

R2D_Mesh

Mesh Generation Program For River2D Two Dimensional Depth Averaged Finite Element

Introduction to Mesh Generation and User's Manual

by

Terry Waddle and Peter Steffler

U.S. Geological Survey

September, 2002

Introduction

This document describes the R2D_Mesh.EXE program, giving both development background and instructions in its use. Program development is continuing and this document will be updated periodically to reflect current changes and additions.

Program Purpose

The purpose of the R2D_Mesh program is to provide a relatively easy to use but effective mesh generation facility for two-dimensional depth-averaged finite element hydrodynamic modeling. Essentially, a bed topography file, containing pointwise elevations and roughnesses over the reach of interest, is taken as input to the R2D_Mesh program. The points can be independent or connected in breaklines or featurelines. A finite element mesh is defined interactively and graphically by the user with the aid of various tools. Finally, when the user is satisfied, an input file for the River2D finite element hydrodynamic model is generated. This file is complete and the flow solution may proceed directly although some changes to the default run options may be desired.

The R2D_Mesh program is not a fully self-contained data preprocessor. There is no facility for modifying the input topography (to delete an erroneous data point, for example). The interpolation of the topographic input data to the finite element mesh is on the basis of a constrained Delauney triangulated irregular network. Therefore it is usually desirable that the raw field data be processed through the R2D_Bed program or a GIS application to the point where an adequate bed topography model is achieved.

Program Status

At the present time, the R2D_Mesh program generates data files for the River2D hydrodynamic model only. The functionality of the generator is basic, but adequate for effective mesh generation. The code is not completely debugged, but problems occur only in unusual circumstances. The generated mesh should be saved often to minimize the impact of program crashes.

Requirements

The R2D_Mesh program runs on any Pentium processor running a 32-bit version of Windows (including Windows 95). Memory requirements vary somewhat with the size of the input topography file and the desired resolution of the finite element mesh. Generally, 32 megabytes are adequate, though very large meshes will require more memory. These requirements are related more to general Windows operation than to specific R2D_Mesh needs. The R2D_Mesh program is computationally intensive and interactive response will benefit significantly from having a faster processor present.

Installation

The R2D_Mesh program is contained in the R2D_Mesh.exe file which is about 650 KB in size. Simply copy this file to a convenient directory. In addition, a simple set of example files is contained in the example.bed, example.msh, and example.cdg files. These files are extensively commented to explain their content and format.

License

The initial development of the R2D_Mesh program has been supported in part by Fisheries and Oceans Canada and the Department of Civil Engineering of the University of Alberta. The copyright is retained by these parties and the authors of the program. This software may be freely used, copied and distributed as long as the copyright notice and this license are included without alteration. Later modifications of the program have been provided by the U.S. Geological Survey. The U.S. Government does not claim copyright to technical works created at government expense.

Overview of Use

The normal course of events in using the R2D_Mesh program is described in this section. The basic procedure is to input a bed topography file which then appears in the mesh generation window

as a contour map. The user then draws a boundary outline within the map to delineate the modeled region. Optionally, a default boundary may be specified with the input topography file. The boundary geometry consists of an enclosing polygon and any number of internal polygons representing islands. Discharge and stage boundary conditions are defined as desired. The defined boundary is then discretized by subdividing it into a number of mesh points or “nodes”. Then the interior of the modeled region is filled with nodes by various means.

Any combination of uniform, radial, area, and region fills can be used as well as inserting individual fixed or floating nodes. There is also a facility to implement breaklines, which guarantee interpolation along linear features. Each inserted node takes its bed elevation and roughness values by linearly interpolating from the underlying topography model. The inserted nodes can be triangulated at any time. When the desired density pattern is achieved, the overall mesh is smoothed (by adjusting nodal positions – elevations and roughnesses change correspondingly) one or several times to regularize the triangle shapes and give more gradual transitions between areas of different node density. Finally, the generated mesh is output in the form of an input data file that contains all of the necessary components to be run by the River2D hydrodynamic model program.

The following figures illustrate the layout of the R2D_Mesh window and certain commands that are available as icons on the toolbar at the top of the window. Throughout this document references to certain commands are followed by the word “icon” in parentheses. This indicates that the particular command also has an icon on the toolbar. Readers may prefer to keep Figure 2 close at hand when first using this document.

Figure 1. R2D_Mesh window showing bed contours in color and a partially completed mesh.

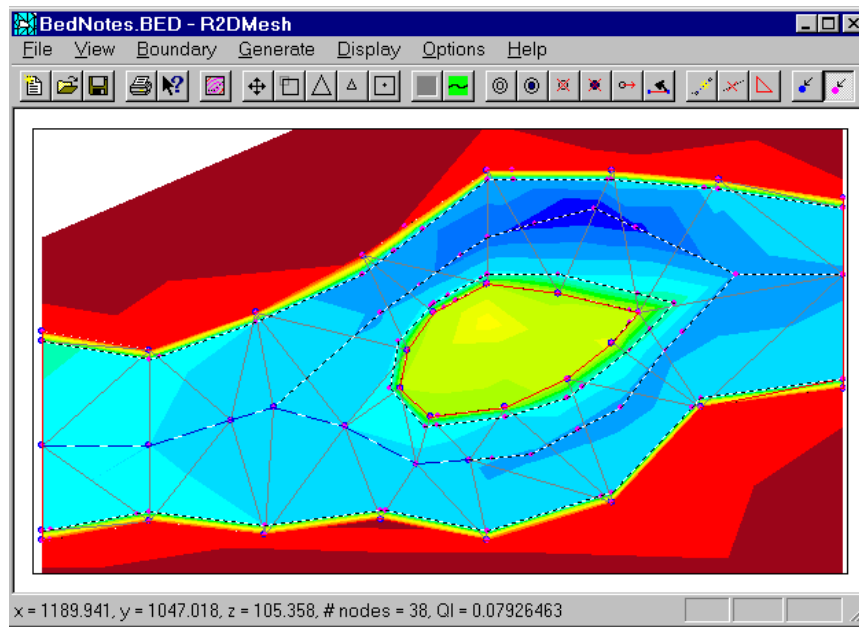
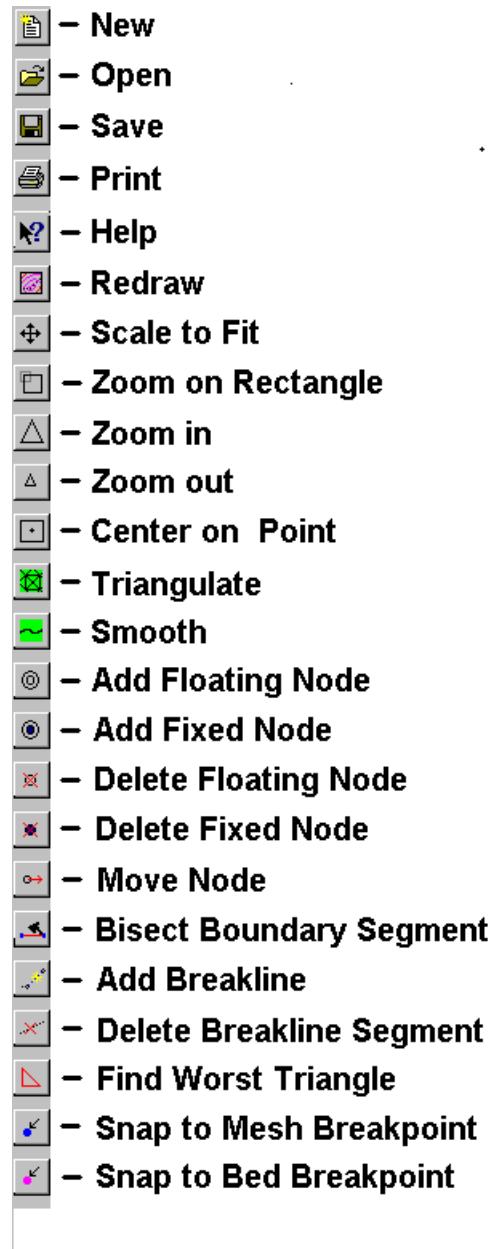


Figure 2. Icons used in R2D_Mesh



Getting Started

The following section describes the general procedure and options available for generating a computational mesh using the R2D_Mesh program. While it is not a formal tutorial, it is advisable to have the program running and experiment with the

commands and options as they are introduced. The bed topography file, “Bednotes.bed”, is available for this purpose.

When the program starts up a single blank window titled “Untitled – R2DMesh” appears. When a mesh file is loaded or named during a save operation, the term “Untitled” will change to the mesh file name. This window will eventually display the bed topography and the overlain finite element mesh. As it is the only permanent window generated by the program, it is advisable to maximize it by clicking the mouse in the appropriate box in the top right-hand corner.

Overlay of Bed and Mesh

R2D_Mesh uses a procedure by which the computational mesh is generated as an overlay on the bed topography. Several functional procedures are applied to the points in the bed topography and to the computational mesh as the nature of the data in these overlays are similar. Thus, for example, the triangulation algorithm used on both overlays is the same. When the computational mesh is generated, values for nodal elevation and roughness height are interpolated from the bed topography. It is useful to keep the distinction between the bed topography and the computational mesh overlay in mind when reading the next sections.

Inputting the Bed Topography File

The first step the user needs to perform is to input a bed topography file. The “**Open Bed File**” (Open icon) command under the “**File**” menu brings up the standard Windows file open dialog. The bed topography file must have a .bed (or .BED) extension to be accessible. Navigate to the appropriate directory and double-click on the BedNotes.bed file name.

The bed topography file contains a pointwise listing of elevations and bed roughnesses. Each line in the file corresponds to a single measured or inferred point and consists of a point number (integer), x-coordinate (floating point number), y-coordinate (floating point number), bed elevation (floating point number), bed roughness height (floating point number), and an optional code (up

to twenty alphanumeric characters) all separated by any number of spaces or tabs. Any consistent unit system may be used in the mesh generation but the River2D program expects all length dimensions in meters. Ordinary plain text output (tab or space delimited) from a text editor, word processor, or spreadsheet is entirely suitable (remember to use Save As Text, as appropriate). The file may contain any number of comments as long as the comments do not contain numerical digits, periods, or any form of brackets. Also, no comments should appear within the set of numbers defining a point. Blank lines are ignored as well. The end of the list of data points should be indicated with a period (which may be part of a comment, such as “end of data points.”).

Breaklines or featurelines can be specified in the input topography file as well. The effect of defining a breakline is that the resulting triangulation is constrained to interpolate linearly along the breakline. To insure this property, all breakline segments must coincide with triangle sides. Breaklines are most useful to obtain reliable representations of linear features with a minimum number of points. The top and bottom of river banks are common examples. The R2D_Bed program provides a graphical user interface for topography file development and refinement of the final topography model and is the recommended topographic data preparation tool for most applications. See the R2D_Bed documentation for more information.

The topography data may also be prepared using a text editor. There are two methods to designate a breakline in the input topography file. The first, and simplest, method is to enclose the points forming the breakline in brackets. Curved brackets “(…)” indicate an “open” breakline, square brackets “[…]”, indicate a closed breakline, which starts and ends at the same point like a polygon, and curly brackets “{…}”, are used to delineate default computational boundaries.

The second method of designating breaklines is to explicitly specify the segments forming the breakline. After the period signifying the end of the data point specifications, these segments can be entered. Each one consists of a segment number (integer)

and the numbers (two integers) of the points at each end of the segment. These points must have been previously entered either as individual points or as part of another breakline (using brackets). This is the method employed by the R2D_Bed program.

When all desired segments have been entered, the section is terminated with a period. Comments are allowed in this section of the data file as in the data point section. The segment method of specifying breaklines is more cumbersome but allows more complex topologies than simple open and closed paths.

The R2D_Mesh program performs a simple comparison check for crossing breaklines. In case a crossing is encountered, the first entered breakline takes precedence. The subsequent offending segment of the crossing breakline is simply ignored. Generally, if breaklines cross, there is likely a problem with the input data that should be investigated.

Breaklines are implemented in R2D_Mesh by a recursive segment bisection procedure. Essentially, each segment is checked and if it is not co-incident with a triangle edge, then a new point is added within the segment, forming two new segments. The new point is incorporated into the triangulation and then the new segments are rechecked and further subdivided if necessary. Overall, this procedure results in a refined nodal distribution along the breakline to roughly match the overall point distribution in the vicinity of the breakline.

The bed topography input process involves a number of stages. First, the data is read into the program. A check for duplicate points (within 10^{-5} units) is made as each point is input. The second stage is triangulation of the input topography data. After triangulation of the given points, the breaklines are threaded through the triangulation of the topographic data.

The last stage of the input process is contouring the topography. Contour lines are generated on an element by element basis for the entire mesh. If this step appears to take an excessively long time, or the resulting map is seriously distorted, a problem with the input

data is likely. In particular, an incorrect bed elevation value may be causing the contour range to be excessive. A missing or extra number in the file will cause the input sequence to be disrupted, and horizontal coordinates may be taken for vertical elevations.

Inputting the Ice Topography File

Ice topography files are used when modelling ice-covered domains. They have the same format as bed topography files except that bed elevation and bed roughness height are replaced with ice thickness and ice roughness height, respectively. Ice topography files are input using the “**Open Ice File**” command found under the “**File**” menu. Unlike bed topography, it is not necessary to open the relevant ice topography file when developing the computational mesh. This is because nodal ice properties are not interpolated in the R2D_Mesh. They are only interpolated upon inputting the ice topography into River2D.

Breaklines can be specified in the ice topography file to define linear features such as the edge of an ice cover. To represent ice features accurately in the computational mesh, it is advisable to input corresponding breaklines in the mesh. In these circumstances, it is helpful to input ice topography into R2D_Mesh so that it may be used as a visual aid in the placement of mesh breaklines.

Triangles in ice topography can either be ice-covered or ice-free. A triangle is considered to be ice-covered if all vertices that make up the triangle have ice thickness values that are greater than zero. Otherwise the triangle is considered to ice-free. The transition from ice to no ice and vice versa is defined in this manner so that it easy to locate ice edges in the ice topography. If ice parameters were interpolated of all triangles, then ice edges would be smeared over the area of a triangle.

Viewing the Display

All display options may be accessed by a right click anywhere in the graphical display window. This opens a “Display Options” dialog box. For each of the display options menu items described

below, there is a corresponding item in the display options dialog so further description of the “Display Options” dialog will be omitted.

Bed topography breaklines are shown as black dashed lines. They can be turned off using the “**Breaklines**” toggle item under the “**Display**” menu. The default computational boundary, if defined, is outlined in red. The drawing is scaled to fit the window. The default contour spacing is 0.5 vertical units. The “**Set Contour Interval**” selection under the “**Display**” menu allows this to be changed. The contour map can be used as a guide for boundary and fill specification. The information box on the bottom of the screen shows the x and y position of the mouse pointer in map units. If the cursor is within the data mesh, the bed elevation of the point is also indicated. The elevations of the various contour lines can thus be determined by simply pointing at them with the cursor. The status line also indicates the number of nodes and the QI value (Mesh Quality Index) of the generated mesh (defined below). As there is no generated mesh to begin with, these values start at zero.

The window can be zoomed at any time to allow a closer look at some part of the mesh. To do so, select the “**Zoom on Rectangle**” (icon) item under the “**Display**” menu. Then move the cursor to one corner of the area you wish to expand. Press and hold the left mouse button and move the cursor to opposite corner. A red “rubber-band” rectangle is drawn and updated as the cursor moves. Zoom by rectangle may also be initiated by clicking the icon showing two superimposed rectangles. When the mouse button is released, this rectangle is expanded to fill the full window. Note: the red rectangle mysteriously does not appear on Windows NT machines.

Since an aspect ratio of unity is enforced, the display may show more area than was selected in one or the other direction. The zoom operation may be repeated an unlimited number of times. Zooming back out (all the way) is accomplished by selecting the “**Scale To Fit**” (icon) command under the “**Display**” menu or the icon containing four arrows on the toolbar. Incremental zooming can be accomplished using the “**Zoom on Point**” (icon) and

“**Zoom Out**” (icon) commands under the “**Display**” menu. These are indicated by icons containing a large and small triangle respectively on the toolbar. To zoom in on an area, click the “**Zoom on Point**” icon and then click the center of the area to be enlarged. To zoom out, click the “**Zoom Out**” icon and the display will zoom out while retaining the same center point. Combinations of the three zoom commands may be used to view any area of the topography at any desired scale.

The display can be shifted, or panned, at any time by using the “**Center on Point**” (icon) command under the “**Display**” menu or by clicking on the icon with a dot in the middle of a square. After selecting the command, simply click the left mouse button while the cursor is pointing to the desired new center point. The display will then shift, retaining the same scale.

In addition to these display features, the “**Display Options**” dialog has toggles for a “**Colour Map**” of bed elevation or ice thickness; adjusting the “**Colour Range**” to a fixed range or automatic scaling; displaying a “**Colour legend**”, plot axes, generated mesh, mesh nodes, mesh breaklines, large elevation differences, triangles with a quality index less than a target value, node numbers and node display size. It is useful to toggle the “**Display Options**” on and off to see the display effects provided by each.

Colour maps of ice thickness are a slightly different than colour maps of bed elevation. The main difference is that ice thickness is displayed using shades of blue rather than a colour spectrum. For a given range, the highest values of ice thickness will be close to white while the lowest values (greater than 0) will be blue. Regardless of the range (automatic or fixed), ice-free elements will be coloured a solid blue (darker than any blue used to colour ice-covered elements) to represent the open water areas.

Defining the Boundary

The data mesh, represented by the contour map, shows the overall area defined by the limits of the topography data. The area is always “convex” in the sense that any line drawn between any two

points will always be entirely contained in the area. In many cases, large areas are contained for which no measured data has been taken and there is no interest in modeling. In addition, the area that is to be modeled may be only a sub-area of the whole. Because of these considerations, it is necessary to define the boundary of the area to be modeled. This can be done in the input topographic data file with the curly bracket breakline specification, by using the boundary definition tools in the R2D_Bed program, or it can be done as the first step in using the mesh generator. In any event, the external boundary must be defined before any further mesh generation steps can take place.

If the boundary defined in the topography data file is appropriate, then proceed to the boundary condition specification stage. If it is not appropriate, then select the “**Clear Boundary**” command under the “**Boundary**” menu to remove it.

In R2D_Mesh, the boundary is defined by graphically “drawing” a polygon around the area to be modeled. First, select the “**Define External Boundary**” command under the “**Boundary**” Menu, then position the cursor and click boundary node positions. The cursor changes to a cross-hair after the first position is clicked. Then move to the next point, keeping the modeled area to the left of the line generated (counter-clockwise direction). Continue clicking the left mouse button on any point on the desired boundary. Close the boundary polygon by clicking the mouse button very close to the starting point. The selected boundary is then shown in red, superimposed on the contour map.

Note that the program does not allow selection of boundary points outside the topography data area. When the cursor is placed outside of the topographic data area, the elevation is not displayed on the status bar so you may readily determine if the cursor is within the useable topography.

If desired, the defined boundary can be cleared with the “**Clear Boundary**” command under the “**Boundary**” menu. A new boundary can then be defined. See the **Modifying an Existing Boundary and Mesh** section below.

Internal boundary polygons are specified with the “**Define Internal Boundary**” command. These boundaries are defined in the same way as the external boundary. Care should be taken during this process not to select points outside the external boundary. *No checks are performed and the program will crash if this condition is violated.* Note that internal boundaries can be added to a mesh at any time, even after internal points have been added and triangulated. Once added, they cannot be removed without clearing the entire mesh and boundary.

Ideally, the external boundary should be set at the water’s edge at the highest flow to be modeled. Exact placement is not necessary if a conservatively high location is selected. The use of internal boundaries is optional in the sense that the River2D program will accommodate “dry” (above water surface elevation) areas. The choice between including or excluding islands depends on preference, convenience, efficiency (fewer nodes and less connectivity if excluded), and flexibility (ability to model varying discharges with the same mesh).

Specifying Boundary Conditions

The red color of the drawn boundary indicates the default boundary condition, which is a “no cross-flow” or “vertical wall” condition. That is, the velocity field may have a component parallel to the wall, but not perpendicular to the wall. Generally, most of the lateral boundaries in a stream flow problem should be of this type.

A specified total discharge across a boundary segment is defined using the inflow condition. It is simplest (but not necessary) if the boundary has been defined so that a single boundary segment covers the complete inflow section. Select the “**Set Inflow**” command under the “**Boundary**” menu. Point at the desired boundary segment and click the left mouse button. A dialog box appears asking for the inflow discharge for this segment. If there is more than one inflow segment, each must be defined separately with its own discharge. Selecting “**Set Inflow by Area**” under the “**Boundary**” menu allows the user to draw a “rubber band”

rectangle around a group of boundary segments and set the discharge for all selected segments at once. The River2D program will later sum and reapportion the total among the actual elements forming this boundary. The apportionment across the inflow section is based on the relative conveyance, depending primarily on flow depth. Note that the boundary is redrawn with the inflow segment(s) shown in green.

A fixed downstream water surface elevation boundary condition is specified using an outflow condition. Select “**Set Outflow**” from the “**Boundary**” menu and point and click on the desired outflow segment. A dialog appears asking for the outflow water surface elevation. As with the inflow boundary condition, selecting “**Set Outflow by Area**” under the “**Boundary**” menu allows the outflow elevation for a group of outflow boundary segments to be set at one time. When defined, outflow segments are shown in blue.

Any segment defined to be inflow or outflow can be changed back to a no flow boundary by selecting the “**Set Noflow**” command under the “**Boundary**” menu. A “**Set Noflow by Area**” command is provided under the “**Boundary**” menu. The boundary conditions can be set at any time during the mesh generation process. It is easiest to do it first, however, because the number of boundary segments is smallest at this stage.

The usual modeling situation is to set the discharge at the inflow section and the elevation at the outflow. Mathematically (assuming subcritical flow at the boundaries), it is possible to use different combinations (one condition at each flow boundary). For example, setting both inflow and outflow elevations is possible and useful in some practical cases. The resulting discharge through the system is then a solution variable. The River2D model also supports a depth-discharge intensity relation boundary condition, but this must be set using the “**Set Boundary Conditions**” command in the River2D program.

Discretizing the Boundary

Now that the boundary is defined and the boundary conditions are specified, the actual discretization points can be inserted. Normally the boundary is discretized first. Select the “**Boundary Nodes**” item from the “**Generate**” menu. A dialog appears which asks for the desired spacing. The default value of 1000 meters is generally adequate as intermediate points will be automatically generated as required.

After entering this value, the previously defined boundary will now show points at each point of inflection on the defined boundary no matter how close they may be. Since each boundary segment that exceeds 1000 meters (or the specified value) in length is subdivided into equally spaced intervals, the spacing may not be exactly what was specified. The appropriate boundary conditions and other parameters are automatically transferred from the boundary definition and the topography data to the new points and segments.

Boundary nodes should be generated only once. Boundary discretization and location can be changed as described in the **Modifying Boundary and Mesh** section below. The “**Clear Mesh**” command under the “**Generate**” menu erases the mesh but leaves the internal and external boundaries with boundary nodes yet undefined. Generally, a fairly coarse initial boundary spacing is best (hence the 1000 meter default). Subsequent automatic refinement will occur only where required.

Filling the Interior with Nodes

The key feature of the R2D_Mesh program is its ability to triangulate any arbitrary distribution of input nodes into a usable mesh. Various tools are used to interactively insert distributions of nodes or individual nodes into the bounded domain and then the program builds the mesh of triangles.

Generally, the objective is to provide a high node density in critical areas while having lesser densities in unimportant or slowly varying areas. Higher node density means finer resolution and

greater accuracy while lower node density means less computational effort and calculation time. The biggest advantage of the finite element method is its ability to dramatically vary the discretization resolution over the computational domain. It is, however, up to the user's judgement to determine the criteria for varying the resolution.

The most basic fill option available is the **Uniform Fill**. Since this option places nodes throughout the domain, the spacing chosen is usually the coarsest desired. The dialog box first asks for a spacing and then for an angle. The spacing is used to set the location of the nodes such that all nodes will be equidistant from each other in an equilateral triangular pattern. The angle (between 0° and 90°) is the pattern angle from the horizontal (x) direction. The nodes will be laid out on lines parallel to this direction. The generated nodes are automatically screened to lie inside the defined boundary. Again, they automatically acquire attributes from the underlying topography mesh.

The other node fill options provide a means for increasing the node density in various locations of interest. **Area Fill** allows the user to select a rectangular sub-area which is filled with a uniform density of nodes in a similar fashion to the uniform fill. **Region Fill** allows the user to define an arbitrary polygon which is uniformly filled. The polygon is defined in a counter-clockwise direction or with the inside to the left of the line. The **Region Refine** command places a single new node at the center of each existing element within the region polygon defined by the user. After triangulation this area will have a node density of roughly twice the previous density. The region is defined in exactly the same way as for a region fill. The region must be defined in a counter-clockwise direction.

Radial Fill gives a concentric pattern which is very dense near the specified centre with diminishing density further away. Individual points can be inserted at will using the “**Add Floating Node**” (icon) and “**Add Fixed Node**” (icon) commands found under the “**Generate**” menu. The floating nodes may move during the smoothing process. Fixed nodes can be used to define sharp interfaces at fixed locations. These fixed nodes should be used

sparingly and inspected carefully, however, as they do not benefit from the smoothing process. Note that all points generated by the pattern fills are floating points.

Breaklines in the mesh can also be defined. After selecting the “**Breakline**” (icon) command found under the “**Generate**” menu, point and click on the points forming the breakline. Select any “**Generate**” menu item to get out of breakline insertion mode. Note that the mesh may be zoomed or recentered or scaled to fit during breakline definition. The points entered as the ends of breakline segments are fixed points and are displayed as blue dots.

Breaklines can be used to define ice features such as the edge of the ice cover in a partially ice-covered domain. If breaklines are to be used for this purpose, it is advisable that breaklines be placed slightly to the “ice side” of an ice edge. This is to ensure that ice properties are interpolated appropriately when the ice topography is input into River2D.

When triangulated, the breaklines will show as blue dashed lines. The triangulation will reflect the extra points which are generated by the breakline process. These extra points generated on breaklines are called sliding nodes as they may move along the breakline during the smoothing process. This triangulation should be smoothed and carefully inspected. Additional floating nodes may be inserted or deleted in areas where the triangle shapes are poor or where there are abrupt changes in triangle size. Sometimes, after smoothing, deleting a few of the automatically generated nodes will improve the mesh quality as well.

The “**Snap to Mesh Breakline Node**” (icon) command (found under the “**Options**” menu) allows the connection of new breaklines to existing breakpoints. A mouse click within 6 pixels of an existing mesh breakline point will use the existing point instead of creating a new point. For example a loop can be constructed by selecting “**Snap to Mesh Breakline Node**” and clicking on the first point again after all other points on the breakline have been created. An existing breakline can be extended

or a new branch added. It is best to toggle this command off after the snapped connection is made.

The “**Snap to Bed Breakline Node**” (icon) command is intended to allow important bed topography breaklines to be copied into the generated mesh. When the snap to bed option is on and the define breakline generate command is active, bed breakpoints will be highlighted. Clicking within a few pixels of a bed breakpoint will generate a mesh point at the exact coordinates. Care should be taken to insure that the correct point is chosen, as several points may be very close together. Note that in addition to the breakpoints in the bed file, new points generated by the triangulation process may also be present. These new points will be co-linear with the original points and need not be selected for an accurate representation.

All fill nodes which have not been triangulated can be cleared at any time with the “**Clear Untriangulated Nodes**“ command under the “**Generate**” menu.

Triangulating the Mesh

The inserted nodes can be triangulated at any time with the “**Triangulate**” (icon) command under the “**Generate**” menu. This command invokes a constrained Delauney triangulation routine which gives the “best” possible triangles. That is, it generates triangles which are as close as possible to equilateral. The constrained nature of the triangulation is necessary to insure that no triangle oversteps the defined domain boundary. The resulting triangles are displayed overlaid on the topography map. More triangles may have been generated than are displayed since a complete “convex hull” is generated. Depending on node placement, the resulting triangulation may be more or less acceptable. Usually it requires one or more smoothing steps, evaluation of the quality index and evaluation of the accuracy to which the underlying bed is represented in the computational mesh is described later.

Smoothing the Mesh

To make the triangles more regular in shape and to give a more gradual transition between triangles of different sizes, the mesh should be smoothed. Select the “**Smooth**” (icon) command from the “**Generate**” menu. The smoothing process moves each point to a more central position with respect to neighbouring points, as defined by the triangles. After the points have been moved, the entire mesh is re-triangulated, to insure the best possible triangulation. The smoothing process may be repeated as often as desired. The mesh will become smoother and more regular, but the discretization contrast will gradually diminish.

The smooth command causes the floating and sliding nodes in the mesh to move to new positions using a combination of element shape and topographic representation as criteria. The direction of the node movement will tend to be toward the adjoining triangle with the largest elevation difference between the bed and the mesh. This difference is measured at the center of the element. Points will tend to gravitate toward breaks in slope, in some cases acting like flexible breaklines. A parameter controlling the relative importance of the topographic criteria can be set in the “**Options**” menu. Select “**Set Topo Smoothing Parameter**” and enter a value between 0.0 and 1.0. A value of zero indicates no topographic effect and a value of one indicates no consideration for element shape. With higher values of this parameter, larger differences in adjacent element size are allowed. The default value is 0.5.

The QI value displayed on the status line is a mesh quality index which may be used as a rough guide. The number presented is the minimum “triangle quality” value for all triangles generated. The triangle quality is defined as the ratio of triangle area to circumcircle area (the circle which passes through the three points defining the triangle) normalized to the corresponding ratio for an equilateral triangle. Thus, an ideal mesh (all equilateral triangles) would have a QI of 1.0. Real meshes will have a QI of less than one. Typical acceptable values may be in the order of 0.15 to 0.5.

Smoothing will usually increase the QI dramatically, if it is very low. Once in the acceptable range, improvement in QI with subsequent smoothing may be fairly slow. The QI may even decrease slightly, especially if breaklines are present. The worst triangle in the mesh, which governs the mesh QI value, is displayed with a red outline. Sometimes the poorest triangle may be small and hard to find. An icon containing a red triangle is provided to aid in finding the worst triangle. When clicked the display moves to the triangle with the lowest QI and performs a “Zoom In on Point” operation.

During the smoothing process, the boundaries receive special attention. If a long thin triangle ($QI < 0.5$) has one (long) edge on the boundary, a new boundary point is inserted to break the triangle into two. Repeating the smoothing process will insert additional points until the resulting triangles are reasonably close to equilateral. Thus a fine discretization in an area near a boundary will automatically lead to a correspondingly fine discretization of the boundary itself.

As the smoothing process is an iterative one (including insertion of additional points on boundaries and breaklines) it may take several trials. The number of nodes indicated in the status bar will increase as needed to improve triangles on breaklines and on the boundaries. When the number of nodes on the status bar is no longer changing, additional smoothing alone will not improve the QI.

Points automatically generated by the program along either boundaries or breaklines are able to “slide” during the smoothing process. That is, they adjust their positions as ordinary points except that they are constrained to move only along the line segment on which they were generated. To insure line integrity, they are also constrained in the distance that they may shift.

After triangulation and smoothing, additional points may be inserted by any of the fill methods. The additional points may then be incorporated into the mesh by subsequent triangulation and smoothing.

Individual triangulated nodes may be deleted at any time, using the “**Delete Fixed Node**” (icon) and “**Delete Floating Node**” (icon) commands in the “**Generate**” menu. Select the desired deletion mode and point and click the left mouse button on the offending node. A red “X” will appear across the node. The node will actually be deleted the next time the mesh is triangulated. Often, the quality and appearance of a mesh can be improved more quickly and easily by the deletion of a few “stray” nodes compared to inserting more nodes. Of course, fewer nodes leads to faster River2D program execution. Note that independent “fixed” nodes, “floating” nodes, and “sliding” nodes can easily be deleted. Deleting “fixed” nodes used to define breaklines requires deleting the adjacent breakline segment. See the **Modifying an Existing Boundary and Mesh** section for more on moving and deleting nodes once the mesh has been generated.

Viewing the Generated Topography

Select the “**Generated Contours**” item under the “**Display**” menu. The triangulation on the screen will be replaced by another contour map which overlays and is hopefully similar to the original input bed topography contours. This new contour map is a representation of the topography which has been transferred to the generated mesh. Since the points in the generated mesh are not at the same locations as in the original mesh, there will be some difference.

The “**Highlight Large Dz Triangles**” command in the “**Display**” menu will cause all triangles with an elevation difference larger than a set threshold value to be coloured yellow. The elevation difference is defined at the center of the triangle and is the difference between the bed and mesh elevations at that point. Since all mesh nodes take their elevations from the underlying bed elevations, their elevations are exact. However, since the mesh nodes are not at the same location as the bed nodes, the interpolations between the mesh nodes differ from the bed values at the mesh element center. Usually, but not always, the maximum difference is in the center of the triangle.

The highlighted triangles are a guide to suggest areas in need of further refinement. Normally, the best approach is to refine the whole area around the highlighted triangle rather than just add a node within the triangle. Note that steep bank areas will require very fine discretization to reduce the elevation error. Copying the bed breaklines into the mesh is normally the most effective means of guaranteeing a faithful representation in important areas. The dz threshold for the yellow element display can be changed using the “**Set Elevation Difference Threshold**” command found under the “**Options**” menu.

To improve the generated contours, insert more nodes into areas of greatest discrepancy. Note also that the information line at the bottom of the map includes an extra number, dz, which is the elevation difference between input and generated topographies. Triangulation and smoothing operations may be performed at any time, as usual. A new generated contour map is displayed each time the mesh is triangulated or smoothed. Inserting individual points at key locations should improve the “fit” significantly. Breaklines have the added characteristic of not moving during a smoothing operation, thus ensuring that important topographic features (such as channel banks) are represented in the computational mesh.

To return to viewing the generated mesh, select “**Mesh**” under the “**Display**” Menu.

Saving the Mesh

Once an acceptable mesh has been developed (or at any other time), it can be saved either as a mesh or as a River2D input file by using the “**Save As Mesh**” or “**Save As River2D Input File**” commands respectively under the “**File**” menu. The “**Save As Mesh**” (same as “**Save**”) command prompts for a file name and then saves a text file with a .msh extension containing all the points, boundary segments and breakline segments, each as a line in the file. The “**Save**” icon causes the current mesh to be saved with the current mesh file name defined by the last “**Save As Mesh**” command.

A saved mesh file can be opened later using the “**Open Mesh File**” command on the **File** menu. The opened mesh replaces any generated mesh present. The bed topography must be compatible with the input mesh in the sense that all points in the input mesh must be located within the defined topography. If this is not the case, then a message will be displayed and the input will be aborted. When input, the mesh will update automatically to the elevation and roughness values in the displayed topography. This is a convenient means of investigating design alternatives (changing elevations) or calibrating (changing roughness values) by making the changes in the .bed file and updating the mesh file, from which a new River2D file can be generated. An update bed feature is also included in the River2D model itself.

The “**Save As River2D Input File**” command saves the mesh in a River2D input file (.cdg extension) that is ready to be run. Default values are supplied for all of the run parameters. These may be changed, if desired, in the River2D model interface or by editing the resulting input file.

Before the input file is saved, a dialog box appears requesting an estimate of the inflow water surface elevation. Since the inflow boundary condition is usually discharge, the inflow elevation is an unknown which the program must solve for. The estimate supplied by the user is used to set up initial conditions for water surface throughout the modeled domain. The starting elevation at any point is calculated as a distance weighted average between the inflow estimate and fixed outflow values. Thus, depending on the inflow estimate (should be higher than outflow), an initial water surface slope is set up. The initial velocity components are assumed to be zero. During the first few time steps of in the River2D model calculations, the water will begin to flow from the inflow toward the outflow, throughout the channel.

Experience indicates that guessing an inflow elevation on the high side tends to work somewhat better than a low estimate. This is likely because drying of edge elements is faster than wetting so that a falling water surface is desirable. For the same reason, when

simulating a range of discharges, it is advisable to proceed from highest to lowest.

The saving operation also performs a Reverse Cuthill-McGee node ordering algorithm to minimize River2D execution time. There may be some delay during saving as this algorithm executes.

After saving, additional modifications, including more points or further smoothing, can be performed and subsequently saved. The previously saved meshes can also be reloaded at any time.

Modifying an Existing Boundary and Mesh

Once the system boundary and mesh are defined, several reasons to change or refine the mesh may be encountered. These include recognizing that an area needs more refinement after running an initial solution at the calibration discharge; finding that a breakline does not track along a feature as closely as desired due to drawing it at too coarse a zoom level; finding there are not enough elements across some portion of the channel; or any other reason the mesh needs further refinement.

This section describes tools that can be used to modify an existing mesh (and boundary) to better describe the study area or improve model performance. This description begins with tools for mesh modification and concludes with tools for modifying an existing boundary.

Deletion of floating and independent fixed nodes has been described earlier. Removal of certain nodes may improve the structure of the mesh and increase the QI without losing topographic resolution. In many cases, however, adding or moving nodes and breaklines will yield better capture of the underlying topography in the finite element mesh.

Sliding nodes on breaklines have several unique characteristics so the operations to alter them are unique. Sliding nodes may be deleted using the “**Delete Floating Node**” command. An alternative method to delete sliding nodes is to remove their

adjoining breakline segments. This converts the nodes to fixed nodes that may then be removed using “**Delete Fixed Node**”.

Breakline segments may be deleted using the “**Delete Breakline Segment**” (icon) command. The icon acts as a toggle so you may click on it to begin deleting segments, click on all segments to be deleted, and then click on the icon again to end the deletion process. When a breakline segment is deleted, any sliding nodes at the end of that segment are converted to fixed nodes and the connecting information between them is deleted. The newly converted fixed nodes can then be moved using the process described above. To delete an entire breakline, a combination of sliding node deletions, breakline deletions and fixed node deletions is required. Using “**Delete Fixed Node**” on a fixed node within a breakline results in all breakline segments and floating nodes between the deleted fixed node and the next fixed nodes along the breakline to be deleted.

Fixed and floating (but not sliding) nodes may be moved by a drag and drop operation. Click on the “**Move Node**” icon, then click on the node to be moved and hold the left mouse button down while dragging the node to the desired new location. The “**Move Node**” command is found under the “**Generate**” menu.

Fixed nodes on breaklines may also be moved provided that all sliding nodes on the adjacent breakline segments are deleted (using “**Delete Floating Node**”) prior to attempting the move. Once the sliding nodes all have the red deletion “X” across them, the fixed node may be moved to a new location and the breaklines will remain connected. Thus, though tedious, an entire breakline can be relocated by deleting the sliding nodes and moving the fixed nodes until the desired alignment is reached. The move operation should be conducted prior to triangulation as triangulation will likely add some sliding nodes to the breakline. Experience suggests that only a few breakline nodes will need to be moved to achieve good feature alignment when zoomed in.

To find all mesh elements with QI below a desired value, use the “**Set Bad Triangle Threshold**” dialog in the “**Options**” menu to

set a target value. Then toggle the “QI values” item in the “**Display Options**” menu (right click) to mark all triangles with QI less than the threshold with red.

Through a combination of these mesh refinement tools, the mesh can be adjusted to closely reproduce topographic contours, have adequate density to simulate complex regions of the channel, and have an adequately high QI.

In studies where a very large range of discharges must be simulated, it may be necessary to enclose significant areas of flood plain within the simulation boundary. For most discharges, this will produce a large area of dry nodes that will require computational effort and thus increase solution time. Once an overall mesh that covers the highest flows to be simulated is created, it may be desirable to be able to reduce the mesh size and speed up simulation of lower flows. In other situations, you may find that you did not encompass a sufficiently high portion of the site topography within the boundary along one side or other of the stream. Where possible, redefining part or the entire boundary would allow you to retain all of the work spent defining the mesh within the channel. The following tools are designed for such a task.

The “**Delete Fixed Node**” and “**Delete Floating Node**” commands in the “**Generate**” menu act on the boundary as well as within the mesh. When a fixed boundary node is deleted, the two adjacent sliding nodes are changed to fixed nodes and the boundary is connected between them.

Fixed boundary nodes may be moved as well. As with breaklines, the adjacent sliding nodes must be deleted first and then clicking the “**Move Node**” icon allows the fixed boundary node to be repositioned. If the boundary moves such that mesh nodes or breaklines fall outside the revised boundary, then those mesh components are deleted. Thus, it will usually be better to start with a large boundary and move in, though moving a smaller boundary to encompass more area is allowed as long as it falls within the defined topography.

Note that internal boundaries cannot be entirely removed by deleting nodes. The last two internal boundary nodes will remain so caution is advised when defining internal boundaries.

In some situations it is desired to place additional fixed nodes along the boundary. A segment bifurcation tool is provided that allows any boundary segment to be divided into two equal sections with a fixed node. The “**Bisect Boundary Segment**” command found under the “**Generate**” menu performs this function. Once placed on the boundary, the new fixed node can be moved when adjacent floating nodes are removed, as described above.

Mesh Generation Strategy

Previous sections of this document describe the basic order of mesh generation. Experience suggests that the use of breaklines is vital to producing the best resolution with the smallest number of nodes. Thus the following order is suggested.

1. Define the external boundary at the largest extent needed for the highest flow to be simulated for the study.
2. Set inflow and outflow boundary conditions for the calibration discharge.
3. Define mesh breaklines along all major topographic breaks. Include top of bank, toe of bank, thalweg, etc. for all channels.
4. Add additional breaklines such that all channels that convey significant volumes of flow or are important to various study objectives are divided into at least 8 – 10 elements across the channel.
5. Triangulate and smooth. More than one smooth operation may be desired.
6. Check for large elevation difference triangles in the channel. If needed, add additional breaklines to capture the underlying topography.

7. Add floating nodes in floodplain areas at a relatively large spacing compared to the main channel to conserve nodes.
8. Triangulate.
9. Inspect the inflow and outflow boundaries. If they have not been subdivided such that 10 to 20 nodes lie across the inflow and outflow boundaries, add floating nodes near the boundary or use the boundary segment bisection tool, then triangulate and smooth until the desired division is achieved.
10. Finally generate the River2D input file, run the model and determine if further mesh refinement is needed.

Appendix 1

R2D_Mesh Menu Tree

File

-
- New
- Open Bed File
- Open Ice File
- Open Mesh File
- Open CDG File
- Save Mesh
- Save As Mesh
- Save As R2D_Flow Input File
-
- Print
- Print Preview
- Print Setup
-
- Exit

View

-
- Toolbar
- Status Bar

Boundary

-
- Define External Boundary
- Define Internal Boundary
- Set Inflow
- Set Inflow by Area
- Set Outflow
- Set Outflow by Area
- Set Noflow
- Set Noflow by Area
-
- Clear Boundary

Generate

-
- Boundary Nodes
- Bisect Boundary Segment
-

- Add Floating Node
- Add Fixed Node
- Move Node
- Uniform Fill
- Area Fill
- Region Fill
- Radial Fill
- Breakline
- Region Refine
- Region Refine (Yellow Triangles)
-
- Triangulate
- Smooth
-
- Delete Floating Node
- Delete Fixed Node
- Delete Breakline Segment
-
- Delete Floating Nodes in Region
-
- Clear Untriangulated Nodes
- Clear Mesh

Display

-
- Redraw
-
- Scale To Fit
- Zoom On Rectangle
- Zoom On Point
- Zoom Out
- Center On Point
-
- Display Options <Rt-Click>
- Nodes
- Breaklines
- Mesh
- Generated Contours
- Find Worst Triangle
- Large Elevation Difference Triangles

Options

-
- Snap to Mesh Breakline Node
- Snap to Bed Breakline Node

Set Contour Interval
Set Topo Smoothing Parameter
Set Elevation Difference Threshold

Help

-

Help Topics
About R2DMesh