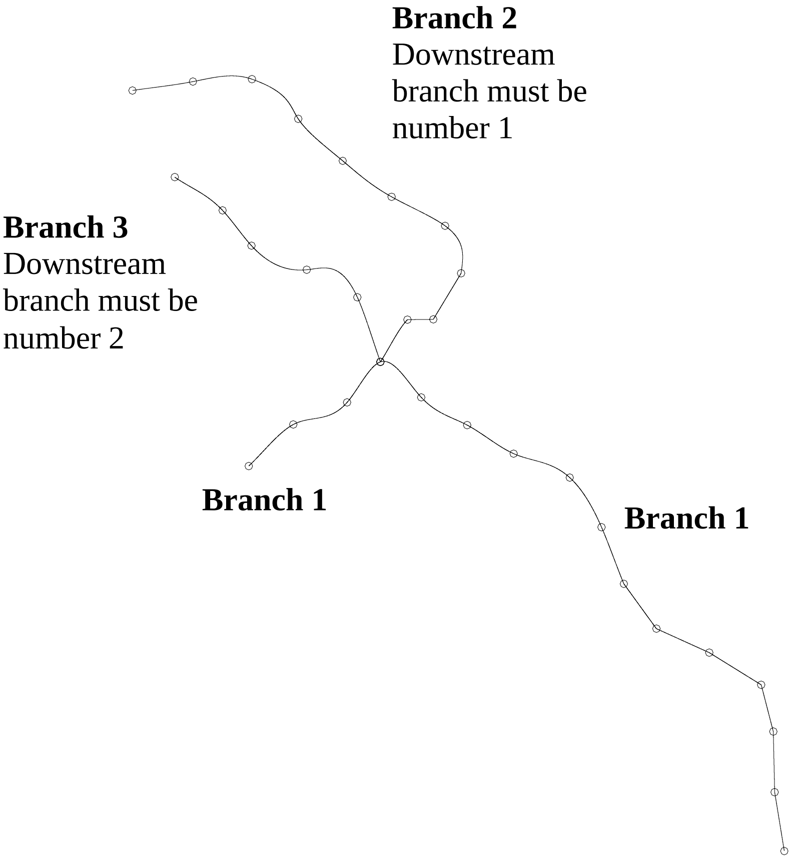
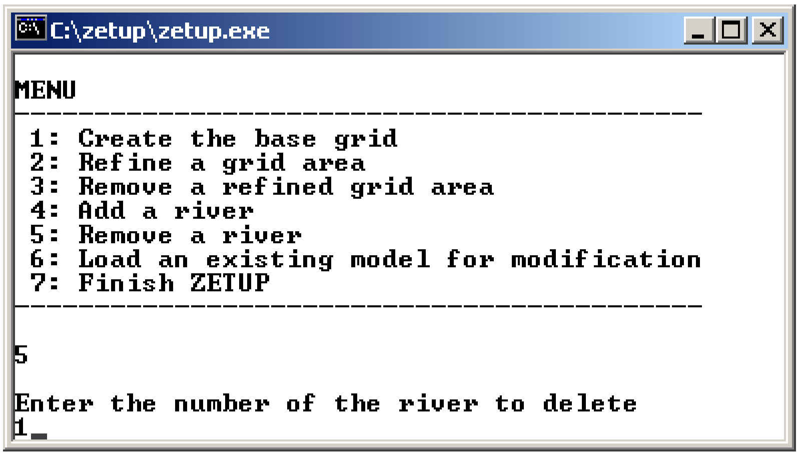


Figure 24 Multiple river spline branches joining at a single spline point

5.4 DELETING A RIVER FROM THE MODEL

Rivers are easily removed from the model. On selection of option 5 from the menu, ZETUP requests the number of the river to delete. This number is the unique identifier for the river, specified on line 2 of the river map input file (Figure 24). The procedure is shown in Figure 25.



```
MENU
-------------------------------------------
 1: Create the base grid
 2: Refine a grid area
 3: Remove a refined grid area
 4: Add a river
 5: Remove a river
 6: Load an existing model for modification
 7: Finish ZETUP
-------------------------------------------

5

Enter the number of the river to delete
1_
```

Figure 25 Deleting a river from the model using ZETUP

6 Output files

ZETUP produces the input files required by ZOOMQ3D that define the structure of the model mesh (‘grids.out’, ‘aquifer.map’ and ‘boundary.dat’) and the structure of rivers (‘rivers.out) which are listed in Table 1. The files with the out extension need to be renamed with the dat extension for input into ZOOMQ3D. ZETUP also produces generic templates of all of the other files required by ZOOMQ3D, which need be modified using a text editor before being used as input to ZOOMQ3D (Table 2). Finally, ZETUP creates a set of files that enable the visualisation and checking of the model structure (Table 3). Detailed descriptions of the files listed in Table 1 and 2, which form input to ZOOMQ3D are presented in the ZOOMQ3D manual (Jackson and Spink, 2004) and consequently, not given here. Each of the files listed in Table 3, which enable the visualisation and checking of the model structure are described next, in individual sub-sections. These do not form input to ZOOMQ3D.

6.1 ‘BOUNDARY.BLN’

ZETUP produces a Surfer (Golden Software Inc., 1994) *blanking file* ‘boundary.bln’. This is used to ‘blank out’ the areas outside the model boundary when contouring the groundwater heads computed by the simulation model, ZOOMQ3D. Information regarding the use of Surfer blanking files is presented in the Surfer User’s Guide (Golden Software Inc., 1994). The use of this file is also discussed in the ZOOMQ3D manual in reference to the visualisation of groundwater head contours.

6.2 ‘BOUNDARY.DXF’

The second output file is ‘boundary.dxf’. This is a standard file format used by CAD software and can be viewed using for example, AutoCAD, ArcView / ArcMap, or Surfer. If these commercial products are not available to the user, dxf viewers can be downloaded from the internet. The file contains a single letter written at each of the nodes on the boundary of the finite difference mesh. These letters represent the *type* of the boundary node. The types are illustrated in Figure 26.

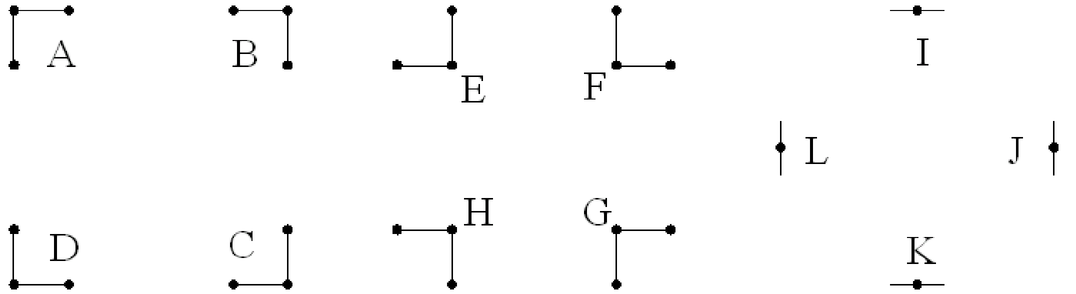


Figure 26 Boundary node types

6.3 ‘BOUNDARY_MAP.ZTP’

The text file ‘boundary_map.ztp’ contains a matrix of characters organised to correspond to the number of columns and rows in the base grid. It is a ‘map’ of the base grid. Nodes on the boundary of the model are indicated by the letter ‘b’. The letter ‘o’ indicates points outside the model boundary. Node objects are not created at these locations. Nodes inside the model boundary are specified by the letter ‘i’. The file allows the user to visually check that the boundary has been defined correctly, however, this is done more easily by examining ‘grid.dxf’.

6.4 ‘GRID.DXF’

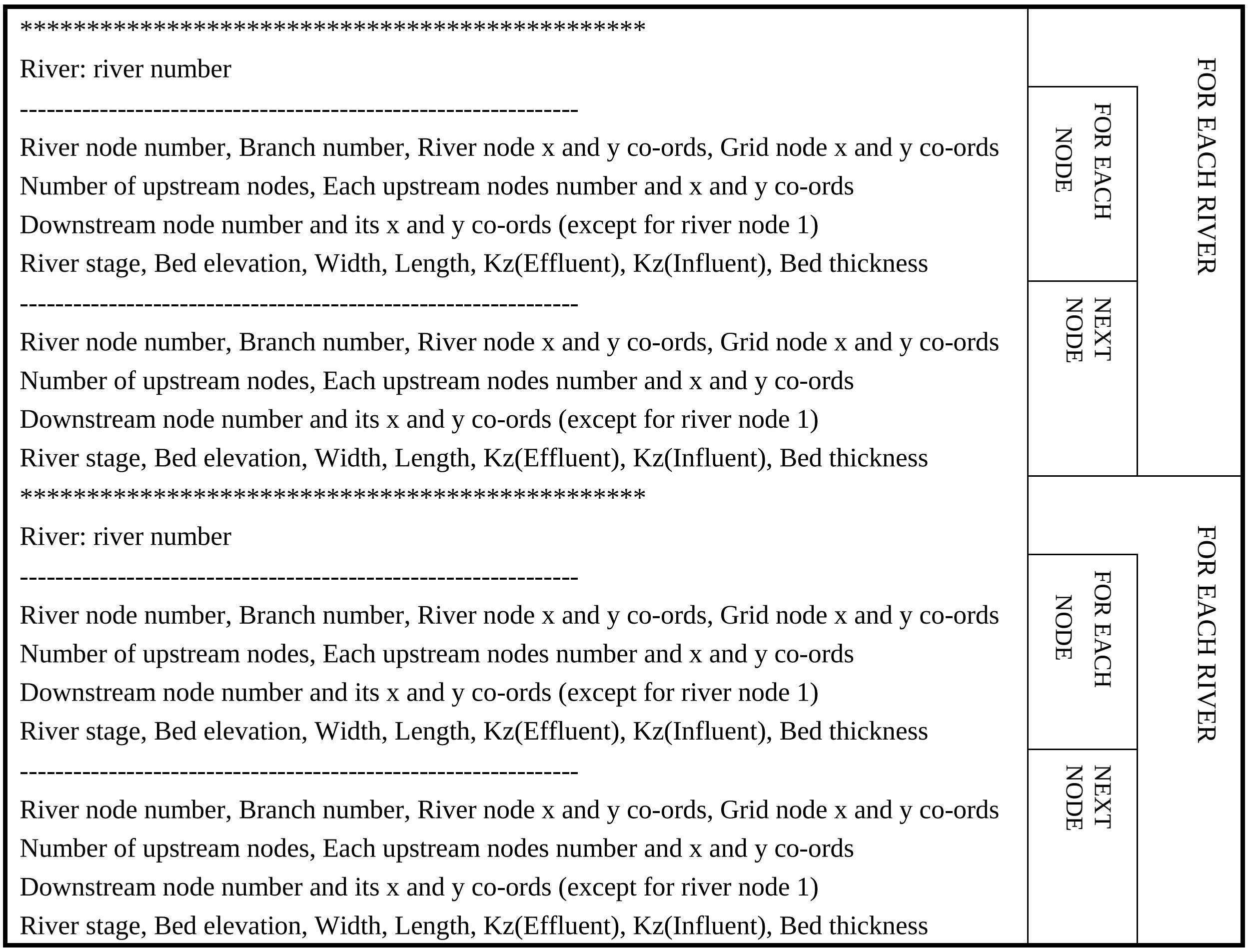
This file enables the user to visualise the structure of the model mesh. It can be viewed using ArcView / ArcMap, Surfer, or AutoCAD, for example.

6.5 ‘RIVER.DXF’

This file enables the user to visualise the structure of the model river, which are composed of a series of river nodes. It can be viewed using ArcView / ArcMap, Surfer, or AutoCAD, for example.

6.6 ‘RIVERCHECK.ZTP’

This text file is output by ZETUP so that the user can manually check that the rivers have been created correctly. The file has the format shown in Figure 27. The co-ordinates of each river node in the file should be the same as the co-ordinates of a grid node. These pairs of co-ordinates are listed next to each other.



```
***********************************************
River: river number
------------------------------------------------------------------
River node number, Branch number, River node x and y co-ords, Grid node x and y co-ords
Number of upstream nodes, Each upstream nodes number and x and y co-ords
Downstream node number and its x and y co-ords (except for river node 1)
River stage, Bed elevation, Width, Length, Kz(Effluent), Kz(Influent), Bed thickness
------------------------------------------------------------------
River node number, Branch number, River node x and y co-ords, Grid node x and y co-ords
Number of upstream nodes, Each upstream nodes number and x and y co-ords
Downstream node number and its x and y co-ords (except for river node 1)
River stage, Bed elevation, Width, Length, Kz(Effluent), Kz(Influent), Bed thickness
***********************************************
River: river number
------------------------------------------------------------------
River node number, Branch number, River node x and y co-ords, Grid node x and y co-ords
Number of upstream nodes, Each upstream nodes number and x and y co-ords
Downstream node number and its x and y co-ords (except for river node 1)
River stage, Bed elevation, Width, Length, Kz(Effluent), Kz(Influent), Bed thickness
------------------------------------------------------------------
River node number, Branch number, River node x and y co-ords, Grid node x and y co-ords
Number of upstream nodes, Each upstream nodes number and x and y co-ords
Downstream node number and its x and y co-ords (except for river node 1)
River stage, Bed elevation, Width, Length, Kz(Effluent), Kz(Influent), Bed thickness
```

Figure 27 Format of the output file ‘rivercheck.ztp’

6.7 ‘RIVERNUM.DXF’

This file enables the user to visualise the river node numbering scheme. The order in which the river nodes are numbered needs to be known when entering data into some ZOOMQ3D input files. It can be viewed using ArcView / ArcMap, Surfer, or AutoCAD, for example.

6.8 ‘SPLINE.DXF’

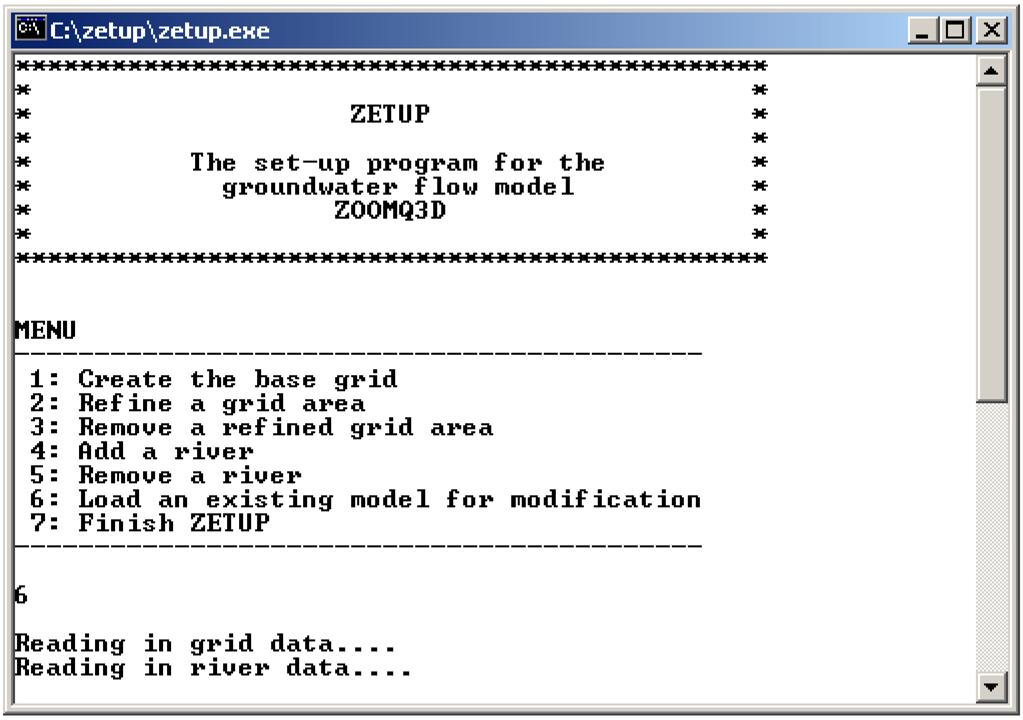
This file enables the user to visualise the river map, or splines, which are used to represent the real geometry of the rivers under consideration. It can be viewed using ArcView / ArcMap, Surfer, or AutoCAD, for example.

7 Reloading a model for modification

To reload a model for modification option 6 is selected from the ZETUP menu as shown in Figure 28. The files ‘grids.dat’, ‘aquifer.map’, ‘boundary.dat’ and ‘rivers.dat’ must exist in the same directory as ‘zetup.exe’ in addition to the river map files containing the spline point data used to create the model rivers. The river map file names are contained in ‘rivers.dat’. The user may need to recall that when ZETUP finishes the files ‘grids.dat’ and ‘rivers.dat’ are created with the .out extension i.e. they are named ‘grids.out’ and ‘rivers.out’. The extension of the boundary file depends on whether the model is being created for the first time (in which case ‘boundary.dat’ is produced) or whether an existing model is being reloaded into ZETUP (in which case ‘boundary.out is created by ZETUP).

After the model has been re-loaded into ZETUP it can be modified. The grid can be refined further or made coarser by removing refined grids and, rivers can be added or deleted. After making the required modifications, the user selects option 7 to finish running ZETUP. ZETUP then writes a new set of all the model files required to run ZOOMQ3D. Instead of overwriting ‘grids.dat’, ‘boundary.dat’ and ‘rivers.dat’ the files ‘grids.out’, boundary.out’ and ‘rivers.out’ are produced. If after the user has checked that the modifications have been made correctly, for example by examining the dxf files, the original files can be deleted and the extensions of the newly created output file changed to ‘dat’ so that they can be read by ZOOMQ3D.

Care must be taken when reloading a model into ZETUP as on completion existing ZOOMQ3D model input files could be overwritten by incomplete ZETUP output files. Consequently, **it is strongly recommended that ZETUP and ZOOMQ3D are not run in the same directory and that their files are kept within separate folders**.



```
***********************************************
*                                             *
*                   ZETUP                     *
*                                             *
*         The set-up program for the          *
*           groundwater flow model            *
*                  ZOOMQ3D                    *
*                                             *
***********************************************


MENU
-------------------------------------------
 1: Create the base grid
 2: Refine a grid area
 3: Remove a refined grid area
 4: Add a river
 5: Remove a river
 6: Load an existing model for modification
 7: Finish ZETUP
-------------------------------------------

6

Reading in grid data....
Reading in river data....
```

Figure 28 Loading a model for modification in ZETUP

References

GOLDEN SOFTWARE INC (1994). Surfer for Windows. User's Guide. Golden Software Inc, Colorado.

JACKSON C.R. AND SPINK A.E.F. (2004). User's manual for the groundwater flow model ZOOMQ3D. British Geological Survey Internal Report, IR/04/40. British Geological Survey, Keyworth.

PRESS W.H., FLANNERY B.P., TEUKOLSKY S.A., AND VETTERLING W.T. (1988). Numerical recipes in C. The art of scientific computing. Cambridge University Press, Cambridge.

Appendix 1 Representing rivers as cubic splines

ZETUP uses cubic spline interpolation to approximate the shape of a real river by a smooth curve, which is fitted through a series of user-defined points along the river. An example of such a curve is shown in Figure 29. The following discussion is based on the description of cubic spline interpolation presented by Press et al. (1988).

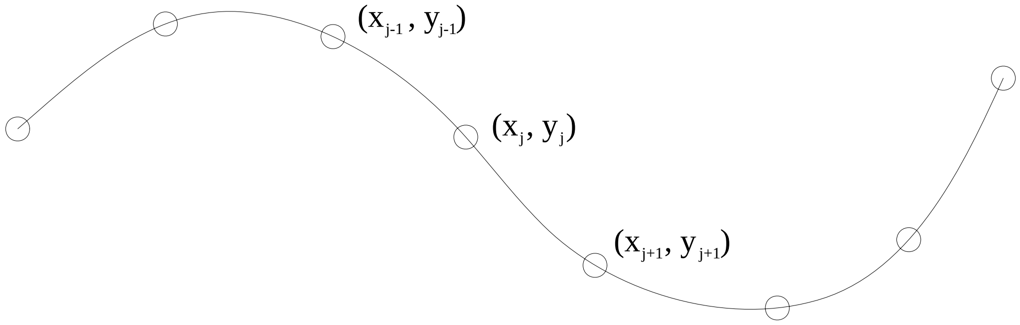


Figure 29 Example of a cubic spline fitted through set of arbitrary points

Suppose the set of tabulated points (x_i, y_i) shown in Figure 29 have been defined. As a first approximation linear interpolation can be used between two points j and j+1. The linear interpolation formula is:

$$y = Ay_j + By_{j+1} \tag{A.1}$$

where

$$A = \frac{x_{j+1} - x}{x_{j+1} - x_j} \quad \text{and} \quad B = 1 - A = \frac{x - x_j}{x_{j+1} - x_j}$$

In this case the second derivative is zero between the two points but becomes undefined at x_j and x_{j+1}. Cubic spline interpolation constructs a curve that has both a smoothly varying first derivative and a continuous second derivative at the boundaries of the interval. This is achieved by first adding a cubic polynomial, whose second derivative varies linearly from y''_j on the left to y''_{j+1} on the right, to the right hand side of Equation A.1.

This cubic polynomial has the form:

$$y = Cy_j + Dy_{j+1} + Ey''_j + Fy''_{j+1} \tag{A.2}$$

where y''_i is the second derivative at point i.

If the addition of Equation A.2 to A.1 is not to spoil the agreement with the values at the fixed points (x_i, y_i), then:

$$C = D = 0,$$

$$E = \frac{1}{6}\left(A^3 - A\right)\left(x_{j+1} - x_j\right)^2,$$

$$F = \frac{1}{6}\left(B^3 - B\right)\left(x_{j+1} - x_j\right)^2$$

Therefore, adding Equation A.2 to A.1 gives:

$$y = Ay_j + By_{j+1} + Ey''_j + Fy''_{j+1} \tag{A.3}$$

Differentiating this once gives:

$$\frac{dy}{dx} = \frac{y_{j+1} - y_j}{x_{j+1} - x_j} - \frac{3A^2 - 1}{6}\left(x_{j+1} - x_j\right)y''_j - \frac{3B^2 - 1}{6}\left(x_{j+1} - x_j\right)y''_{j+1} \tag{A.4}$$

By requiring that the first derivative is continuous across the boundary between intervals the values of y''_i can be calculated. A set of equations is obtained by setting the first derivative at $x=x_j$, evaluated using Equation A.4 in the interval x_{j-1} to x_j, to the first derivative at $x=x_j$ evaluated in the interval x_j to x_{j+1}. For j = 2, …. , N-1 this gives:

$$\frac{x_j - x_{j-1}}{6}y''_{j-1} + \frac{x_{j+1} - x_{j-1}}{3}y''_j + \frac{x_{j+1} - x_j}{6}y''_{j+1} = \frac{y_{j+1} - y_j}{x_{j+1} - x_j} - \frac{y_j - y_{j-1}}{x_j - x_{j-1}} \tag{A.5}$$

This results in N-2 equations in N unknowns, where N is the number of points on the curve. Consequently, two boundary conditions are required to solve for the second derivative values, y''_i. In ZETUP the boundary conditions are obtained by setting the first derivative at the end of the spline equal to the gradient of the line between the boundary point and its neighbour one interval inside the region. The second derivative at the boundary is then calculated using Equation A.4. After the second derivative values have been calculated at each fixed point, the value of y can be calculated for any value of x using Equation A.3.

Whilst the use of cubic splines is considered well suited to the description of rivers because they are often smoothly varying, there are certain situations in which this algorithm fails. If an insufficient number of points are defined along a stretch of river, which is either tightly curving or irregularly meandering, cubic splines can oscillate wildly. However, this is easily rectified by defining more points along the river. A more serious problem occurs when *reversals* occur in the river. For example, consider the section of river shown in Figure 30.

In Figure 30 the x co-ordinates of the spline points do not increase continuously with each step. Instead between points 6, 7 and 8 and 12, 13 and 14, a *reversal* occurs. Unfortunately the algorithm described above only performs correctly if the x co-ordinate increases or decreases continuously with the spline point number. In the case of Figure 30 the curve fitted through the points will be similar to that shown in Figure 31. This is obviously a poor approximation to the shape of the river.

To solve this problem the algorithm is modified in ZETUP. Instead of attempting to spline the whole of the river reach shown in Figure 30 at once, the river is divided into sections. These are shown in Figure 32. Section 1 ends at point 7 and section 2 ends at point 15. Within these sections, either the x or y co-ordinate of the spline points increases continuously with point number. In section 1 and 3, the x co-ordinate of the spline points increases continuously. In section 2 the y co-ordinate increases continuously. Consequently, sections 1 and 3 are splined in the x direction and section 2 is splined in the y direction. That is, in sections 1 and 3 the algorithm solves for d^2y/dx^2 and in section 2 the algorithm solves for d^2x/dy^2. In effect the co-ordinate system is being rotated.

A consequence of this technique of splitting the river channel into sections is that additional conditions must be defined at the boundaries between sections. These additional conditions are based on the first derivative at the section boundary. The boundary condition uses the gradient of the line joining the spline points either side of the section boundary. The appropriate first derivative boundary conditions are shown in Figure 32.

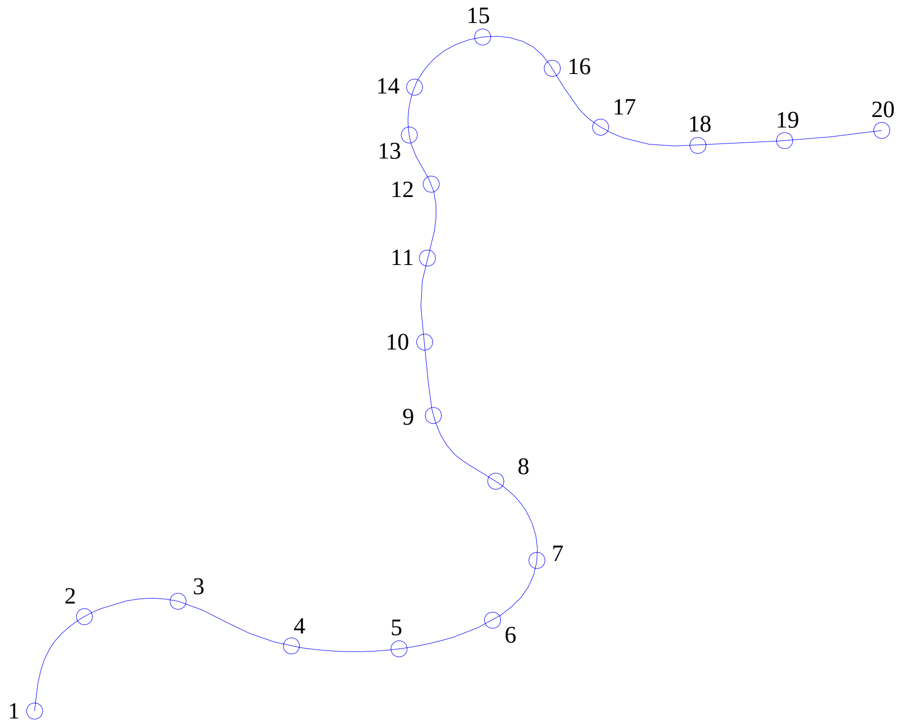


Figure 30 Example river containing reversals at point 7 and 13.

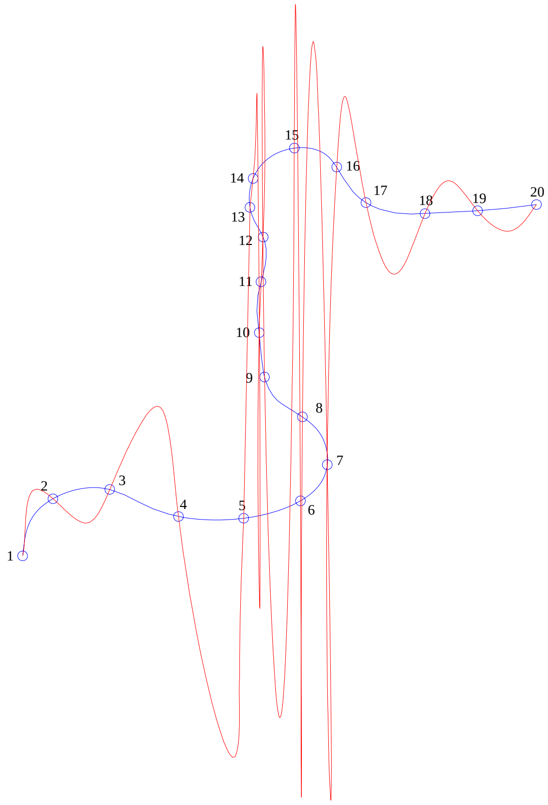


Figure 31 Possible shape of cubic spline when interpolating set of points containing reversals

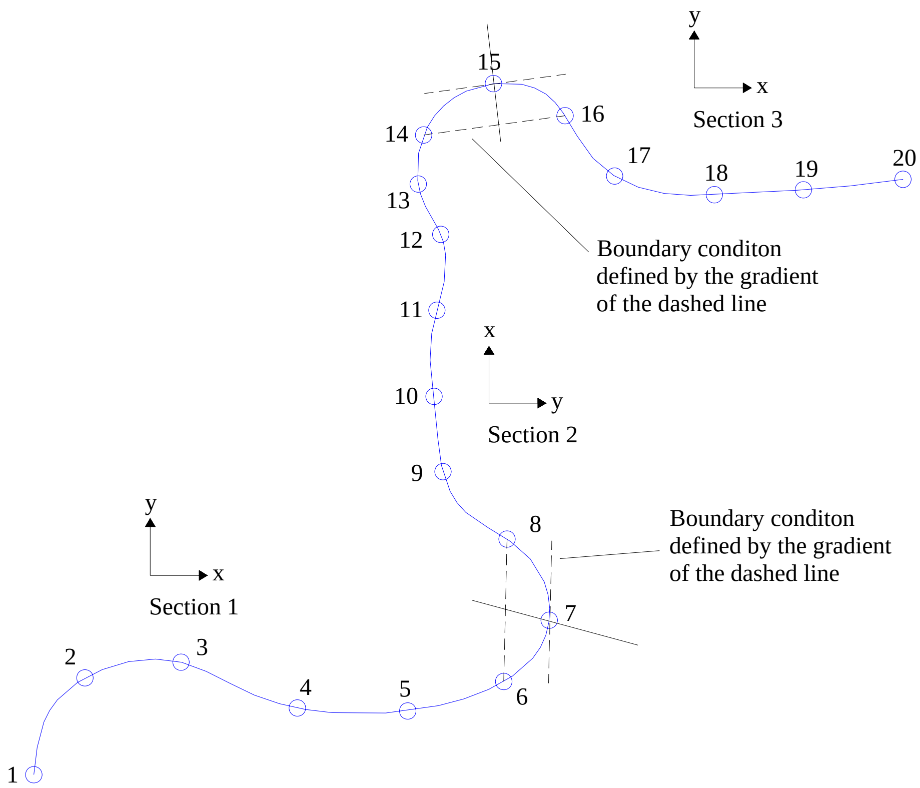


Figure 32 Division of the river channel into sections for cubic spline interpolation