

## 14 ADVANCED FEATURES FOR UNSTEADY FLOW ROUTING

HEC-RAS has several advanced features that can be used when modeling complex unsteady flow situations. These features include mixed flow regime capabilities (subcritical, supercritical, hydraulic jumps, and draw downs); the ability to perform a dam break analysis; levee overtopping and breaching; hinge pool calculations for navigation dams; how to model pressurized pipe flow in HEC-RAS; and using generic rules to control gate operations at hydraulic structures.

The advanced features discussed in this section are listed below.

- Mixed Flow Regime
- Dam Break Analysis
- Levee Breaching and Overtopping
- Pump Stations
- Navigation Dams
- Modeling Pressurized Pipe Flow
- User Defined Rules for Controlling Gate Operations
- Automated Calibration of Manning's n Values for Unsteady Flow

### Mixed Flow Regime

Modeling mixed flow regime (subcritical, supercritical, hydraulic jumps, and draw downs) is quite complex with an unsteady flow model. In general, most unsteady flow solution algorithms become unstable when the flow passes through critical depth. The solution of the unsteady flow equations is accomplished by calculating derivatives (changes in depth and velocity with respect to time and space) in order to solve the equations numerically. When the flow passes through critical depth, the derivatives become very large and begin to cause oscillations in the solution. These oscillations tend to grow larger until the solution goes completely unstable.

In order to solve the stability problem for a mixed flow regime system, Dr. Danny Fread (Fread, 1986) developed a methodology called the "Local Partial Inertia Technique." The LPI method has been adapted to HEC-RAS as an option for solving mixed flow regime problems when using the unsteady flow analysis portion of HEC-RAS. This methodology applies a reduction factor to the two inertia terms in the momentum equation as the Froude number goes towards a user entered Froude Number Threshold (Default = 0.8). The modified momentum equation is shown below:

$$\sigma \left[ \frac{\delta Q}{\delta t} + \frac{\delta \left( \frac{BQ^2}{A} \right)}{\delta x} \right] + gA \left( \frac{\delta h}{\delta x} + S_f \right)$$

and

$$\sigma = 1.0 - \left( \frac{F_r}{F_T} \right)^m \quad (F_r \leq F_T; \ m \geq 1)$$

$$\sigma = 0 \quad (F_r > F_T)$$

where:

$\sigma$  = LPI factor to multiply by inertial terms.

$F_T$  = Froude number threshold at which factor is set to zero. This value has a practical application range from 0.0 to 2.0 (default is 0.8). If you use zero, the inertial terms are always zeroed out, and in effect you have a Diffusion Wave routing scheme, rather than Full Unsteady Flow equations.

$F_r$  = Froude number.

$m$  = Exponent of equation, which changes the shape of the curve. This exponent can range between 1 and 128 (default value is 4). A practical upper limit would be 32.

$h$  = Water surface elevation

$S_f$  = Friction slope

$Q$  = Flow rate (discharge)

$A$  = Active cross sectional area

$g$  = Gravitational force

The default values for the equation are  $FT = 0.8$  and  $m = 4$ . When the Froude number is greater than the threshold value, the factor is set to zero. The user can change both the Froude number threshold and the exponent. As you increase the value of both the threshold and the exponent, you decrease stability but increase accuracy. As you decrease the value of the threshold and/or the exponent, you increase stability but decrease accuracy. To change either the threshold or the exponent, select **1D Mixed Flow Options** tab from the **Options/Computational Options and Tolerances** menu of the Unsteady Flow Analysis window. When this option is selected, the unsteady mixed flow options window will appear as shown in Figure 14-1.

As shown in Figure 14-1, the graphic displays what the magnitude of the LPI factor will be for a given Froude number and a given exponent  $m$ . Each curve on the graph below represents an equation with a threshold of 1.0 ( $FT$ ) and a different exponent ( $m$ ).

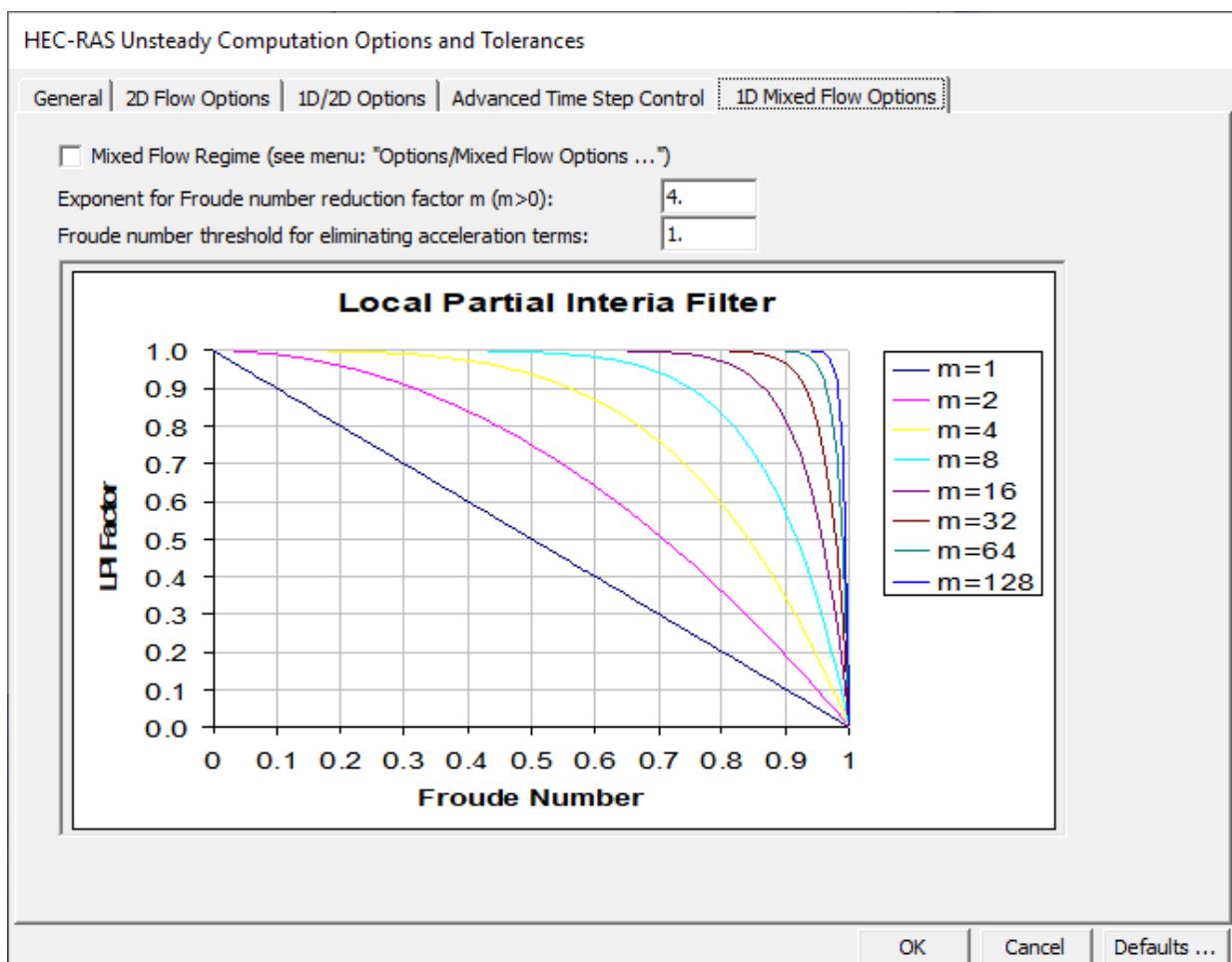


Figure 14 1. Unsteady Mixed Flow Options Window

By default, the mixed flow regime option is not turned on. To turn this option on, check the **Mixed Flow Regime** box, which is contained at the top of the Mixed Flow regime window. This window and option is shown in the Figure 14-2.

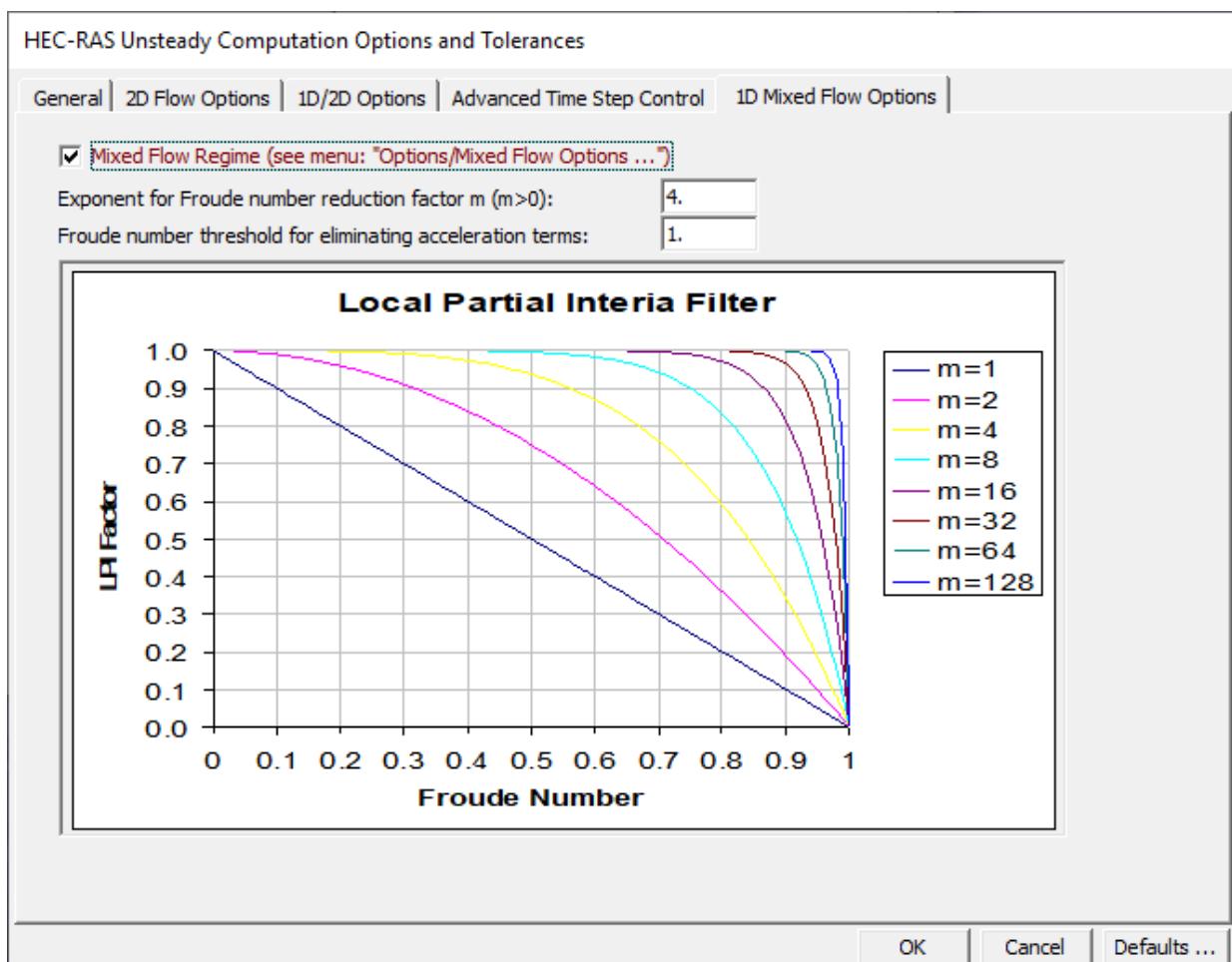


Figure 14 2. Unsteady Flow Analysis Window with Mixed Flow Regime Option Turned On

In general, when modeling a river system that is completely subcritical flow, you should not turn on the mixed flow regime option. If the system is mostly subcritical flow, with only a few areas that pass through critical depth, then this option can be very useful for solving stability problems. However, there may be other options for modeling the areas that pass through critical depth. For example, if the system has a location with a drop in the bed where flow passes through critical depth over the drop, but is subcritical just downstream of the drop, this would be a good location to model the drop as an inline weir within HEC-RAS. By modeling the drop as an inline weir, the program is not modeling the passing through critical depth with the momentum equation, it is getting an upstream head water elevation for a given flow from the weir equation. If the river system has several areas that pass through critical depth, go supercritical, and go through hydraulic jumps, then the mixed flow methodology may be the only way to get the model to solve the unsteady flow problem.

A profile plot of a mixed flow regime problem is shown in Figure 14-3. This example was run with the unsteady flow simulation capability within HEC-RAS using the mixed flow regime option. The example shows a steep reach flowing supercritical, which then transitions into a mild reach. A hydraulic jump occurs on the mild reach. The mild reach then transitions back to a steep reach, such that the flow goes from subcritical to supercritical. Because of a high downstream boundary condition (for example backwater from a lake), the flow then goes from supercritical to subcritical through another hydraulic jump.

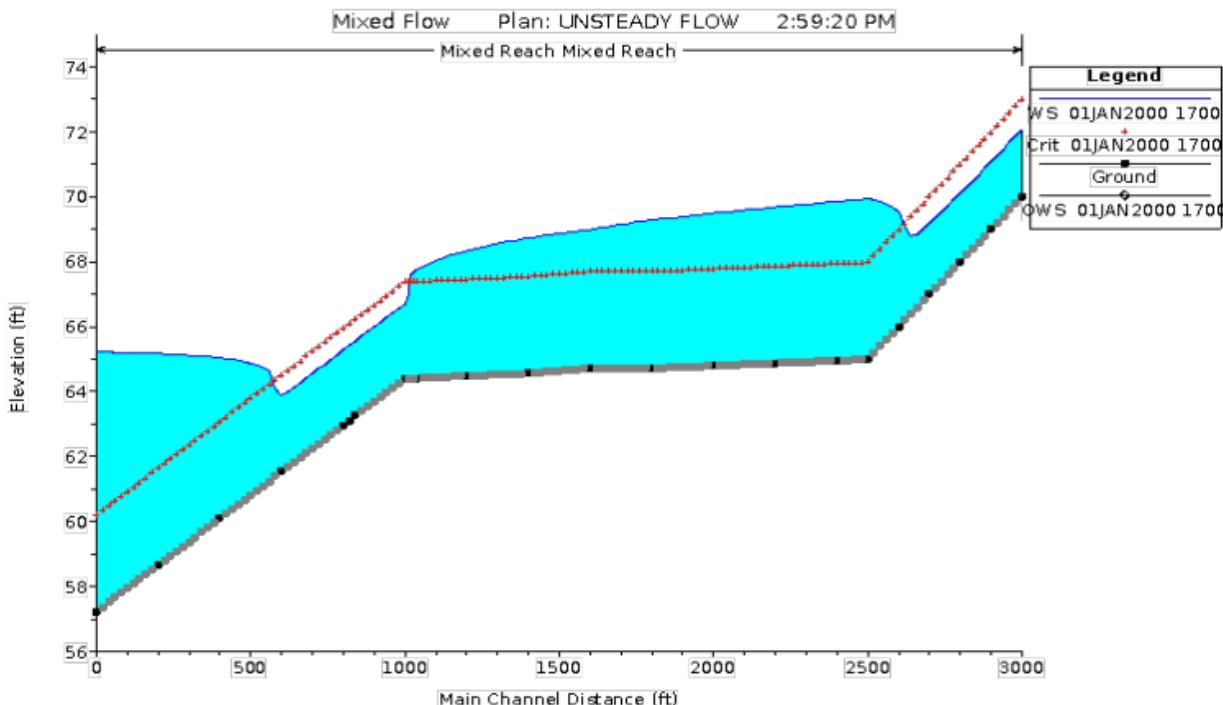


Figure 14.3. Example Mixed Flow Regime Run with Unsteady Flow Routing

## Dam Break Analysis

The failure of several dams in this country (Buffalo Creek, West Virginia 1972; Teton dam, Idaho 1976; Laural Run Dam and Sandy Run Dam, Pennsylvania 1977; Kelly Barnes Dam, Georgia 1977; and others), has led our nation to take a strong look at dam safety. One aspect of dam safety is to answer the question, "What will happen if the dam were to fail?" The ability to evaluate the results of a dam failure has been added into the HEC-RAS software.

HEC-RAS can be used to model both overtopping as well as piping failure breaches for earthen dams. Additionally, the more instantaneous type of failures of concrete dams can also be modeled. The resulting flood wave is routed downstream using the unsteady flow equations. Inundation mapping of the resulting flood can be done with the RAS-Mapper portion of the software when GIS data (terrain data) are available.

Dams are modeled within HEC-RAS by using the Inline Structure editor or the SA/2D Area Connection (for 2D modeling). Each of these editors allows the user to put in an embankment, define overflow spillways and weirs, gated openings (radial and sluice gates), culverts, rating curves, and time series of flow releases. Gated openings can be controlled with a time series of gate openings or using the elevation control gate operation feature in HEC-RAS. For more information on modeling inline hydraulic structures within HEC-RAS, please review Chapter 5 of this manual (Entering and Editing Geometric Data).

The lake area upstream of the dam can either be modeled with cross sections, a storage area (Figure 14-4.), or a 2D Flow Area. If cross sections are used, then HEC-RAS will perform full unsteady flow routing through the reservoir pool and downstream of the dam. If a storage area is used, HEC-RAS uses level pool routing through the lake, then unsteady flow routing downstream of the dam. If a 2D Flow Area is used then full 2D modeling can be used within the reservoir pool and downstream of the

dam. When using a storage area to represent the reservoir pool, HEC-RAS requires two cross sections inside of the reservoir pool, then the inline structure representing the dam, and then the downstream cross sections (see Figure 14-4). The routing reach is hydraulically connected to the reservoir (storage area) with the first (most upstream) cross section. This cross sections water surface is forced to the elevation of the water surface in the storage area during the unsteady flow routing. The second cross section in the pool area is required as a bounding cross section for the inline structure (the dam).

One additional caution for using a storage area to represent the pool area: When the initial conditions are computed by backwater analysis, it is up to the user to ensure that the water surface computed just upstream of the dam (for the two cross sections) is consistent with the starting water surface entered for the storage area. If this is not the case, the model will most likely have stability problems at the very start of the unsteady flow routing. There are two ways to ensure these water surfaces are consistent. The first is to adjust the low flow gate openings and initial base flow in the reach to produce a water surface that is consistent with the desired starting pool elevation. The second way is to use the option that allows the user to force the water surface at a cross section during the initial conditions calculations. This option is called **Internal RS Initial Stages**, and is available from the **Options** menu of the Unsteady Flow Data editor. This option can be used to set the water surface just upstream of the dam to the same elevation as the storage area.

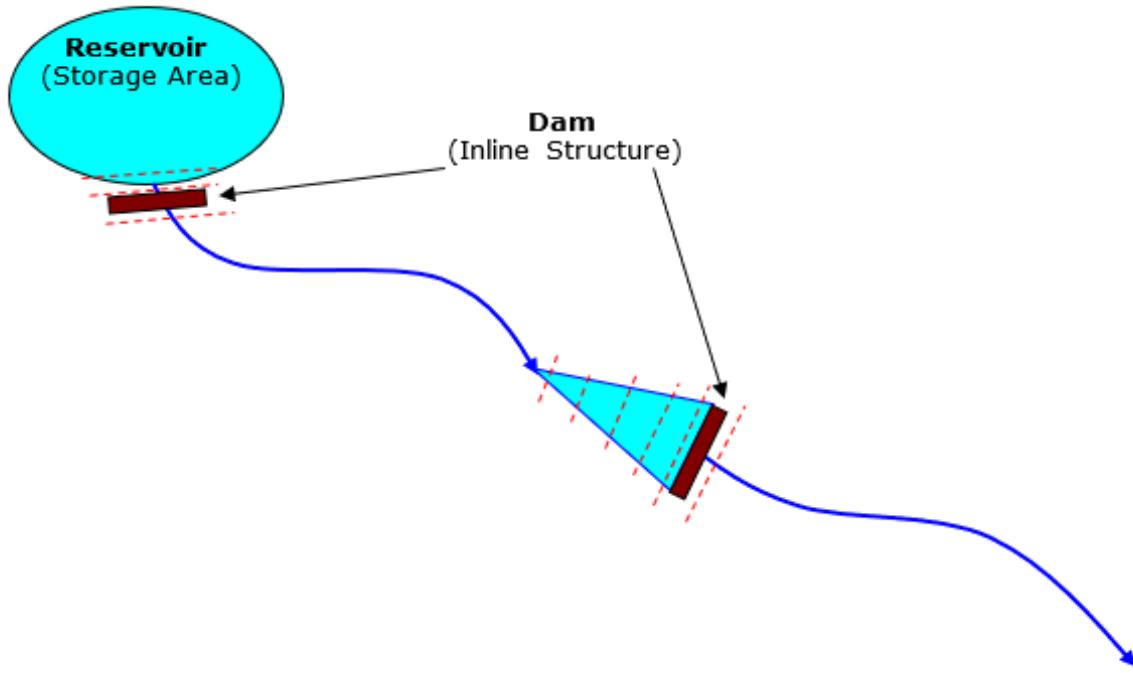


Figure 14.4. Alternate Methods for Modeling a Dam and Reservoir Pool in HEC-RAS.

An example of using the Inline Structure feature to model a dam is shown in Figure 14-5. As shown in the Figure, the user enters the embankment and overflow spillway as one piece using the Weir/ Embankment editor. The embankment is shown as the gray filled in area above the ground. The overflow spillway is the rectangular notch on the upper left hand side of the embankment. The main outlet works consist of two rectangular gates, which are entered through the gate editor. The gates are shown towards the bottom of the embankment in this example.

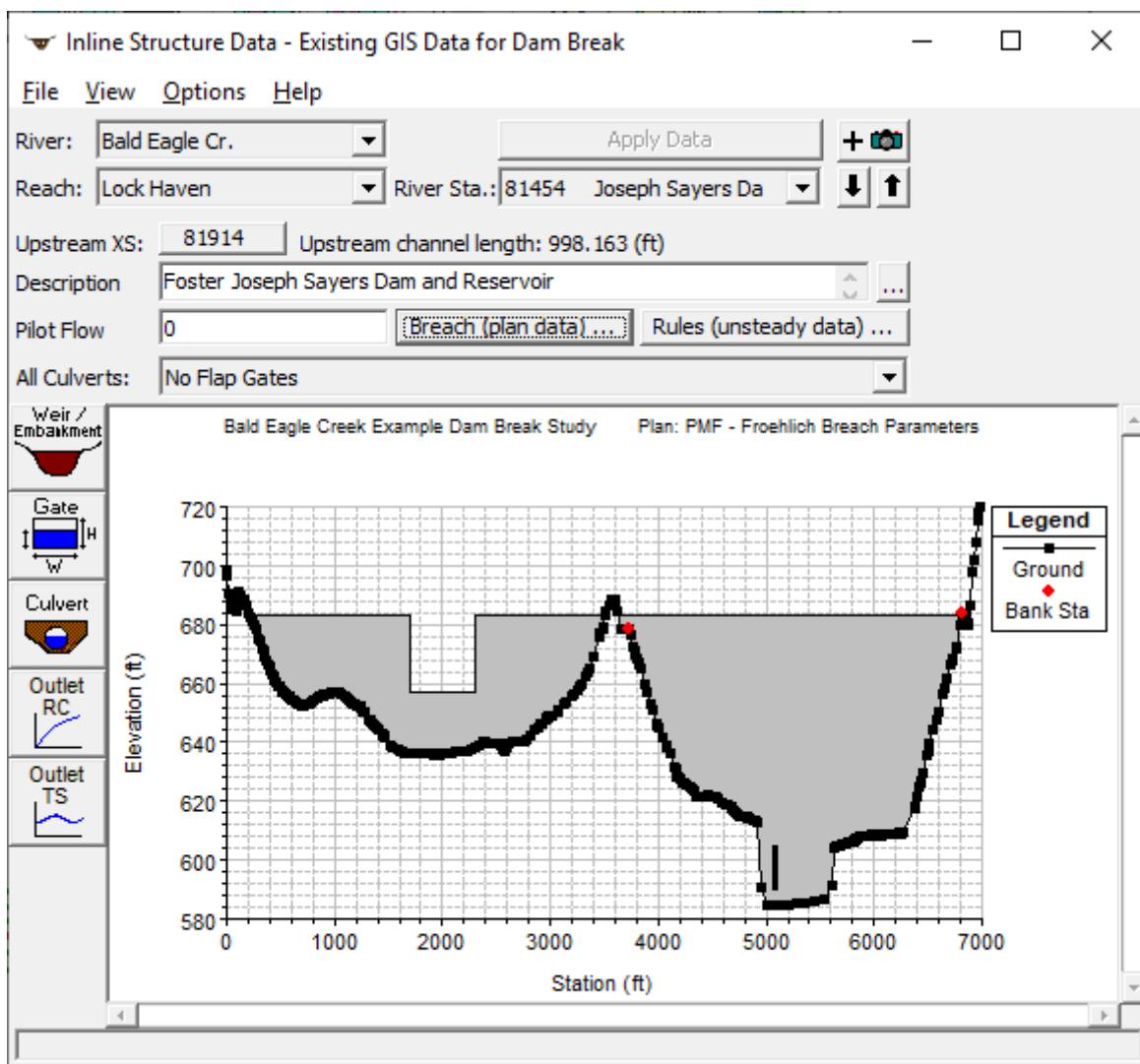


Figure 14.5. Inline Structure Editor with Example Dam Shown

## Entering Dam Break Data

Entering dam breach information is accomplished by pressing the button labeled **Breach (plan Data)**. The breach information is stored as part of the current Plan. This was done to facilitate evaluating dam and levee breaching in a real time river forecasting mode. By putting the breach information in the Plan file, the geometric pre-processor does not have to be run again, thus saving computation time during forecasting. The user can also access dam breach information by selecting **Dam Breach (Inline Structure)** from the Options menu of the Unsteady Flow Analysis window. Once the Breach button is pressed, the Dam Breach window will appear as shown in Figure 14-6.

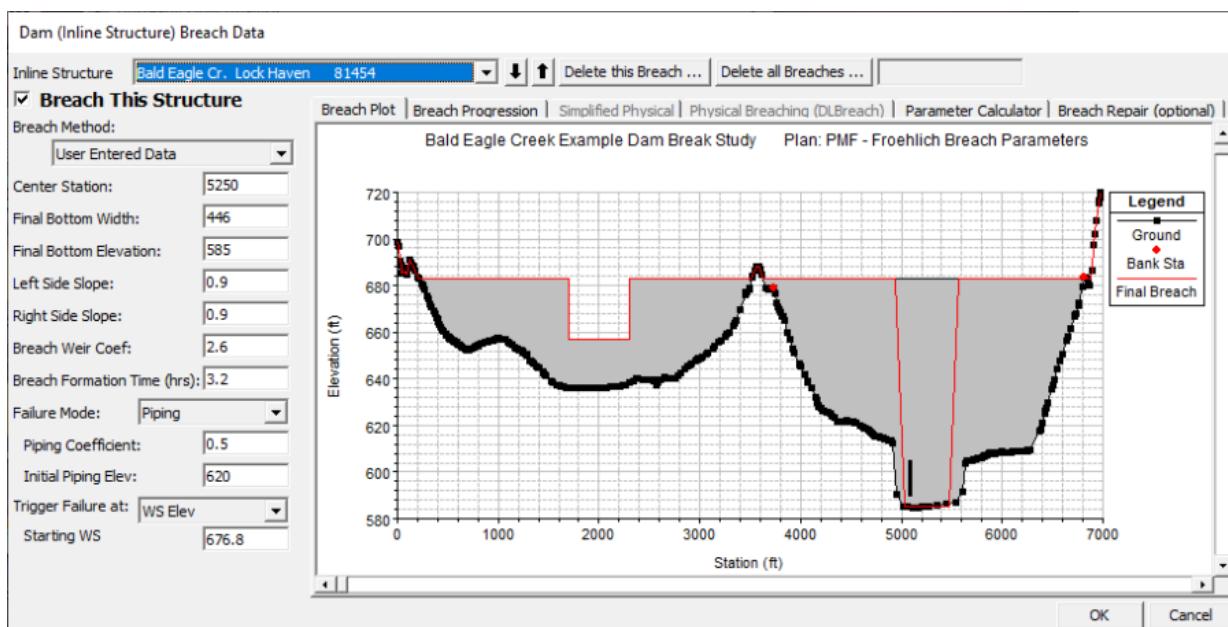


Figure 14.6. Dam Breach Data Editor with Example Dam

As shown in Figure 14-6, the user selects a particular Inline Structure to perform the breach on. At the top left of the editor is an option labeled: **Breach This Structure** - This check box is used to decide if the program will perform the breach or not. In order for the breach to occur this box must be checked. This box was added to allow the user to turn certain breaches on or off, without losing the user entered breach information.

**Inline Structure.** This field is used to select the particular inline structure that you want to perform a breach analysis on. The user can enter breach data and perform a breach for more than one dam within the same model.

**Delete This Breach.** This button is used to clear all of the dam breach information for the currently opened inline structure.

**Delete All Breaches.** This button is used to delete the dam breach information for all of the inline structures in the model.

**Breach This Structure.** This check box is used to turn the breaching option on and off without getting rid of the breach data. This box must be checked in order for the software to perform the dam breach. When this box is not checked, no breaching will be performed on this structure. Next the decision needs to be made as to which "**Breaching Method**" to use. Currently the user has three Breaching Methodologies to choose from, either **User Entered Data**, **Simplified Physical**, or **Physical Breaching (DLBreach)**. The **User Entered Data** method requires the user to enter all of the breach information (i.e. breach size, breach development time, preach progression, etc....). The **Simplified Physical** breaching method allows the user to enter velocity versus breach down-cutting and breach widening relationships, which are then used dynamically to figure out the breach progression versus the actual velocity being computed through the breach, on a time step by time step basis. The Physical Breaching (DLBreach) requires material properties for the dam, the core of the dam, and the outer face of the downstream sideslope.

## User Enter Breach Data

If the "**User Entered Data**" Breaching Method is selected, then the following data must be entered for the breaching analysis:

**Center Station.** This field is used to enter the cross section stationing of the centerline of the breach. The stationing is based on the inline structure that is shown in the graphic.

**Final Bottom Width.** This field is used to enter the bottom width of the breach when it has reached its maximum size.

**Final Bottom Elevation.** This field is used to enter the bottom elevation of the breach when it has reached its maximum size.

**Left Side Slope.** This field is used to enter the left side slope for the trapezoid that will represent the final breach shape. If a zero is entered for both side slopes, the breach will be rectangular. Side slopes are entered in values representing the horizontal to vertical ratio. For example, a value of 2 represents 2 feet horizontally for every 1 foot vertically.

**Right Side Slope.** This field is used to enter the right side slope for the trapezoid that will represent the final breach shape. If a zero is entered for both side slopes, the breach will be rectangular. Side slopes are entered in values representing the horizontal to vertical ratio. For example, a value of 2 represents 2 feet horizontally for every 1 foot vertically.

**Breach Weir Coef.** This field is used to enter a weir coefficient that will be used to compute flow through the breach, when the breach is open to the atmosphere. A default value of 2.6 is set automatically, but user's may want to adjust this depending upon the type of dam, and breaching process.

**Breach Formation Time (hrs).** This field is used to enter the time required for the breach to form. It represents the time from the initiation of the breach, until the breach has reached its full size. The modeler should be very careful in selecting this time. If a linear breach progression rate is selected, then the breach time should be limited to when the breach begins to significantly erode and up to when the major portion of the breach is formed. More information on the breach formation time is provided later in this chapter.

**Failure Mode.** This selection box contains two options for the failure mode of the breach, a **Piping** failure or an **Overtopping** failure. The overtopping failure mode should be selected when the water surface overtops the entire dam and erodes its way back through the embankment, or when flow going over the emergency spillway causes erosion that also works its way back through the embankment. The Piping failure mode should be selected when the dam fails due to seepage through the dam, which causes erosion, which in turn causes more flow to go through the dam, which causes even more erosion. A piping failure will grow slowly at first, but tends to pick up speed as the area of the opening begins to enlarge. At some point during the breach, the embankment above the breach will begin to sluff, at which time a large mass wasting of the embankment will occur.

**Piping Coefficient.** This field is only used if the Piping failure mode has been selected. The user enters an orifice coefficient into this field. The orifice equation is used to calculate the flow through the breach opening while it is acting in a piping flow manner. Once the embankment above the opening sloughs, and the water is open to the atmosphere, the program transitions to a weir equation for computing the breach flow.

**Initial Piping Elev.** This field is used to enter the elevation of the center of the piping failure when it first begins to occur.

**Trigger Failure At.** This field is used to enter the mode in which the breach initiation will be triggered. There are three options available within HEC-RAS for initiating the start of the breach: a water surface elevation (**WS Elev**), a specific instance in time (**Set Time**), and a combination of exceeding a water surface elevation for a user specified duration (WS Elev + Duration). With the third option (WS Elev + Duration) the user enters a threshold water surface elevation to start monitoring the location. A duration is also entered. If the water surface remains above the threshold value for the user entered duration, then the breach is initiated. Additionally the user can enter a water surface elevation labeled "Immediate Initiation WS." If the water surface elevation gets up to or beyond this elevation, the breach is immediately initiated.

**Starting WS.** This field is only used if the user has selected a trigger failure mode of water surface elevation (**WS Elev**). The user enters a water surface elevation into this field. The water surface represents the elevation at which the breach will begin to occur.

**WS Elev+Duration .** If the user selects **WS Elev+Duration** for the trigger mechanism of the failure, then they have three additional fields of data to enter. The first variable is **Threshold WS**. This variable is the water surface elevation at which the program starts to monitor the flow for duration above this water surface. The second variable is **Duration above Threshold**. This variable is used to specify the duration that the water surface must be above the threshold water surface elevation before the failure will initiate. The final variable, **Immediate initiation WS**, is used to instruct the program to begin the breach if the water surface in the structure reaches this elevation or higher, regardless of the duration requirement. This last field is optional.

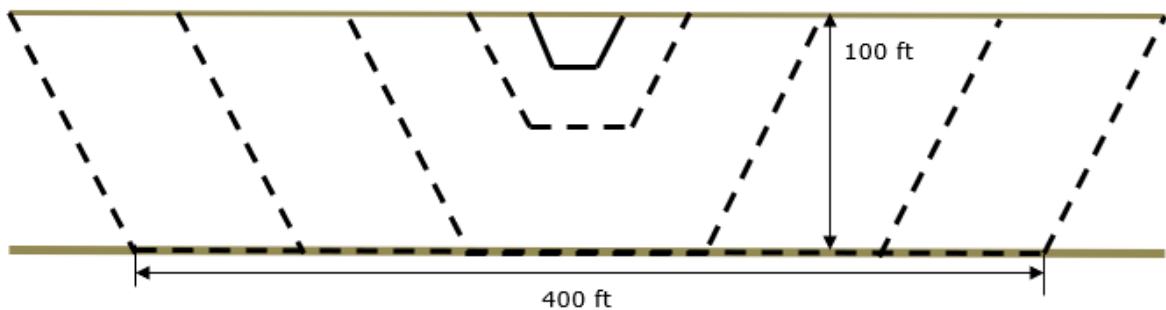
**Set Time.** If the user selects the **Set Time** option, then a starting date and time to initiate the breach must be entered.

#### **Breach Plot.**

When this tab is selected, a plot of the inline structure will show up in the graphic window. The plot will show the proposed breach maximum size and location in a red color.

#### **Breach Progression.**

When this tab is selected a table will appear in the graphic display window. The table is used to enter a user defined progression curve for the formation of the breach. This is an optional feature. If no curve is entered, the program automatically uses a linear breach progression rate. This means that the dimensions of the breach will grow in a linear manner during the time entered as the full breach formation time. Optionally, the user can enter a curve to represent the breach formation as it will occur during the breach development time. The curve is entered as Time Fraction vs. Breach Fraction. The Time Fraction is the decimal percentage of the full breach formation time. The breach fraction is the decimal percentage of the breach size. Both factors are entered as numbers between zero and one. An example of a user entered nonlinear breach progression rate is shown below.



**User Specified Vertical/Horizontal Growth Rate.** When the Breach Progression Tab has been selected, the user can select either the User Specified Vertical/Horizontal Growth Rate (default) or the Proportional Vertical/Horizontal Growth Rate. If the User Specified Vertical/Horizontal Growth Rate option is selected, then the breach will grow in the horizontal direction at a rate that will allow it to reach its full length during the user specified breach time (Figure 14-7.). However, the vertical growth rate, will be based on the user entered Vertical to Horizontal ratio. For the typical case where the breach is wider than it is deep, the breach will reach the final bottom elevation before it has reached the final width. See discussion on Breach Growth Shape, below.

**Use Proportional Vertical/Horizontal Growth Rate.** When the Breach Progression Tab has been selected, the user can select to use the User Specified Vertical/Horizontal Growth Rate (default) or Proportional Vertical/Horizontal Growth Rate. If Proportional... is selected, then the breach will grow in the horizontal direction based on the formation time. A growth rate will be computed based on the breach width and breach development time, which is then applied to the vertical breach growth rate. See discussion on Breach Growth Shape, below.

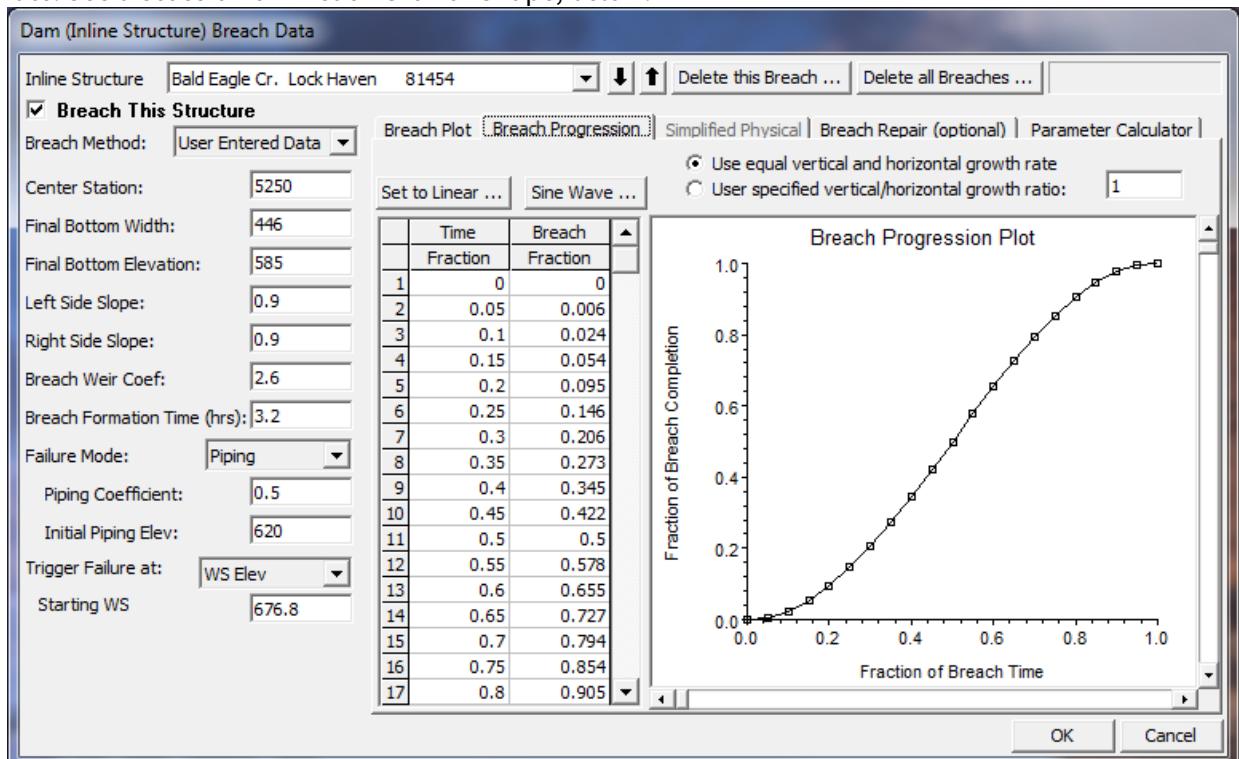


Figure 14 7. Dam Breach Editor with Nonlinear Breach Progression

**NOTE:** Previous to version 4.2, the horizontal and vertical growth rate of the breach was base on reaching the maximum breach depth and width at the user entered "Breach Formation Time". This means if a user put in breach dimensions of 400 ft wide and 100 ft deep, over a period of 2 hours, the horizontal growth rate was 200ft/hr and the vertical growth rate was 50 ft/hr. While this was generally ok for Dam breaches, it was not ok for levee breaching, in that levee breaches are much wider than they are tall. As of version 4.2, RAS computes the breach growth rate based on the breach "Final Bottom Width" and the user entered "Breach Formation Time". Then this same breach growth rate is used for the vertical down cutting of the breach. So in the previous example of a 400 ft breach bottom width and a 2 hour breach development time, the growth rate is 200 ft/hr, which is used for both the down cutting and widening rates. User's can change the vertical Breach Growth Rate by entering a value other than 1.0 under the option labeled "User Specified Vertical/Horizontal Growth Ratio" and the Breach Progression Tab. If a user enters a value of 0.5, that means you want the vertical growth rate to be half of what the Horizontal growth rate gets computed to be.

**WARNING:** The breach growth rate change described in the previous paragraph will generally results in RAS version 4.2 and newer yielding a higher peak flow through the breach, than versions 4.1 and older. If the user wants the same results as version 4.1 and older, you must compute a vertical/horizontal growth rate that will results in the breach reaching its maximum width and depth at the end of the breach formation time. For example (assuming an overtopping breach), if you specified a 400 ft breach bottom width and a 2 hour breach formation time, this is a horizontal growth rate of 200 ft/ hour. However, if you Dam is only 100 ft high, then to reproduce the version 4.1 or older results, the user would need to enter a "User Specified Vertical/Horizontal Growth Ratio" of 0.25. This would cause the program to grow the breach vertically down to the 100 ft depth in exactly 2 hours. Piping breaches are more complicated, in that they have an initial elevation for the hole, and the vertical growth is both up and down.

**Breach Repair (Optional).** This option allows the user to have the breach fill back in during a simulation. This would most often be used for levee breaches, but could also be used for a dam breach if the user were running a long term simulation or if it was assumed that some effort would be put in place to fill a breach back in during a failure. When the Breach Repair tab is selected, the editor will appear as shown in Figure 14-8.

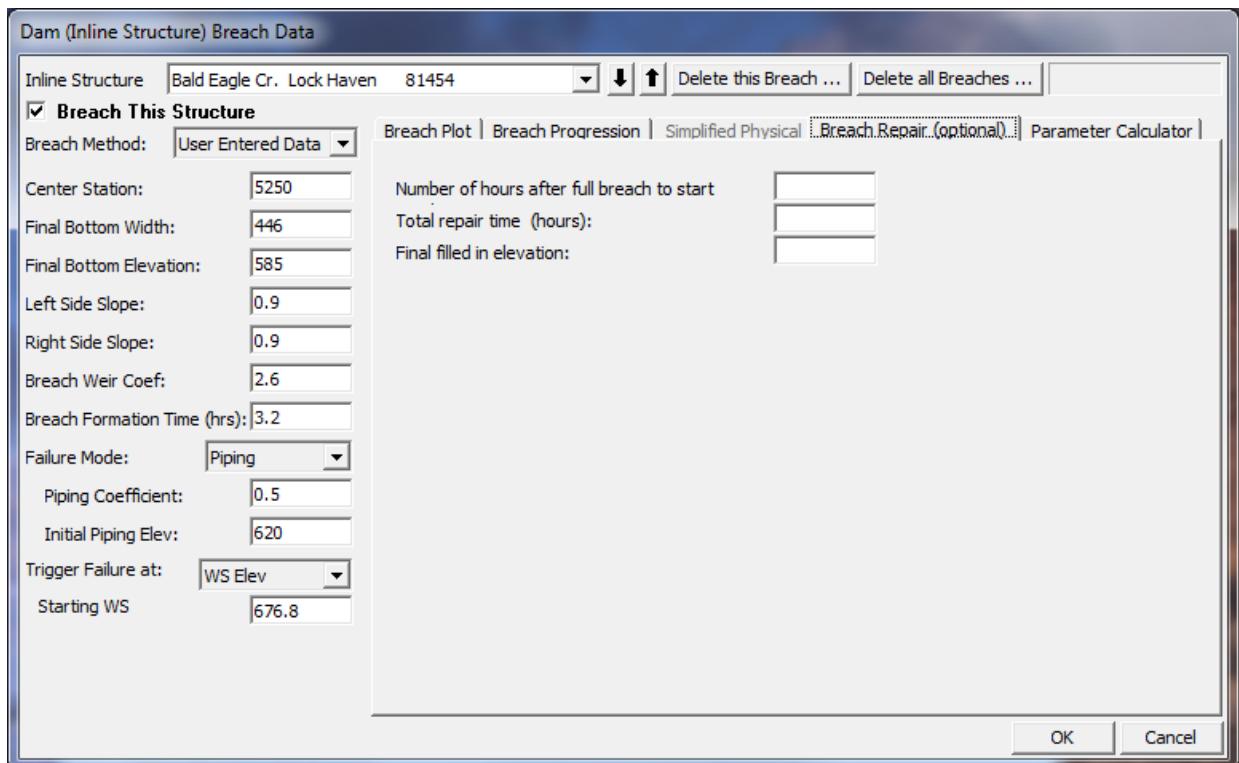


Figure 14.8. Dam Break Editor with Breach Repair Tab Active.

The Breach Repair Option requires the user to enter three pieces of information:

**Number of hours after full breach to start repair:** This field is used to enter the amount of time (in hours) it takes to start the repair process after the breach has occurred.

**Total repair time (hours):** This field is used to enter the total amount of time that it will take to perform the breach repair, in hours.

**Final filled in elevation:** This field is used to enter the top elevation of the final repaired breach.

#### Simplified Physical Breaching

If the User selects to use the "**Simplified Physical**" Breach Method from the drop down at the top of the editor, the Breaching Editor will change to look the following way:

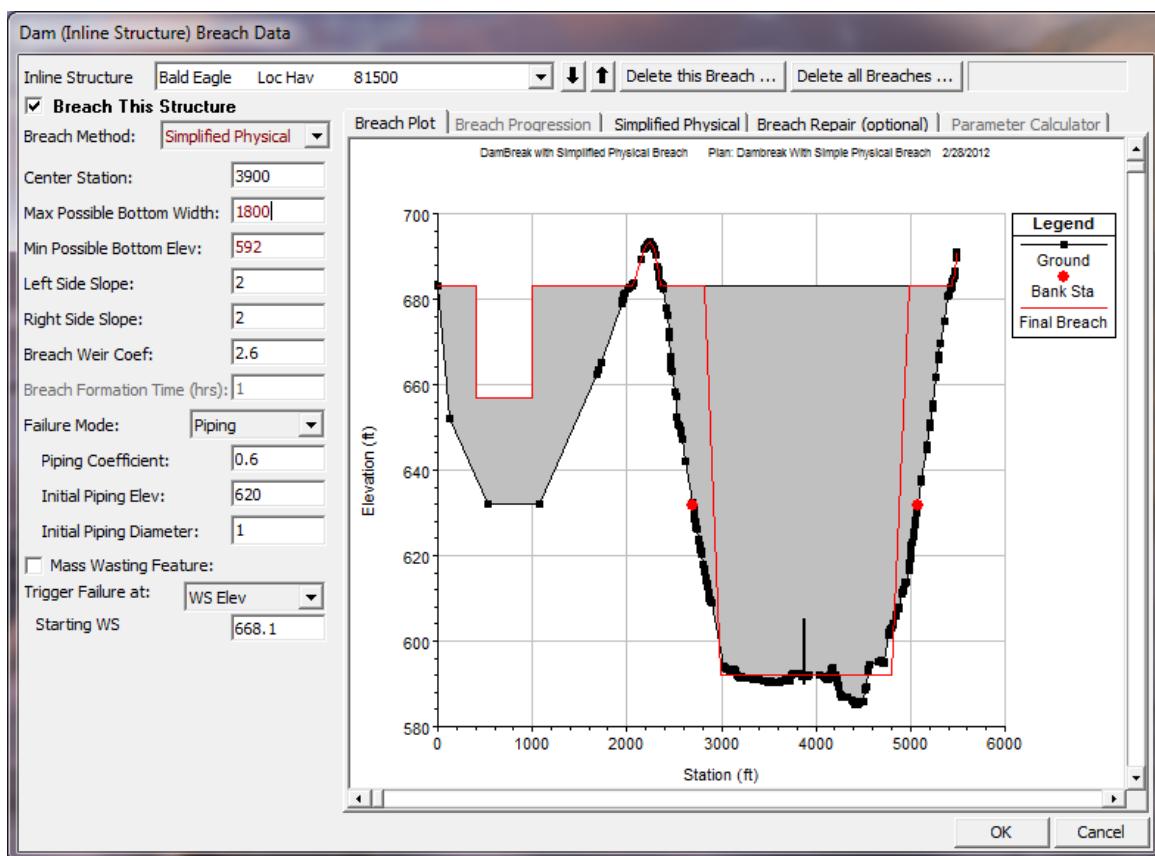


Figure 14.9. Simplified Physical Breaching Option for HEC-RAS

Once the User selects "Simplified Physical" breaching option, there are several fields in which labels change, some additional information required, and some previous information that is not required. The main changes between this method and the "User Entered Data" breach method, are the following:

**Max Possible Bottom Width** – This field is now used to enter a maximum possible breach bottom width. This does not mean this will be the final breach bottom width, it is really being used to limit the breach bottom width growth to this amount. The actual bottom width will be dependent on the velocity verses erosion rate data entered, and the hydraulics of flow through the breach. This field is used to prevent breaches from growing larger than this user set upper limit during the run.

**Min Possible Bottom Elev** – This field is used to put a limit on how far down the breach can erode during the breaching process. This is not necessarily the final breach bottom elevation, it is a user entered limiter (i.e. the breach cannot go below this elevation). The final breach elevation will be dependent on the velocity verses erosion rate data entered, and the hydraulics of flow through the breach.

**Starting Notch Width or Initial Piping Diameter** – If the Overtopping failure mode is selected, the user will be asked to enter a starting notch width. The purpose of this is that the software will use this width at the top of the dam to compute a velocity, from the velocity it will get a down cutting erosion rate (based on user entered data), which will be used to start the erosion process. If a Piping Failure model is selected, the user must enter an initial piping diameter. Once the breach is triggered to

start, this initial hole will show up immediately. A velocity will be computed through it, then the down cutting and widening process will begin based in user entered erosion rate data.

**Mass Wasting Feature** – This option allows the user to put a hole in the Dam or the Levee at the beginning of the breach, in a very short amount of time. This option would probably most often be used in a levee evaluation, in which a section of the levee may give way (Mass Wasting), then that initial hole would continue to erode and widen based on the erosion process. The required data for this option is a width for the mass wasting hole; duration in hours that this mass wasting occurs over (this would normally be a short amount of time); and the final bottom elevation of the initial mass wasting hole (it is assumed that the hole is open all the way to the top of the levee or Dam if this option is used).

**Velocity vs. Downcutting and Widening Erosion Rates.** When using the "Simplified Physical" breaching option, the user is required to enter velocity versus Downcutting erosion rates, as well as velocity versus erosion widening rates. To enter this data the user selects the "Simplified Physical" breach Tab. When this Tab is selected the editor will look like the following:

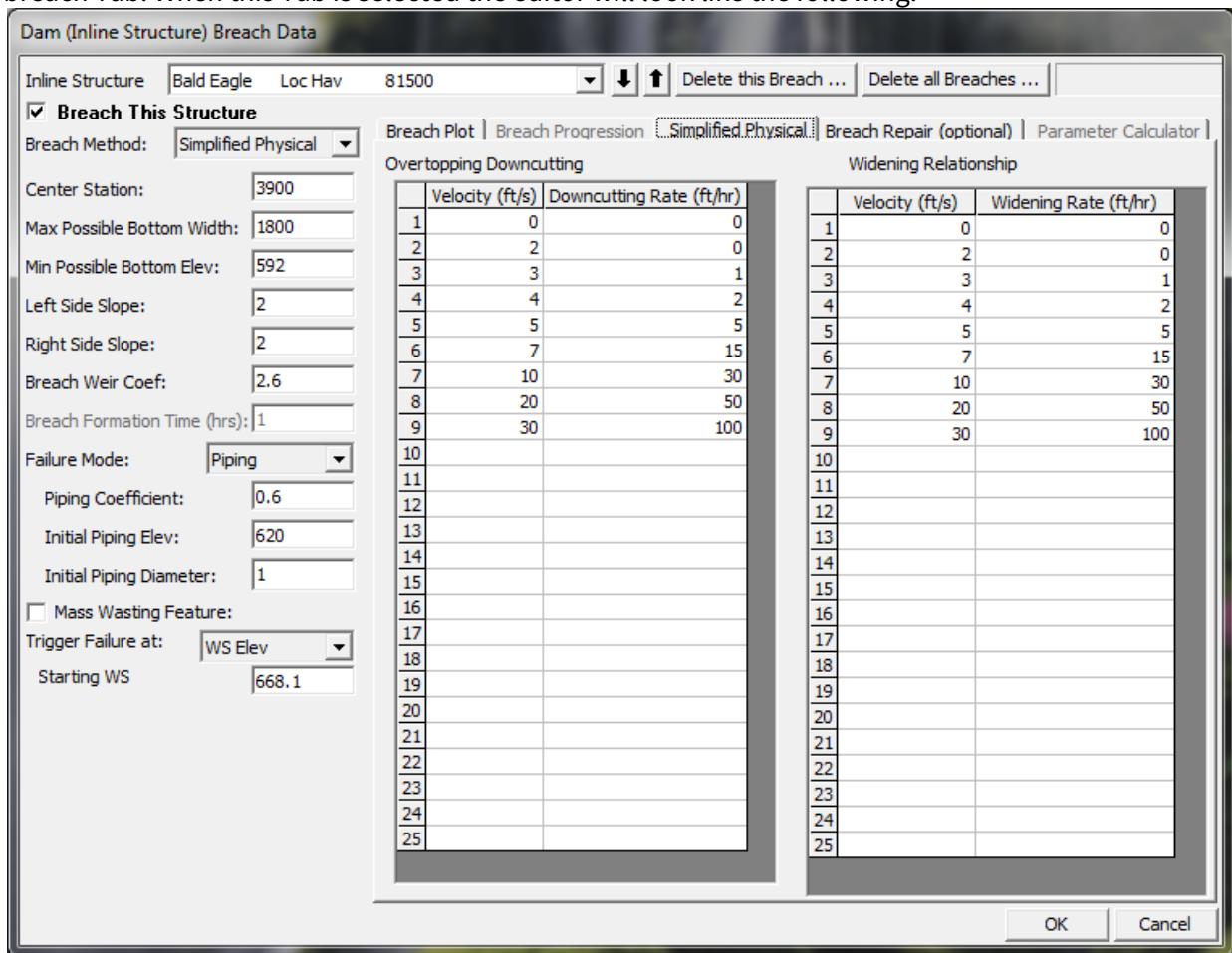


Figure 14 10. HEC-RAS Simplified Physical Breach Option.

As shown in Figure 14-10 above, the user is required to enter Velocity versus Down-cutting erosion rates and velocity versus erosion widening rates. This data is often very difficult to come by. User's will need to consult with Geotechnical engineers to come up with reasonable estimates of this data for your specific Levee or Dam. Another way to estimate this information is to try to derive it by

simulating a historic Levee or Dam breach, and adjusting the velocity versus erosion rate data until the model simulates the correct breach width and time. This is obviously an iterative process, and may require the user to perform this at multiple locations to see if there is a consistent set of erosion rates that will provide a reasonable model for simulating Levee breaches (or Dams) in your geographical area.

We realize that this data is not readily available for any specific levee or dam. The hope is that over time we will be able to develop guidelines for these erosion rates based on analyzing historical levee and dam breaches.

Once all of the Dam Breach data are entered, press the **OK** button to have the data accepted. However, the data is not saved to the hard disk at this point. You must save the currently opened plan in order for the breach information to be saved to the hard disk.

## Breach Growth Shape

The shape and progression of the breach depends on the user entered data. If the User has selected to use the "Simplified Physical Breaching" option, the Breach Growth is dependent upon the user entered Downcutting and Breach Widening rates entered by the user, as well as the event being modeled. If the user has selected the "User Entered Data" breaching method, then the Breach Formation shape is dependent upon the Breach Bottom Width, height of the structure, and the Breach Formation Time. For example, consider a breach with the following parameters: Final Bottom Width is 400 feet, Final Bottom Elevation is 100 feet (top of weir is 200 feet—full breach is 100 feet deep), Formation Time is 4 hours, Breach Progression is Linear. The breach has to grow 400 feet wide in 4 hours so the growth rate is 100 feet/hour (this growth rate is used for both the horizontal and vertical growth rate by default). The breach will start out as a tiny trapezoid (or rectangle if side slopes are zero) at the top of the weir based on the Center Station. The trapezoid will grow such that after one hour, the breach will have just reached the Final Bottom Elevation (100 feet deep) and the breach will be 100 feet at wide (at the bottom). Over the next three hours, the breach will grow horizontally from 100 feet wide to the final width of 400 feet wide, at the bottom. The side slopes of the trapezoid always remain the same. In the less common situation where the breach is deeper than it is wide, the growth rate is based on the vertical depth divided by the formation time—the maximum width would be reached before the maximum depth.

The user has the option to specify a vertical/horizontal growth rate. The default value is 1.0. However, if the user feels the vertical growth rate should be slower or faster than the horizontal growth rate, they can select this option and enter a ratio of vertical to horizontal growth rate (e.g. a value of 0.5 would produce a vertical growth rate that is half the speed of the horizontal growth rate).

If the Breach Progression is non-linear, then the horizontal growth will be adjusted, as needed. Progression in the vertical direction will match the horizontal growth (taking into account whether the Same/Proportional option has been selected). For the above example, if data has been entered such that the breach is only 10% (0.1) formed after 25% of the time, then the breach would be 40 feet wide after one hour. It would be 40 feet deep for Same Growth... or, alternately, it would be 10 feet deep for Proportional Growth.

If the Failure Mode is changed to Piping, then the breach starts out as a tiny square (or rectangle) based on the center station and Initial Piping Elevation. For the original example with Same Growth and an Initial Elevation of 120 feet, the sides of the square will grow at the rate of 100 feet/hour. The vertical growth is split between up and down. After 6 minutes, the square will be 10 feet on a side. The top of the square would be at an elevation of 125 feet and the bottom at an elevation of 115 feet.

Once the bottom of the square reaches the Final Bottom Elevation, all of the growth is applied upward. When the elevation of the top of the square is higher than the water surface elevation in the breach, it is assumed the breach will cave in. The breach will now be an open rectangle with the current bottom elevation and current bottom width (the rectangle extending vertically to the top of the weir). If the breach has not yet reached the Final Bottom Elevation, it will grow downward at the full vertical growth rate. The bottom of the breach will continue to grow horizontally at the same rate (or adjusted rate for non-linear). If the side slopes are not zero (vertical), then the side slopes of the rectangle/trapezoid will progress from vertical to the maximum side slope, linearly over the remaining time (or adjusted for non-linear progression based on user selection). So if the Side Slope is 3 and the piping breach becomes an open breach after one hour, then the side slope would be 1 at the end of the second hour, 2 at the end of the third hour, and 3 at the end of the fourth hour. If the water surface remains high enough, then the piping breach will not turn into an open breach until the top of the piping breach reaches the top of the weir. It will then grow to the final trapezoidal shape as previously described.

For a Piping failure with Proportional Growth, the piping breach would be a rectangle that grows vertically at 25 feet/hour. Whether the breach is growing up and down, only up, or only down, is the same as before, as is the open breach behavior and the non-linear growth option.

## **Estimating Dam Break Parameters**

If the modeler uses the "User Entered Data" option for the breaching method, it is up to the user to enter the key parameters that must be estimated for the breach dimensions, such as the breach formation time and the maximum size of the breach opening. Several researchers have developed regression equations to estimate breach sizes and times from historical dam breach information. Additionally, a few researchers have tried to develop computer models to simulate the physical breach process. The bulk of the research in this area has been summarized in a 1998 publication entitled "Prediction of Embankment Dam Breach Parameters", by Tony L. Wahl of the U.S. Bureau of Reclamation. Wahl documents the data from most of the historical dam breaches that have occurred in the world, as well as describing the equations and modeling approaches developed for predicting the dam breach parameters.

For the HEC-RAS software, the modeler must estimate the maximum breach dimensions and breach formation time either outside of the HEC-RAS program, or with the Breach "Parameter Calculator" provided on the Breach Data editor. Because the breaching process is complex, it is suggested that the modeler try to come up with several estimates of the breach parameters, and then put together a matrix of potential breach sizes and times. One example would be to use two different sets of regression equations and one of the breach simulation models to estimate the breach parameters. In several studies performed at HEC we have used both the Froelich (1995), *MacDonald\Langridge-Monopolis* (1984), and the Van Thun and Gillete (1990) regression equations, as well as the BREACH model by Dr. Danny Fread (Fread, 1988). All four methods give different answers for the breach dimensions, as well as the time for the breach to form. In general, a range of breach parameter estimates should be run as separate trials within HEC-RAS in order to test the sensitivity of the results to the breach dimensions and times. It is always good to test the sensitivity of the breaching parameters, since they are the most unknown factor in this process.

## Breach Parameter Calculator

To assist users in estimating the Breach dimensions and development time, HEC has added a "Parameter Calculator" to the Breach Data editor. To use this calculator select the **Parameter Calculator** Tab from the breach editor, and the editor will look like the following screen:



Figure 14 11. Breach Parameter Calculator from Regression Equations.

As shown in Figure 14-11 above, the Breach Parameter Calculator contains five regression equations (MacDonald\Langridge-Monopolis; Froehlich 1995; Froehlich 2008; Von Thun and Gillette; and Xu & Zhang 2009). The user is required to enter several parameters that describe the Dam and the volume of water behind the structure at the time of failure. These parameters include: Top of Dam elevation; Breach Bottom Width; Pool elevation at Failure; Pool Volume at Failure; Failure Mode; Dam Crest width, upstream and downstream embankment slopes; Earth Fill Type; Dam Type; and Dam Erodibility factor. Not all values are used for all regression equations as noted on the editor. Some of the variables are specific to the MacDonald equation, and the last two are specific to the Xu & Zhang equation. Once the values are entered the calculator computes Breach Bottom Width; Side Slopes; and Breach Development Times from each of the regression equations. The user can then select the answers from one of the equations to by pressing the **Select** button next to the equation results that they would like to use.

Each of the breach parameter estimates will yield a different outflow hydrograph from HEC-RAS. However, once these hydrographs are routed downstream, they will tend to converge towards a common result. How close they get to each other will depend on the distance they are routed, the steepness of the stream, the roughness of the river and floodplain, and the amount of floodplain storage available for attenuating the hydrograph. If the populated areas below the dam are quite a distance away (say 20 miles or more), then the resulting hydrographs from the various dam breaches

may be very similar in magnitude by the time they reach the area of interest. However, if the areas of interest are closer to the dam, then the resulting breach hydrographs could produce a significant range in results. In this situation, the selection of the breach parameters is even more crucial.

## HEC-RAS Output for Dam Break Analyses

Several plots and tables are available for evaluating the results of a dam break analysis within HEC-RAS. Graphics include cross section, profile, and 3 dimensional plots, all of which can be animated on a time step by time step basis in order to visualize the propagation of the flood wave. An example cross-section plot of a dam while it is breaching is shown in Figure 14-12. Additionally, the corresponding water surface profile for the same instance in time is shown in Figure 14-13. Hydrographs can be viewed at any location for which the user requested hydrograph output. Shown in Figure 14-14 is a series of hydrographs from the breach shown in the previous figures. These hydrographs represent the flow leaving the dam and then subsequent locations downstream as the flood wave moved through the river system.

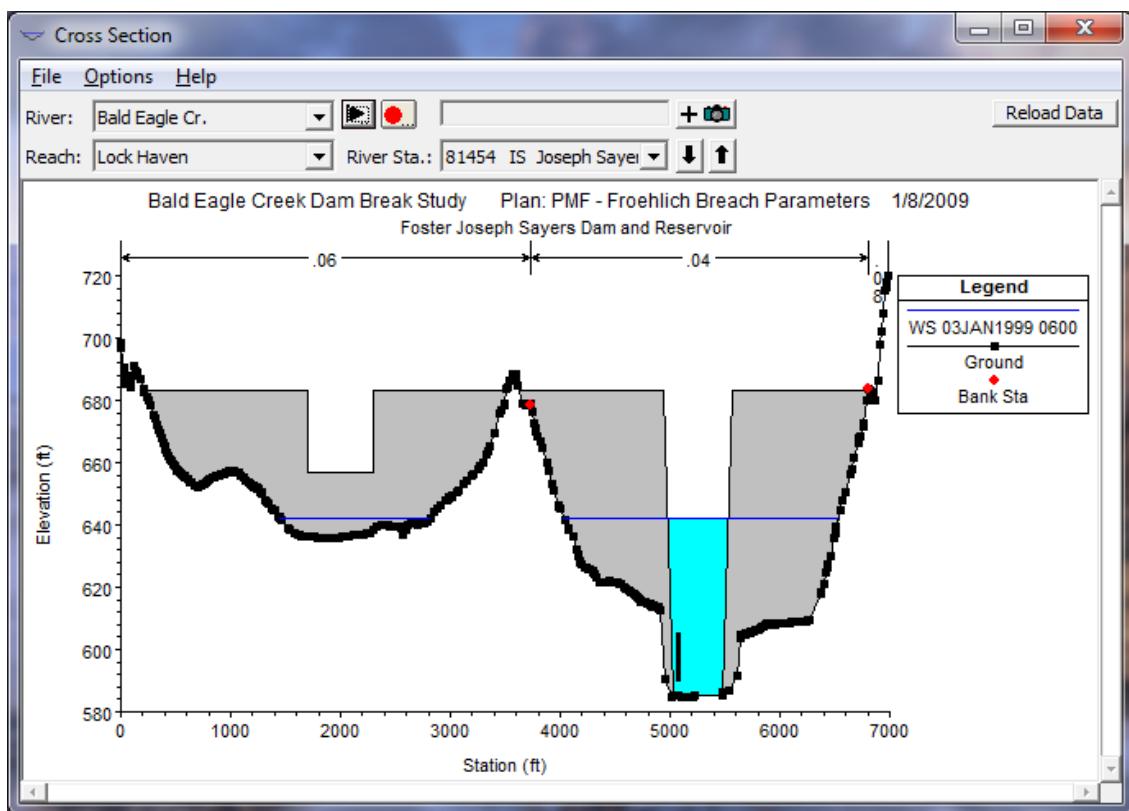


Figure 14 12. Example Plot of Dam While Breaching

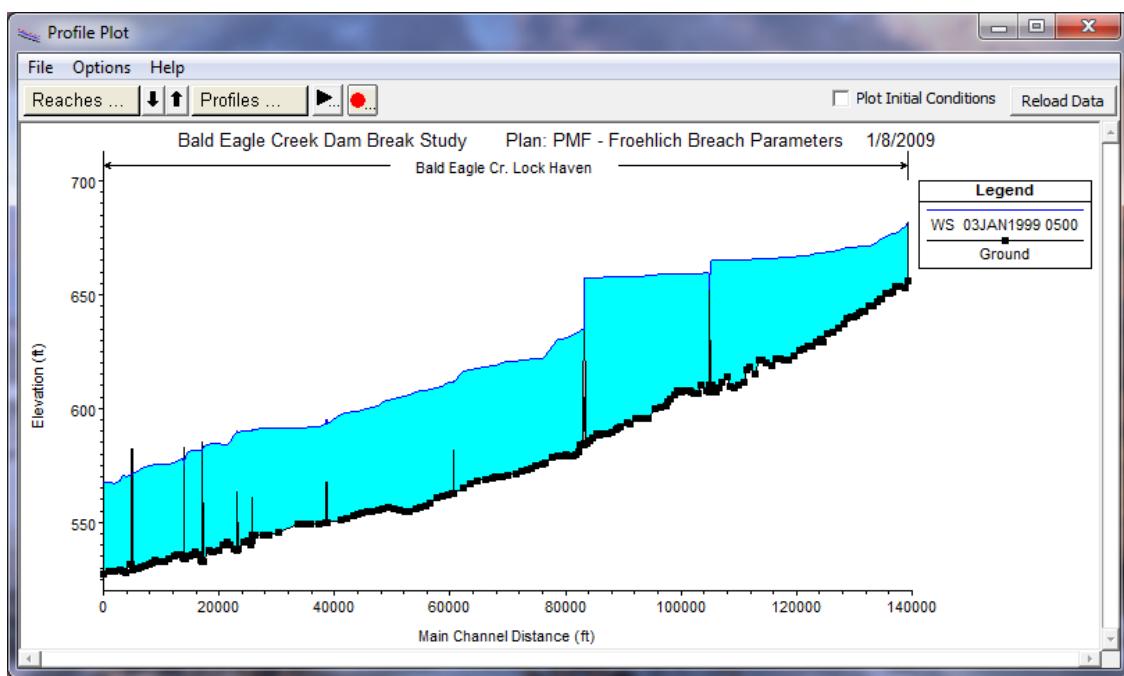


Figure 14 13. Example Profile Plot of Dam Breaching

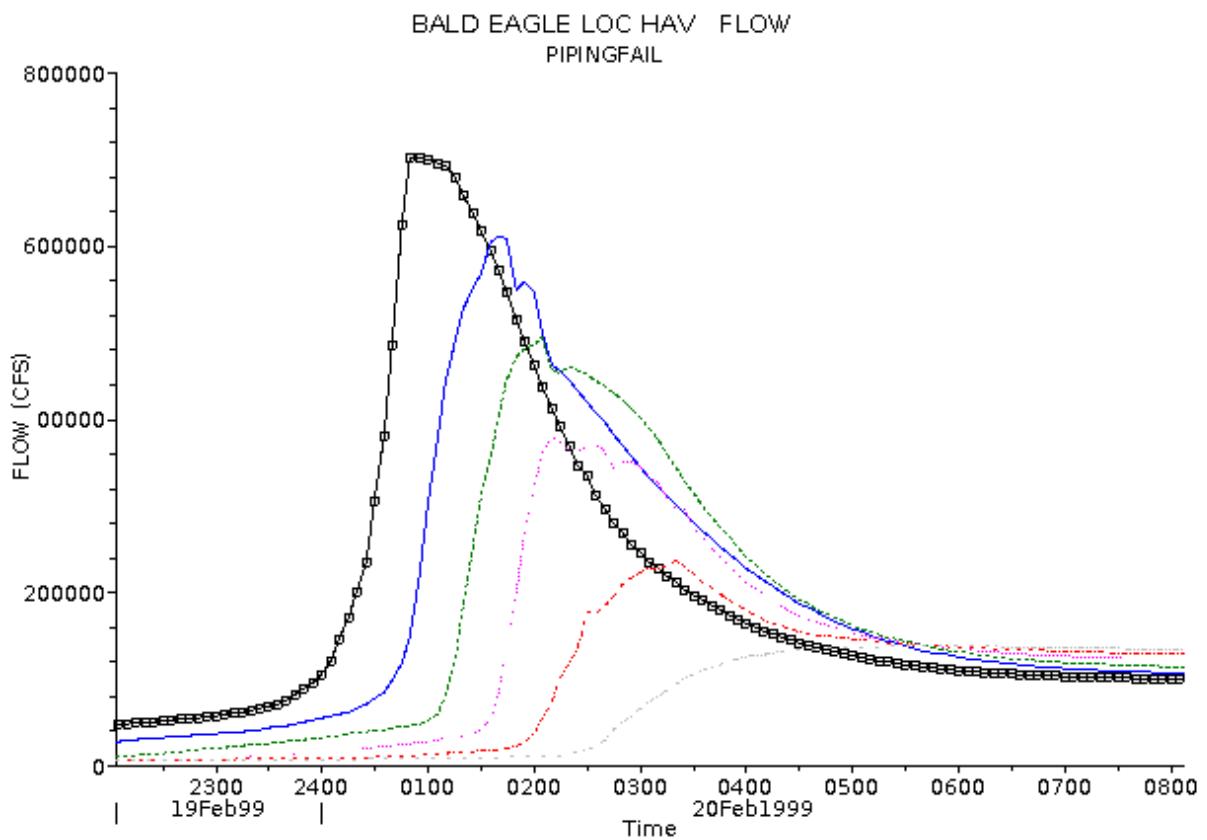


Figure 14 14. Flow Hydrographs from Dam to Downstream Locations

## Levee Overtopping and Breaching

Levee overtopping and breaching can be analyzed within HEC-RAS by modeling the levee as a lateral structure. When modeling a levee with a lateral structure, the area behind the levee should not be included in the cross section data of the main river. The cross sections should stop at the top of the levee. The lateral structure (levee) can be connected to a 2D Flow Area, a storage area, or another river reach. The strategy for modeling the area behind the levee will depend upon what will happen to the water if the levee overtops or breaches. If the water going over or through the levee will take many flow paths and have varying water surface elevations, then a 2D Flow Area is probably the most appropriate for modeling the area behind the levee. If the water going over or through the levee will fill up like a bathtub (Level pool water surface), then a storage area would be the most appropriate for modeling the area behind the levee. If the water will continue to flow in the downstream direction (and acting like a separate 1D channel), and possibly join back into the main river, then it may be more appropriate to model that area as a separate river reach. Shown in Figure 14-15 is an example schematic with a levee modeled as a lateral structure connected to a storage area to represent the area behind the levee. An example cross section with a lateral structure (levee) on the right hand side is shown in Figure 14-16.

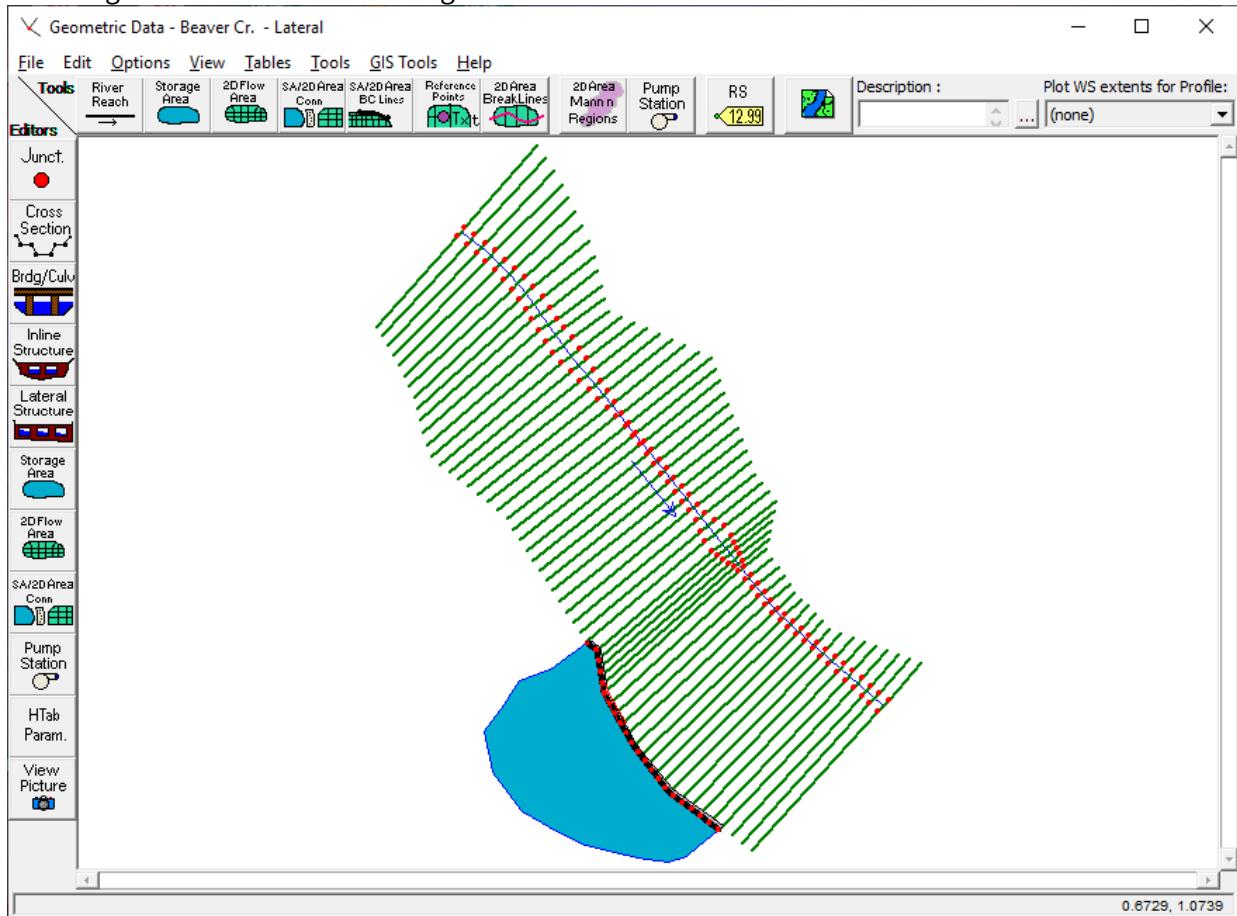


Figure 14-15. Schematic with Example Levee and Storage Area.

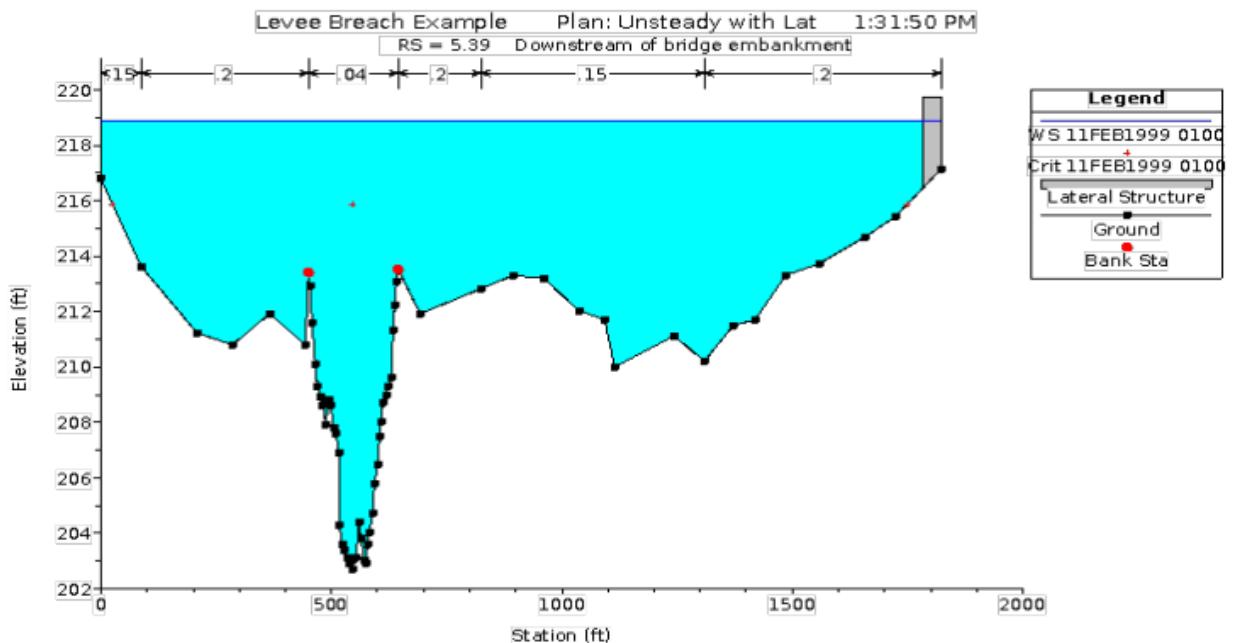


Figure 14-16. Example Cross-Section with Lateral Structure.

The user defines the levee by entering a series of station and elevation points that represent the top of levee profile. This station and elevation data is then used as a weir profile for calculating the amount of water going over top of the levee. An example levee entered as a lateral structure is shown in Figure 14-17.

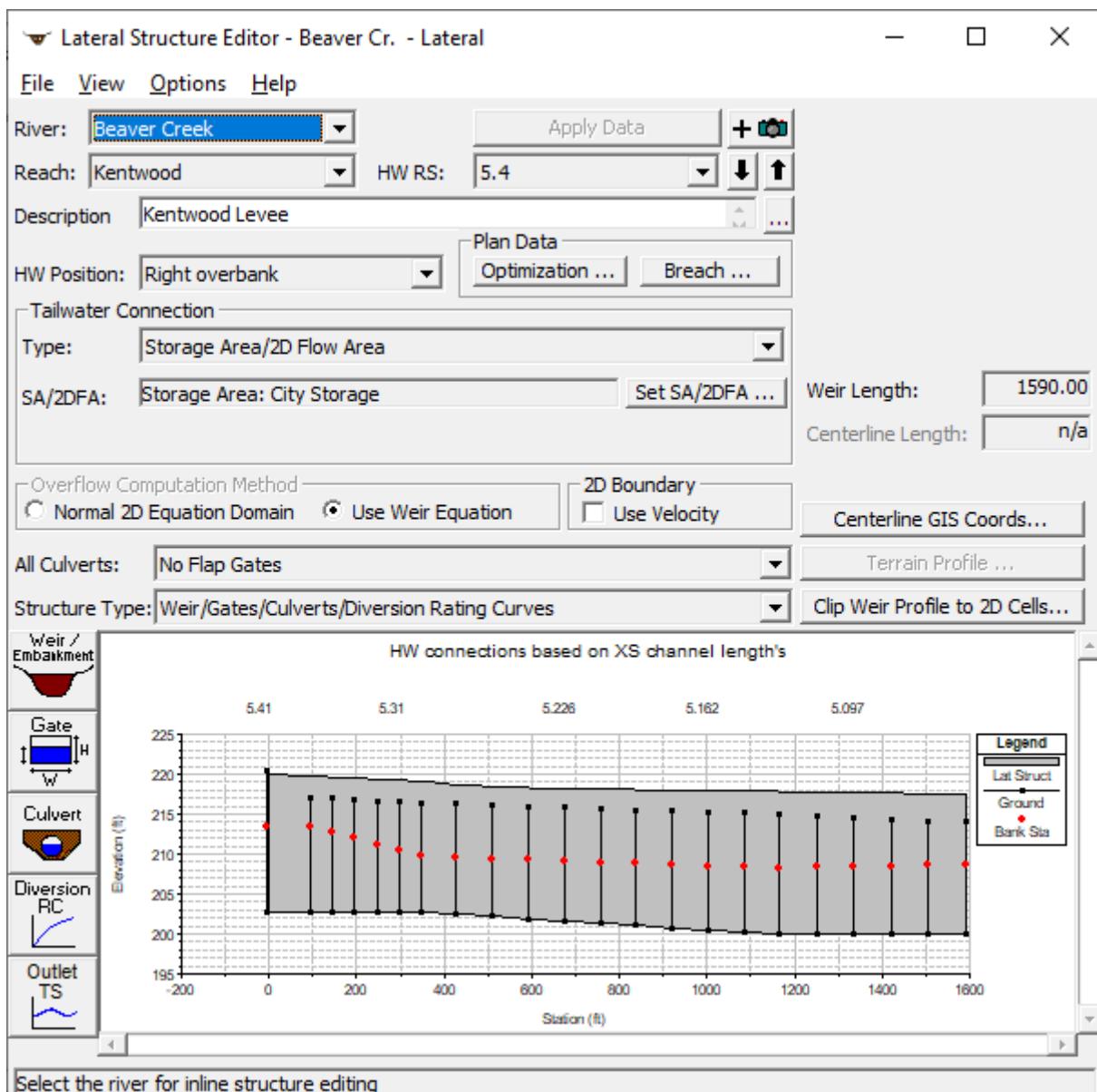


Figure 14-17. Lateral Structure Editor with Levee Modeled as a Weir.

In the example shown in Figure 14-17, the levee is connected to a storage area that will be used to represent the area behind the levee. As the levee overtops and/or breaches, the storage area will fill up until it reaches the same elevation as the water in the river. After the flood passes, the water in the storage area can pass back out any breach that may have occurred.

The levee information is entered as station and elevation data in the Lateral Weir/Embankment editor shown on the Lateral Structure editor. The station elevation data represents the top of the levee. The levee information is entered from the upstream end of the levee to the downstream end of the levee. An example of this editor with levee data is shown in Figure 14-18.

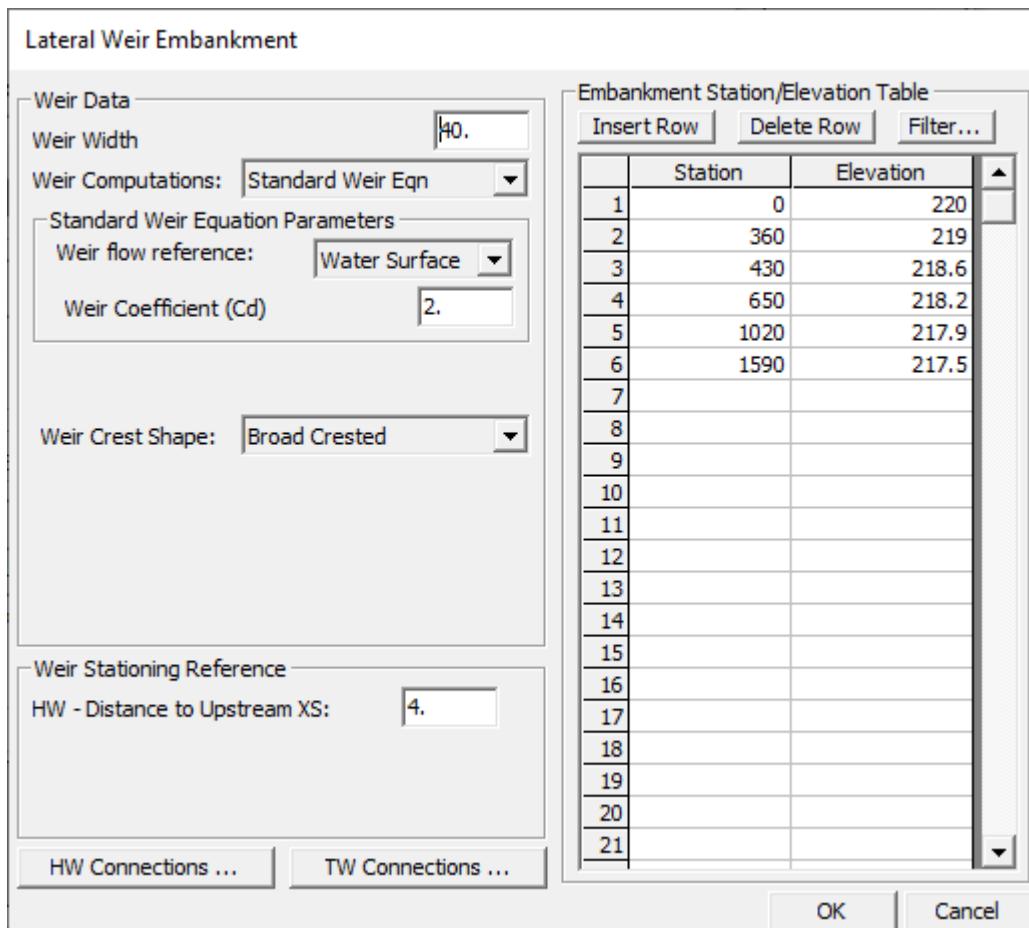


Figure 14 18. Lateral Weir/Embankment Editor with Example Levee Data.

As shown in Figure 14-18, the user enters the width of the levee (which is only used for drawing purposes); the head reference for weir flow calculations; the lateral weir coefficient; the weir crest shape; the distance that the upstream end of the levee is from the nearest upstream cross section; and the station and elevation data representing the top of levee. For more information on this editor, see Lateral Structures in Chapter 6 of this manual.

Once the physical levee information is entered, the user can press the **Breach** button in order to bring up the levee breach editor. The user can have a single breach location per levee (Lateral structure). If more than one levee breach location needs to be modeled, then break the levee into multiple lateral structures (one for each levee breach location). An example of the levee breach editor is shown in Figure 14-19.

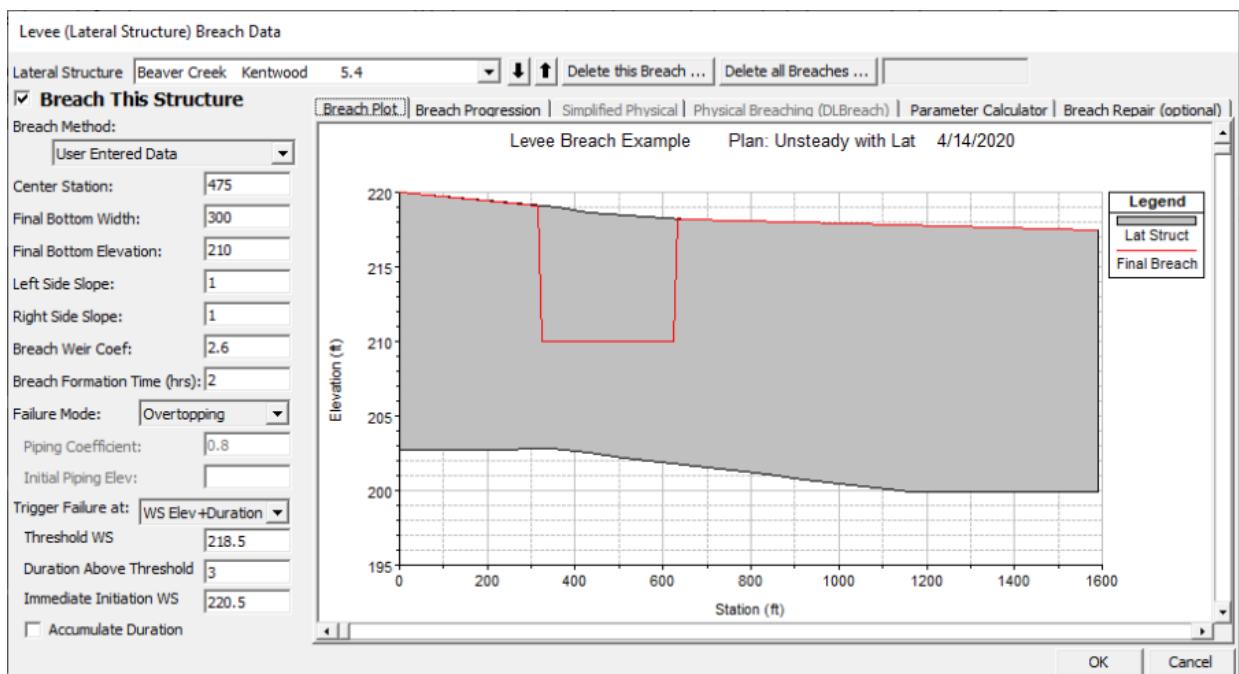


Figure 14-19. Levee Breach Editor with Example Levee and Breach.

As shown in Figure 14-19, the information required to perform a levee breach is the same as performing a dam break. To get the details of each data field, please review the information found under the Dam Break section above in this chapter.

After all of the data are entered and the computations are performed, the user can view output for the lateral structure (levee). Plots such as the profile plot, lateral structure hydrographs, and storage area hydrographs, can be very helpful in understanding the output for a levee overtopping and/or breach. Shown in the figure below is an example profile plot with a levee breach. Shown in Figure 14-21 is a stage and flow hydrograph plot for the lateral structure. In this plot there are three stage lines and three flow lines. The stage lines represent: the stage in the river at the upstream end of the levee (Stage HW US); the stage in the river at the downstream end of the levee (Stage HW DS); and the stage in the storage area (Stage TW). The river is always considered to be the headwater, and the storage area is the tailwater. The flow lines on the plot represent: the flow in the river at the upstream end of the levee (Flow HW US); the flow in the river at the downstream end of the levee (Flow HW DS); and the flow leaving the river over the lateral weir to the storage area (Flow Leaving).

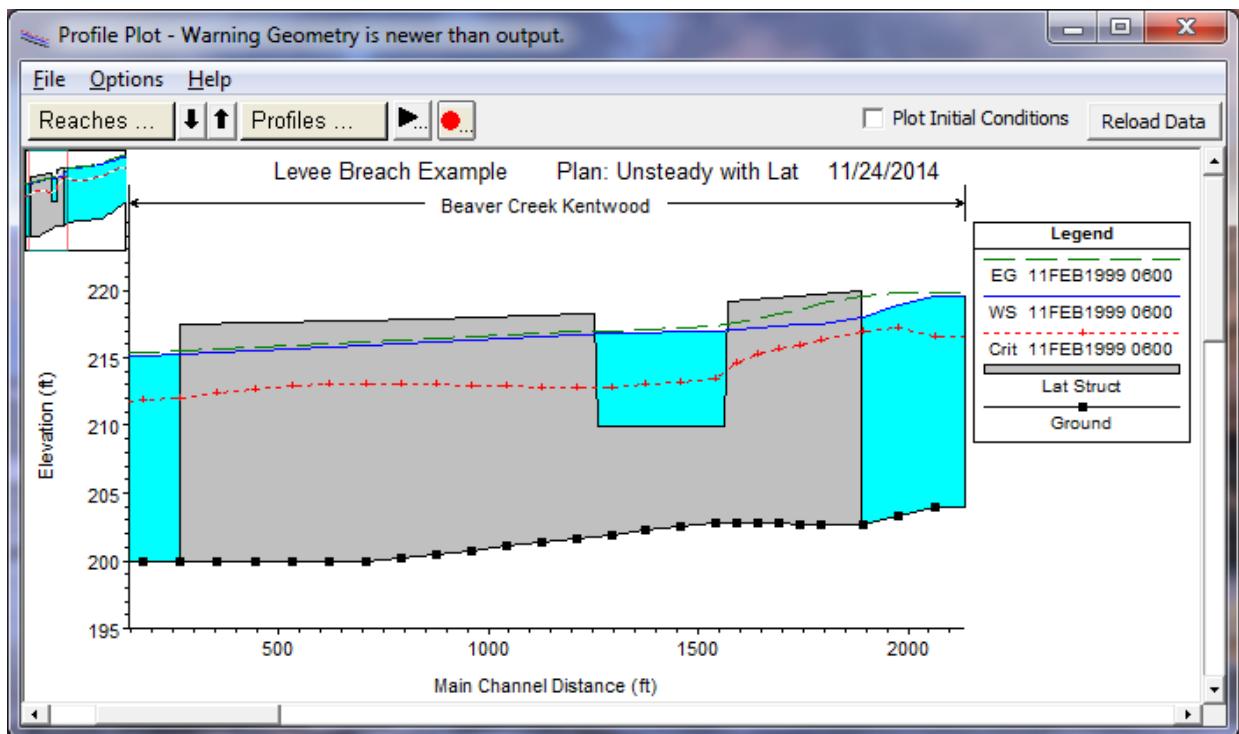


Figure 14-20. Profile Plot with Levee Breach.

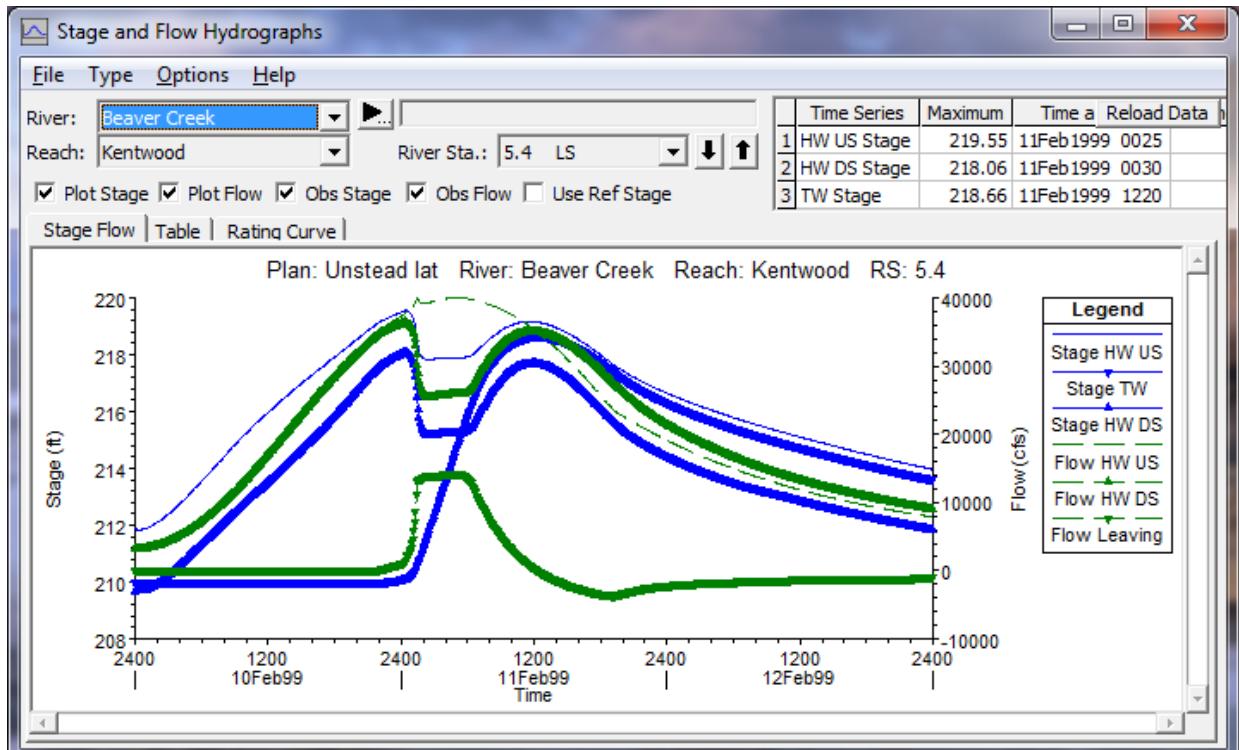


Figure 14-21. Lateral Structure Stage and Flow Hydrographs

In addition to the profile plot and the lateral structure hydrographs, it is a good idea to plot the stage and flow hydrographs for the storage area. This allows the user to easily see the amount of flow coming into and out of the storage area, and the change in the water surface elevation. Shown in Figure 14-22 is the stage and flow hydrograph for the storage area in this example.

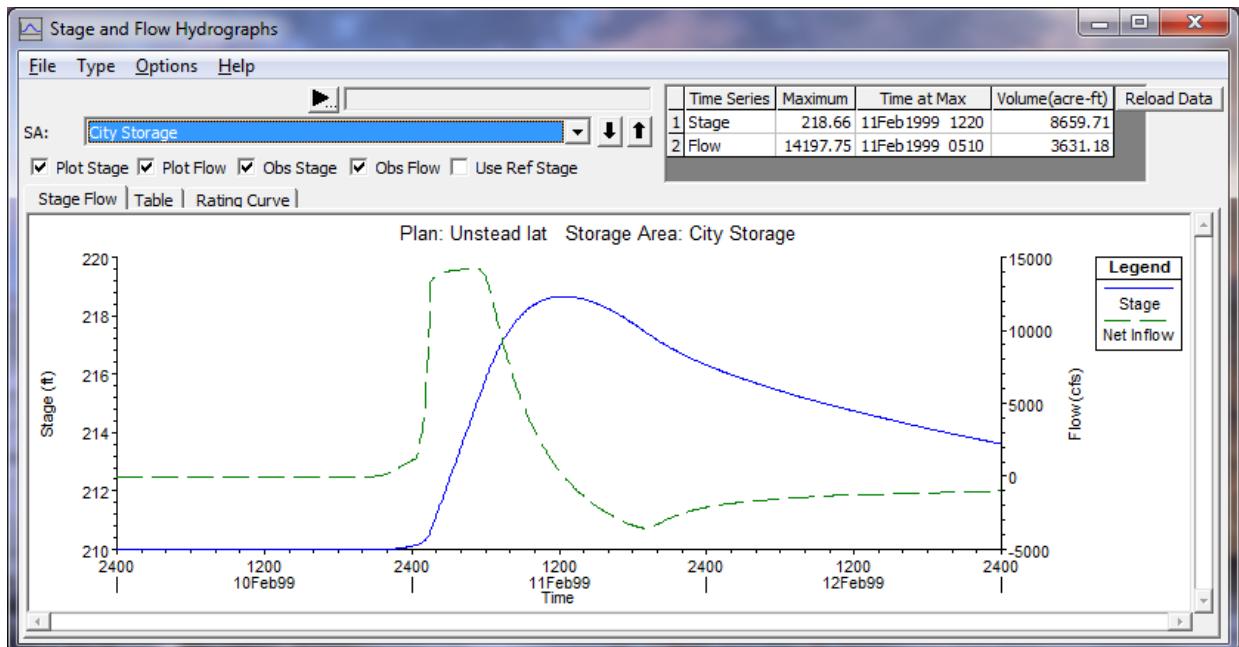


Figure 14-22. Stage and Flow Hydrograph Plot for Storage Area.

Referring to Figure 14-21 and Figure 14-22, as the levee breaches, the flow going into the storage area and the stage increase quickly, while the stage and flow in the main river drop. In addition to the graphics in HEC-RAS, tabular results are also available. Shown in Figure 14-23 is a detailed output table for the lateral structure. The user can select a specific time line for viewing the output by selecting a specific profile. The profiles are labeled by the date and time they occurred in the model simulation.

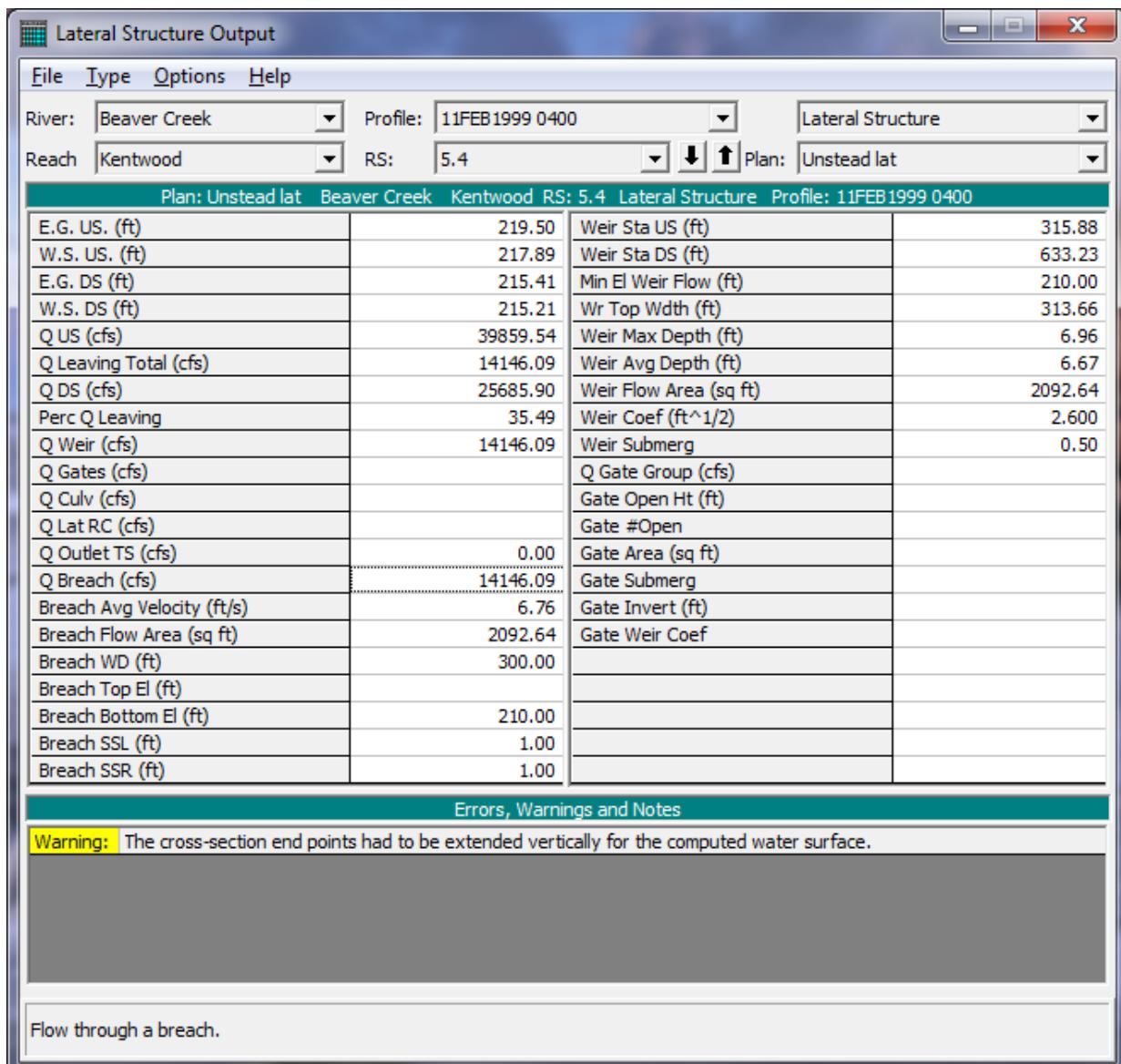


Figure 14-23. Detailed Tabular Output for Lateral Structure.

## Modeling Pump Stations

Pump stations can be connected between storage areas; a storage area and a river; between a storage area and a 2D Flow Area; between two 2D Flow Areas; Between a 1D river and a 2D Flow Area; and between river reaches. HEC-RAS allows up to ten different pump groups at a pump station, and each pump group can have up to ten identical pumps. Each pump can have its own on and off trigger elevation. To learn how to connect a pump, enter pump data, and use pump override rules, please review the section on pumps in Chapter 5 of this user's manual.

Pump stations can be used for many purposes, such as pumping water stored behind a levee (interior sump) into the main river. An example schematic of an interior ponding area behind a levee is shown in Figure 14-24. Note that the pump is connected from the storage area to a river station at

the downstream end of the levee.

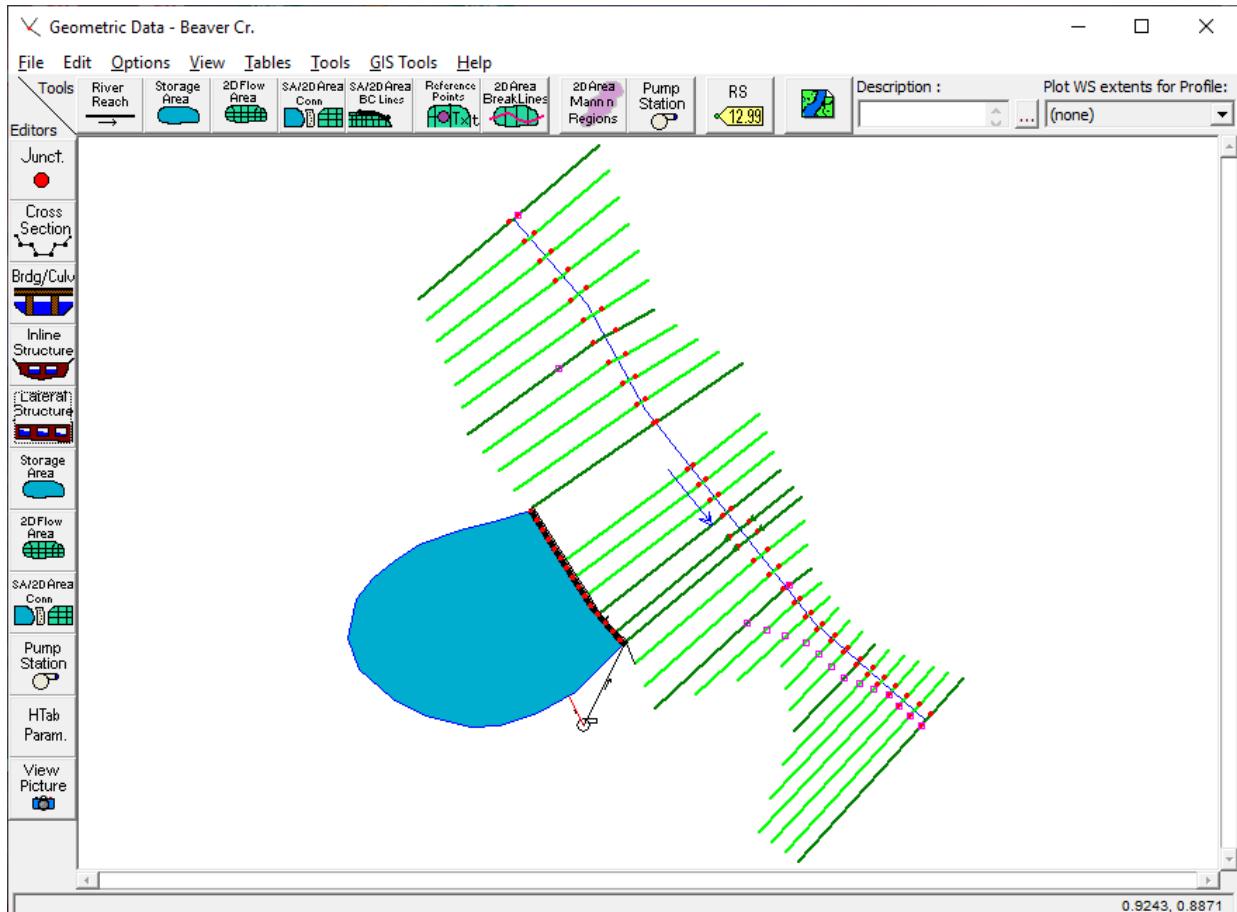


Figure 14 24. Example Pumping Station for Interior Ponding Area

In the example shown in Figure 14-24, a lateral structure was entered to represent the levee. This structure has a gravity draining culvert with a flap gate. The flap gate only allows water to drain from the storage area to the river. Additionally, a pump station is included to pump flows over the levee during a rainfall event. The pump station was drawn by selecting the **Pump Station** tool, then drawing a connection from the storage area to the cross section at river station 5.39.

In this example, there is a hydrograph attached to the upstream end of the river reach, which represents the incoming flood wave to this reach. There is also a lateral inflow hydrograph attached to the storage area, which represents the local runoff collecting behind the levee. The pumps are used to pump water from the storage area, over the levee, to the river. The top of the levee is at elevation 220 feet. Therefore, the pump station is constantly pumping to a head of 220 feet. The Pump station data editor is shown in Figure 14-25.

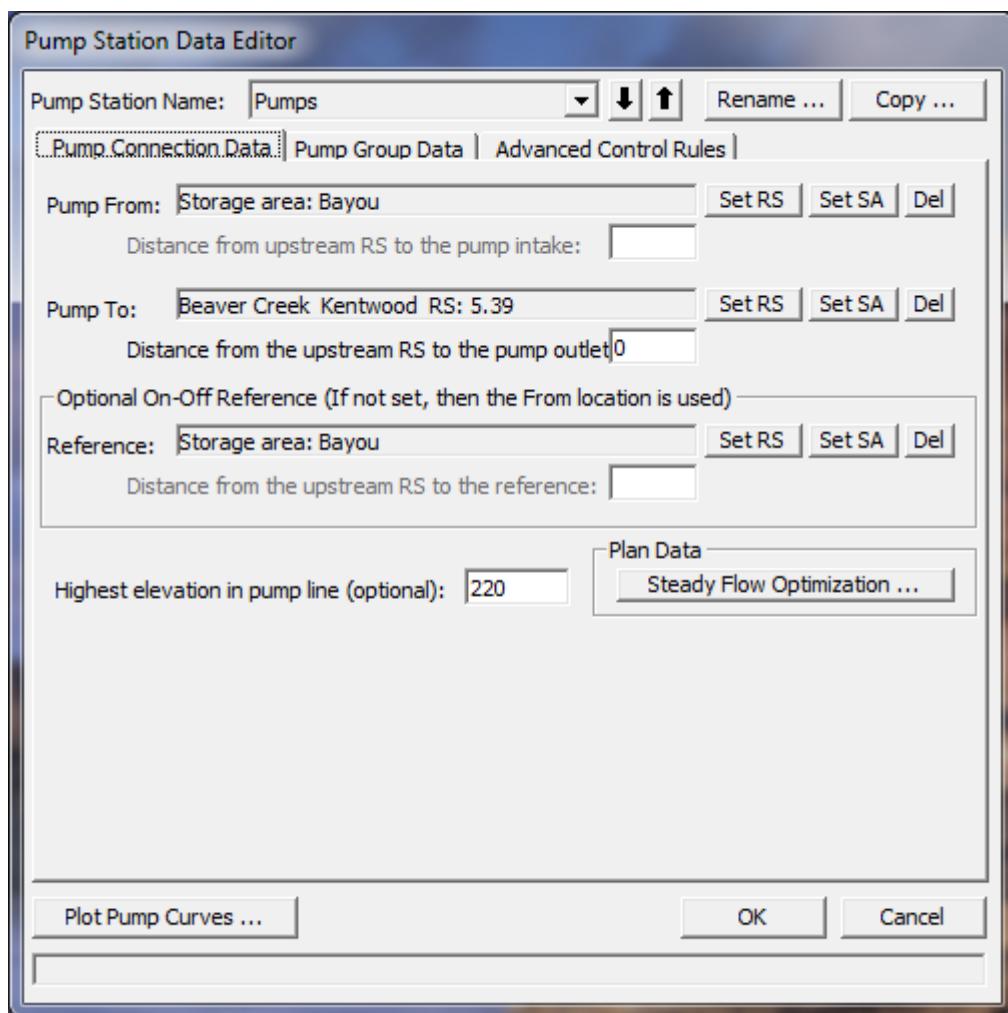


Figure 14 25. Pump Station Data Editor with Example Data

The second tab on the editor brings up the **Pump Group Data** (Figure 14-26). As shown in Figure 14-26, there is one pump group with three identical pumps (pumps are the same size and flow capacity). However, each of the pumps has a different on and off trigger elevation. The pump efficiency curve is used for all three of the pumps.

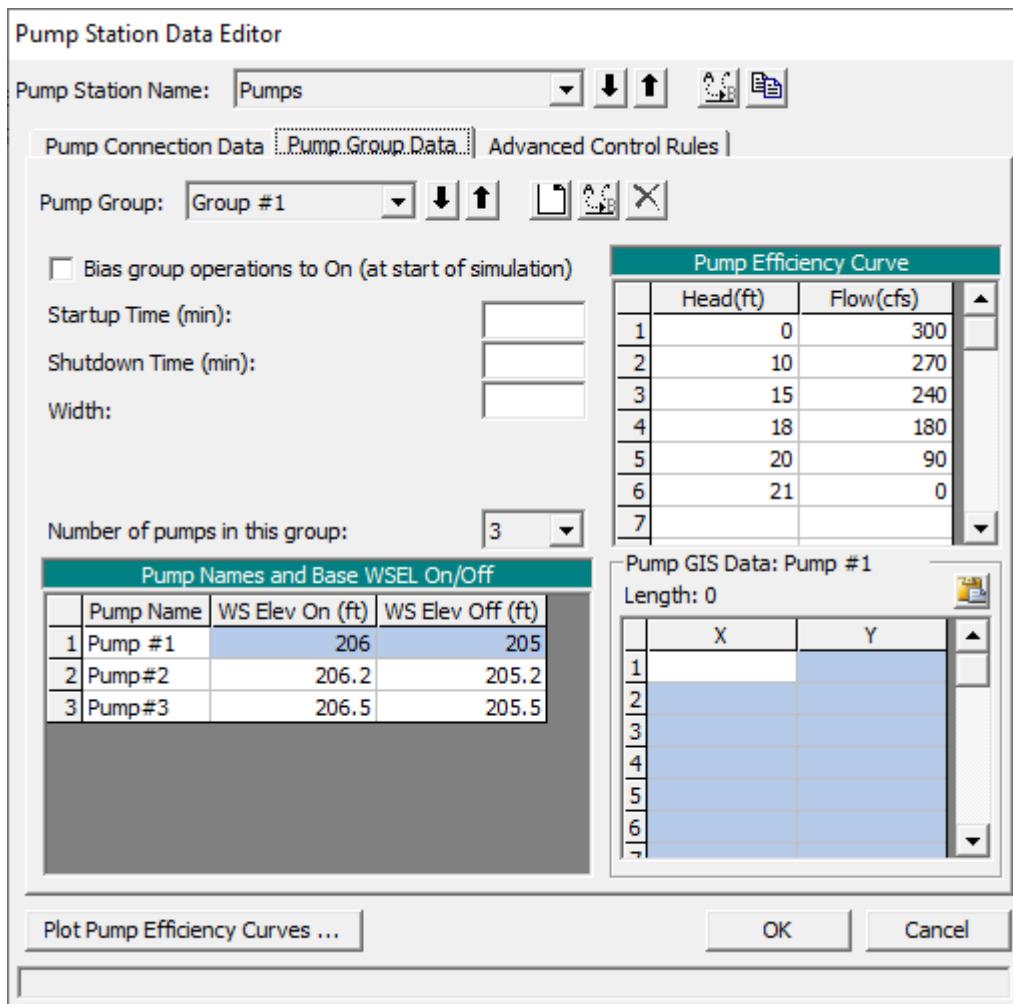


Figure 14 26. Pump Group Data on Pump Editor.

The third tab, **Advanced Control Rules**, allows the user to enter rules to override the normal pump station operations. When this tab is selected a window will appear as shown in Figure 14-27.

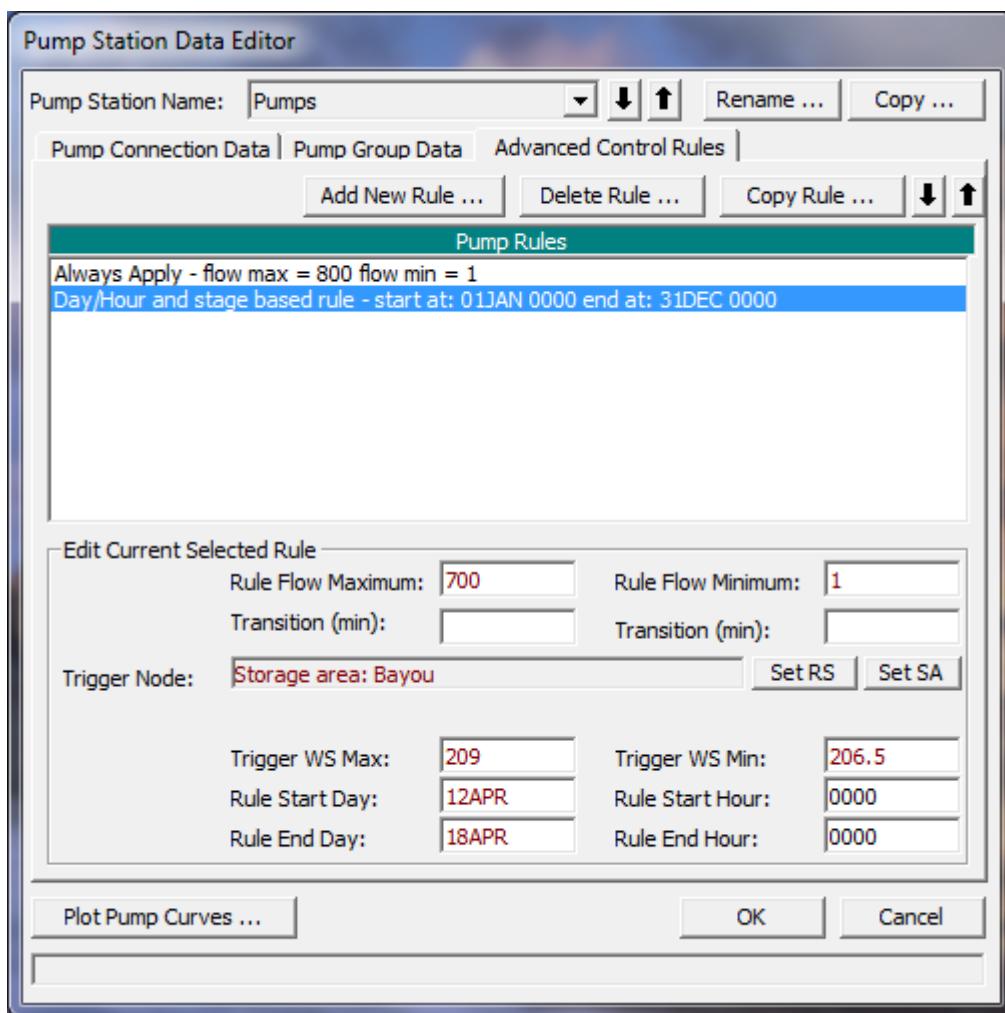


Figure 14 27. Advanced Control Rules editor for Pump Station.

As shown in Figure 14-27, two rules have been applied to this pump station. The first rule sets an absolute maximum pump flow of 800 cfs and a minimum of 1cfs for the entire pump station. This rule will always be applied. The second rule sets a maximum flow of 700 cfs to be applied only between 12 April 0000 and 24 April 0000, but only if the water surface at storage area Bayou is greater than 209. Also, the second part of this rule sets a minimum flow for the same time period, but only if the stage at storage area Bayou is less than 206.5. Details of how to use the rules can be found in chapter 6, under the section on Pumps.

After the computations are performed, the user can view output for the pump station by selecting the stage and flow plotter, then selecting Pump Stations from the Type menu at the top of the window. An example stage and flow plot for the pump station is shown in Figure 14-28. As shown in the figure, the stage for the tailwater location (Stage TW) is a constant 220 ft. This is due to the fact that the pump is constantly pumping over the levee at elevation 220. The stage at the headwater location (stage HW) is the water surface elevations in the storage area. The storage area elevation starts out at an elevation of 205 ft., goes up to around 206.6, and then back down to around 205.1. The flow through the pumps was zero until an elevation of 206 was reached within the storage area, which triggered the first pump. The second pump turned on when the storage area got to elevation 206.2, and the third at elevation 206.5. On the falling side of the hydrograph the pumps began to turn

off as the stage went down in the storage area. Shown in Figure 14-29 are the stage and net inflow to the storage area. The net inflow represents all the inflows minus the outflows at each time step.

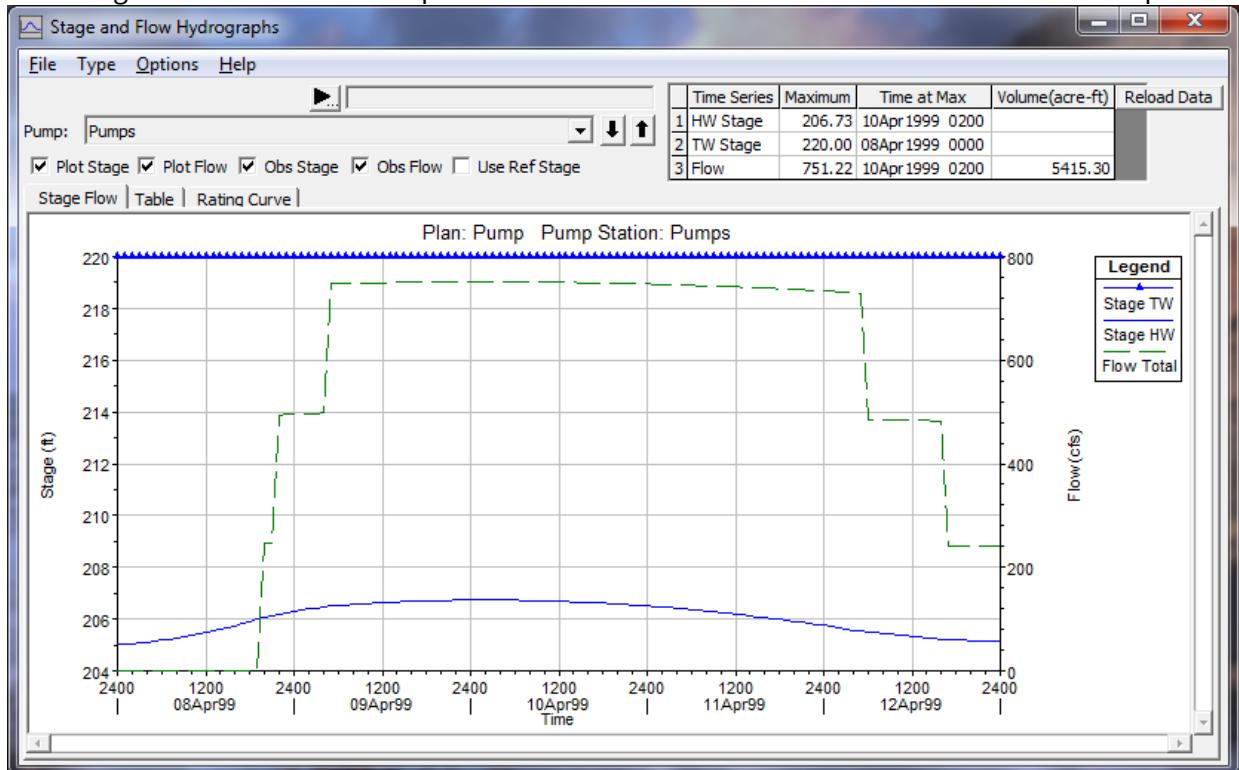


Figure 14 28. Stage and Flow Hydrographs for Pump Station

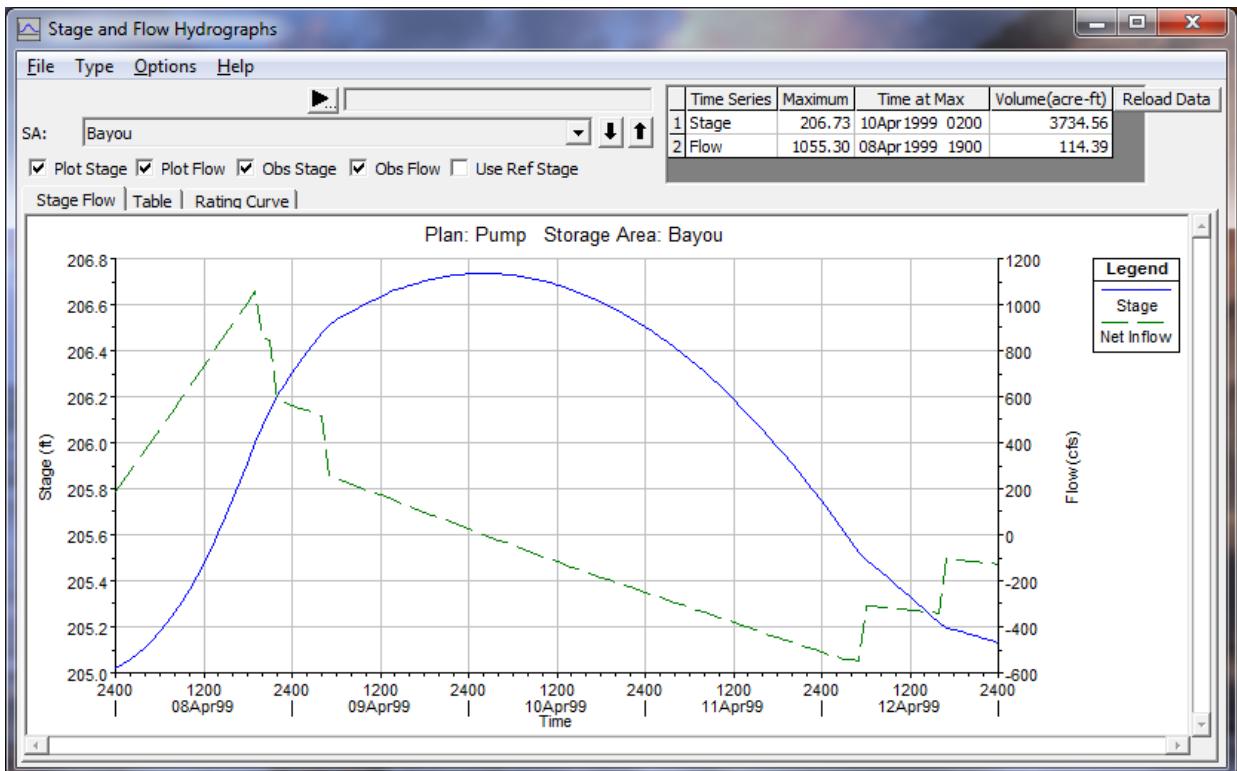


Figure 14 29. Stage and Flow Hydrographs for the Storage Area

## Navigation Dams

This section discusses the navigation dam option in HEC-RAS. For a navigation dam, the program will try to maintain both a minimum and maximum water surface at one or more locations along a navigation channel. The program does this by controlling the gate settings on an inline structure. The user enters a target water surface (and various other calibration data) and the program will adjust the gate settings at user specified time intervals in order to meet the target water surface as closely as possible. This section describes the data requirements for a navigation dam and includes a general discussion of how the gate operations are performed.

The first step in modeling a navigation dam is to add the physical data for the navigation dam by selecting the Inline Structure option on the Geometry Data editor and entering the appropriate information. The next step is to add the inline structure as a boundary condition on the Unsteady Flow editor and then click the **Navigation Dams** button. The editor, as shown in Figure 14-30, will appear (note: the fields will be blank when the editor first appears).

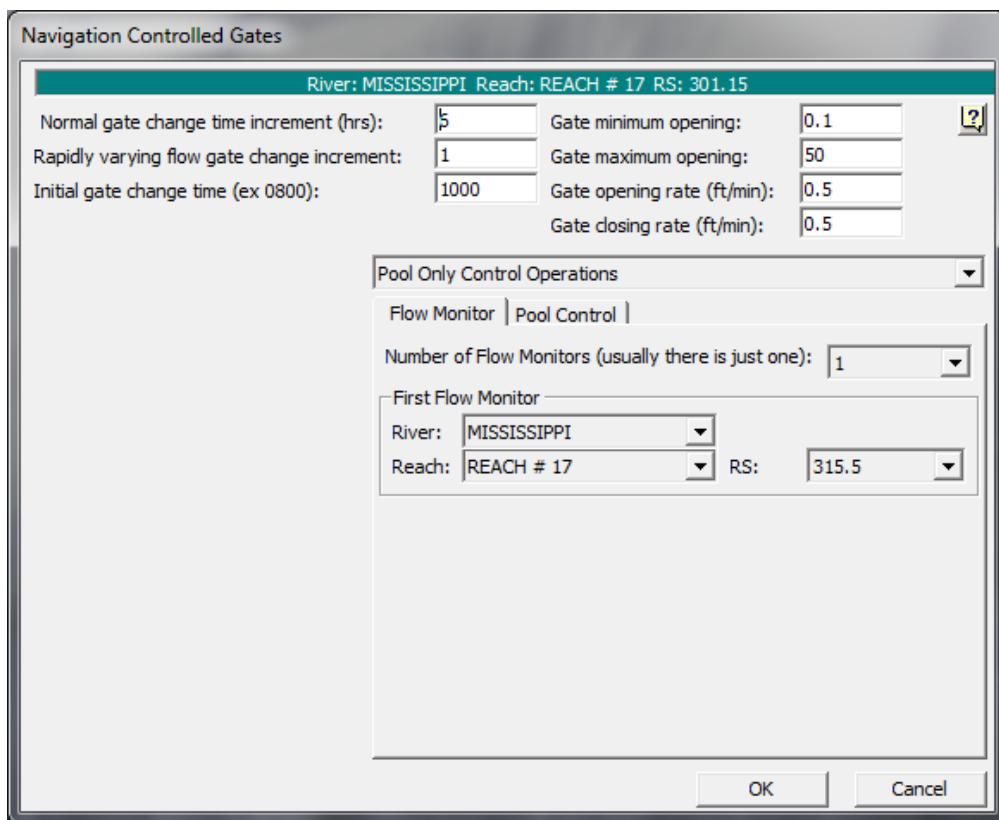


Figure 14-30. Navigation Dam Editor with Flow Monitor

*Normal gate change time increment* – This field states how often the program will adjust the gate settings. In the example shown in Figure 14-30, the program will only make adjustments to the gates every six hours under normal operations.

*Rapidly varying flow change increment* – This field represents the minimum length of time between gate setting adjustments. For example, during rapidly changing conditions, the program can adjust the gates up to once an hour in order to maintain the appropriate water surfaces.

*Initial gate change time* – This field is the time (military style) for when the first gate change will take place. In this example, it is 10:00am. If the simulation starts after 10:00am then the gates will be first adjusted at 4:00pm, 10:00pm, or 4:00am as appropriate.

*Gate minimum opening* – This field is the minimum opening for the first gate group (the first gate group as defined on the Geometry editor). The program will keep the gates on this gate group open to at least 0.1 feet. The other gate groups may be closed completely (see discussion of gate opening and closing below).

*Gate maximum opening* – This field is the maximum opening for the first gate group (the first gate group as defined on the Geometry editor). The program will not allow the gates on this gate group to open more than the specified value. If this field is left blank, then the default is the physical gate maximum opening from the geometric data.

The final two fields [*Gate opening* and *Gate closing rate*] are the maximum speed that the gates in any gate group can be opened or closed. Generally this rate is determined by the physical speed with

which the gates can be adjusted. Sometimes, however, opening or closing the gates too quickly can cause instability in the unsteady solver. In this case, it may be necessary to reduce the opening or closing rate. A shorter time step may also help.

## Pool Only Control

There are several types of navigation dam operations. The simplest is pool only control (as shown in Figure 14-30). In this case, the program tries to maintain the water surface immediately upstream of the dam within user specified targets. In the other operations (see below), the target water surface is located some distance upstream of the dam and there may or may not be limits on the water surface right at the dam.

In order to keep the water surface at the dam within the user specified limits, while only infrequently changing the gate settings (i.e., every six hours), the program needs to know what the approximate inflow at the dam will be some time into the future. This is done by monitoring the flow at an upstream cross section. The user must enter this location. In this example (Figure 14-30), the **Flow Monitor** tab has been activated and the flow monitor location has been entered as river station 315.5. The flow monitor location should be chosen so that the river travel time between the monitor location and the navigation dam is on the order of (or somewhat less than) the normal gate increment. In this example (Figure 14-30) the gate time increment is every six hours, so a location a few hours upstream would be appropriate.

The calibration of the navigation dam control data involves some empirical decisions and trial and error experimentation. This is true of the flow monitor location as well as most of the remaining data explained below.

The flow monitor location must be a normal cross section in the model. This means that cross sections must be extended far enough upstream of the dam to account for this location. Note also that the monitor point can be located upstream of other hydraulic structures, including other navigation dams. As long as another upstream navigation dam does not have a significant storage capacity, it should not affect the results of the flow monitor.

After the flow monitor location has been chosen, the Pool Control tab can be pressed bringing up the editor shown in Figure 14-31.

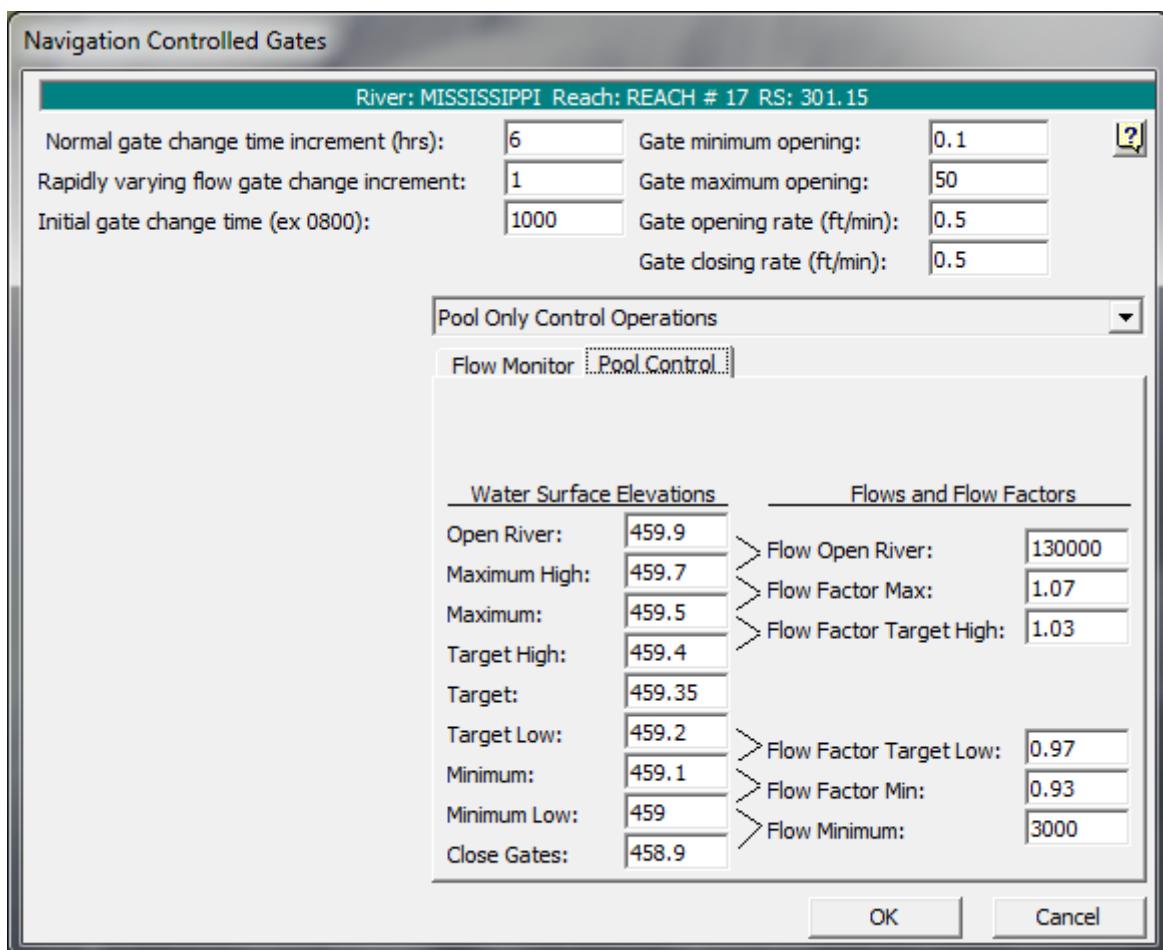


Figure 14 31. Navigation Dam Editor with Pool Control

The user enters a range of water surfaces and corresponding Flow Factors. In this example, the ideal target water surface has been entered as 459.35. The primary target range is from 459.2 (Target Low) to 459.5 (Target High). In general, if the water surface is between Target Low and Target High and it is time to change the gate settings, then the program will adjust the gates to get an average of the current flow at the dam and the monitor flow.

For instance, assume that at time 10:00 the current discharge from the navigation dam is 10,000 cfs, 11,000 cfs of flow is observed at the monitor location, and the water surface at the dam is 459.4 feet. Since 459.4 is in the primary target range, the program will compute the average of the flows, 10,500 cfs. By trial and error, the program will change the gates (and compute the corresponding flow) until there is 10,500 cfs (plus or minus the tolerance) of discharge at the dam. The tolerance is 1% of the flow, in this case 105 cfs. So the program will actually stop iterating whenever it first determines a gate setting that results in a flow that is between 10395 cfs and 10605 cfs. After the gates have been changed, they won't (normally) be adjusted for the next six hours. The flow from the dam will fluctuate as the water surface at the dam fluctuates.

As the water surface at the dam gets out of the primary target range, then the flow (that is, the discharge from the dam) is adjusted by the Flow Factors. In general, when the stage is between Target High and Maximum, then the flow is multiplied by Flow Factor Target High (in this case 1.03). Between Maximum and Maximum High, it is multiplied by at least 1.07. Between Maximum High and water surface Open River, the flow is rapidly increased up to at least Flow Open River (listed as

50,000). Flow Open River does not represent a cap. If the flow at the monitor location gets high enough, the discharge at the dam can go above Flow Open River based on the Flow Factors. Above water surface Open River, all the gates are opened all of the way.

The operations below the target zone work the same way. Flow Factor Target Low and Flow Factor Minimum are applied in the same way. Between Minimum Low and water surface Close Gates, the flow will be rapidly decreased to Flow Minimum, but again, this is not an absolute minimum. If the water surface remains low enough, the program will continue to close the gates and reduce flow. The only absolute minimum is that the program will not close the first gate group below the gate minimum opening.

The water surface targets are basically calibration knobs and no particular water surface targets have to exactly match the operationally prescribed limits on the pool surface. However, the best response will probably be obtained if the Maximum and Minimum are close to the prescribed limits.

## Hinge Point Only Control

The next type of navigation dam operation is hinge point control. This is similar to pool control. The main difference is that instead of the water surface targets being located right at the face of the dam, the water surface targets are located some distance upstream. Figure 14-32 shows the Hinge Point Only editor. (Hinge point control is selected by clicking on the drop down box near the top right of the editor.)

In this example, the navigation dam is located at river station 714.35, and the hinge point is located at river station 728.28. The program will adjust the gates at the dam in order to maintain an approximate water surface of 645.5 feet (the target water surface) at river station 728.28. The target water surfaces and Flow Factors behave the same as for pool control. A flow monitor location is still needed. It should be located an appropriate distance upstream of the hinge point. For this dam, it is located a few hours upstream at river station 750.1.

The Steady Profile Limits Table is an optional feature (see Figure 14-32). It can make the navigation dam operations more robust for rapidly changing flow. It addresses the situation where the water surface for a given flow at the dam diverges significantly from the water surface that would be expected at the dam for a steady state, uniform flow condition.

A typical example is the trailing end of a high flow hydrograph. For instance, the flows at the hinge point and monitor location may have fallen considerably below the Open River condition, but the water surface at the dam is still a little high (compared to the flow). When the program computes a desired flow at the dam, e.g. 10,000 cfs, it adjusts the gates to get this flow. Over the next six hours, however, as the water surface at the dam continues to fall toward a lower equilibrium, the discharge can drop significantly below 10,000. This means that the navigation dam response is either sluggish in returning to the target water surface at the hinge point and/or the gates have to be changed more frequently. This is where the table becomes useful.

The data in this table give the water surfaces at the dam that will produce the target water surfaces at the hinge point for steady state conditions. For this navigation dam, it is desired to keep the water surface at the hinge point between 645.35 and 645.65 feet (Minimum and Maximum values from the water surface elevations table). If, for instance, there is a long term (steady state) flow of 10,000 cfs between the hinge point and the dam, then maintaining a water surface at the dam of 645.19 feet will result in a water surface of 645.35 feet at the hinge point. Similarly, a water surface of 645.59 at the dam will result in a water surface of 645.65 feet at the hinge point, for the same 10,000 cfs flow.

The user can generate these profile limits by putting together a steady flow run from the dam up to the hinge point location. An iterative process of forcing elevations at the dam and computing them at the hinge point is required. The user must find the elevations at the dam that will get to the high and low hinge point elevations from a steady flow backwater computation. Then the values used at the dam to produce the max and min at the hinge point should be entered into this table.

**Navigation Controlled Gates**

River: MISSISSIPPI Reach: REACH # 11 RS: 714.35

Normal gate change time increment (hrs):	6	Gate minimum opening:	0.01
Rapidly varying flow gate change increment:	1	Gate maximum opening:	30
Initial gate change time (ex 0800):	1000	Gate opening rate (ft/min):	0.5
		Gate closing rate (ft/min):	0.5

**Steady Profile Limits Table (Optional)**

	Flow	WSMax	WSMin
1	0	645.7	645.3
2	5000	645.67	645.27
3	10000	645.59	645.19
4	15000	645.45	645.01
5	20000	645.23	644.77
6	25000	644.95	644.45
7	30000	644.56	643.99
8	35000	644.03	643.37
9	40000	643.29	642.41
10	45000	642.24	640.86
11	50000	640.76	640.76
12			
13			
14			
15			
16			

**Hinge Point Only Operations**

Flow Monitor [Hinge\_Control]

River:	MISSISSIPPI
Reach:	REACH # 11
RS:	728.28

**Water Surface Elevations**

Open River:	646.1	> Flow Open River:	50000
Maximum High:	645.9	> Flow Factor Max:	1.07
Maximum:	645.65	> Flow Factor Target High:	1.03
Target High:	645.55		
Target:	645.5		
Target Low:	645.45	> Flow Factor Target Low:	0.97
Minimum:	645.35	> Flow Factor Min:	0.93
Minimum Low:	645.1	> Flow Minimum:	3000
Close Gates:	644.9		

**Flows and Flow Factors**

OK Cancel

Figure 14 32. Navigation Dam Editor with Hinge Control

Continuing on with the 10,000 cfs flow example, before the program starts to iterate, it checks the current water surface at the dam against the table. If the current water surface is between the limits (in this case 645.19 and 645.59), the program continues normally as it would if the table was not being used (that is, the user had left it blank). However, let's assume that the water surface at the dam is 646.0 feet. This would mean that the water surface at the dam is above the limits. In this case the program will temporarily assume a headwater of 645.59 feet at the dam and determine the gate settings that will result in a discharge of 10,000 cfs for this lower, assumed, headwater. After this has been done, the program will use the new gate setting and continue on normally. This gate change will result in the flow at the dam initially being above the 10,000 cfs target. However, as the water surface at the dam drops, the flow should also drop down towards the 10,000 cfs range. This will, hopefully, produce a faster response without over shooting the target water surface at the dam.

If the water surface is on the low side, it works the same way except the lower limit is used. If the water surface at the dam were 645.0 then the gate setting would be based on an assumed water surface of 645.19. The profile table is optional and can be left blank. However, it can produce a better

response, at least for some data sets. That being said, it should also be noted that the table will not perform as well when the flow at the dam is being heavily influenced by tailwater conditions.

## Hinge Point and Minimum Pool Operations

The hinge point navigation dam operation can also be combined with limits on the water surface at the dam. Hinge point and minimum pool operation will try to maintain the water surface within targets at the hinge point, but only when the water surface at the dam is above certain limits. When the water surface at the dam drops too low, the program will adjust the gates based on the water surface at the dam, essentially reverting to pool only control.

The hinge point and the minimum pool operation are each treated as separate control points. In addition to the water surfaces and Flow Factors for the Hinge control, the pool minimum has its own full set of water surfaces and

Flow Factors as shown in Figure 14-33 (these are accessed by clicking on the Min Pool Control button). Even though the minimum pool control is only trying to maintain a minimum water surface at the dam, a full range of water surfaces and Flow Factors are needed. These include the "too high" numbers such as Maximum High and Flow Open River. This allows the program to smoothly transition between hinge control and pool control. It also allows the pool control response to be fully calibrated between sluggish and overly sensitive transitions.

For hinge and minimum pool navigation dams, the program independently determines a desired flow for each control point (that is, the hinge and the pool minimum). It will then take the lower of the two flows and use that for determining the gate settings.

For example, assume the flows at the monitor location and the hinge point are 40,000cfs and that the water surfaces at the hinge point and the dam are 645.6 and 644.9 respectively. Based on the hinge point conditions (water surface at hinge point, Targets and Flow Factors for the hinge point), the program might compute a desired flow of 41,000 cfs. Next, the program will look at the conditions, targets, and Flow Factors at the dam and compute a desired flow of, perhaps, 42,000 cfs. Since the desired flow for the hinge point targets is lower than the desired flow for the navigation dam targets, the pool minimum is not a limiting factor. The program will adjust the gate settings to get 41,000 cfs and the navigation dam is operating under hinge control.

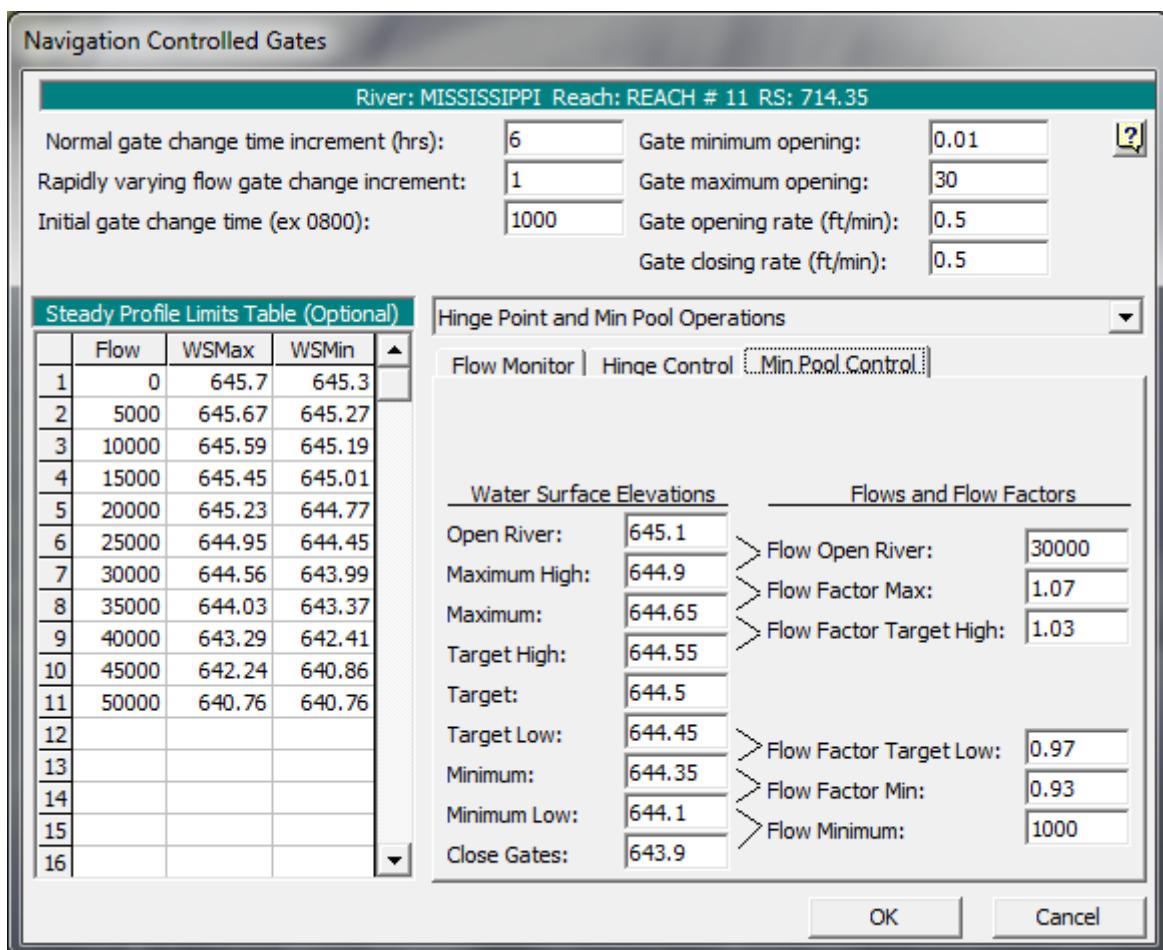


Figure 14 33. Navigation Editor with Hinge Point and Minimum Pool Operations and Control

The next time the gates are adjusted, assume the flow at the monitor and hinge point are still basically 40,000 cfs, but that the water surfaces have dropped to 645.5 feet at the hinge and 644.4 feet at the dam. The new computed flows might be 40,000 cfs at the hinge and 39,000 at the dam. In this case the program would use the 39,000 cfs figure and the dam would be under pool minimum control. In other words, the water surface at the dam has dropped to the point that the program has to operate the gates to maintain a minimum water surface at the dam regardless of what is happening at the hinge point.

The hinge and pool minimum operation is usually under hinge control for low and normal flows. At high flows the water surface at the dam must be lowered in order to keep the hinge point within the target range. At even higher flows, the water surface at the dam cannot be lowered far enough to keep the hinge point in range, thus the dam reverts to pool minimum control. Ideally, the pool would be kept at the specified absolute minimum (perhaps 644.1 feet in the above example) until the hinge point dropped back down into the target range. This is not possible without continuous adjustments of the gates, which is not practicable.

Instead, the water surface at the dam will fluctuate slightly even when it is operating under pool minimum control (just like it would fluctuate for pool only control). This is reflected in the range of target water surfaces for pool minimum control. The spacing of the target water surfaces has to be determined by trial and error. For example, if the water surface Target, Target High, and so on, are set to relatively high elevations (compared to the desired value), then the water surface at the dam might stay significantly above the minimum of 644.1. This is not desired when the water surface at

the hinge point is above the targeted range. Moving the dam target water surfaces closer together (closer to 644.1) will cause the program to increase the flows more quickly in order to drive the water surface back down. However, this can also cause the program to overshoot the desired target leading to frequent gate changes and/or bouncing water surfaces.

If the pool minimum is a hard minimum (a hard minimum might be, the pool should not be allowed to drop below 644.1 feet), then this minimum should be coded as one of the lower target water surfaces. For example, if 644.1 is the operationally prescribed absolute minimum and the user coded the primary water surface Target as 644.1, then the pool would fluctuate around the value of 644.1 during pool control. It would be better, in this case, to code it to the Minimum Low. On the other hand, if the minimum is a "soft" minimum (a soft minimum might be 644.45 +/- .25 feet) then setting Target Low or even perhaps the primary Target to 644.45 might give better results. As already mentioned, the user should be prepared to take a trial and error approach in order to get the best results.

For hinge point and minimum pool operation, the Steady Profile Limits table can still be optionally used. This table is only used when the dam is operating under hinge control. The water surface values in the table can be lower elevations than the actual limits on the pool. These values are still used, but the pool control minimum will still apply. For example, the values in the table go below the 644.1 desired minimum at the pool. During rapidly changing conditions, when the water surface for a given flow diverges from the steady state water surface (for that flow), these lower values can still be used and will (in some cases) give a faster response. However, if the water surface actually drops down to around the 644 to 645 level, the flow based on pool control will eventually be lower than that based on the Hinge/Steady Profile table and the dam will revert to pool control (which, again, does not use the tables).

## **Hinge Point and Minimum and Maximum Pool Control**

The final type of navigation dam operations is combining hinge point control with both a minimum and maximum limit on the water surface at the dam. This editor has a third button as shown in the figures below.

The minimum and maximum pool controls are treated as separate control points even though they are both located immediately upstream of the dam. They each have a full set of target water surfaces and Flow Factors. The program will compute a desired flow for each control point. Therefore, there will be a flow based on the hinge point targets, a flow based on the minimum pool elevation, and a flow based on the maximum pool elevation. During normal operations, the flow will be based on the hinge point target. However, the desired flow will not be allowed to go below the minimum pool control flow and it will not be allowed to go above the maximum pool control flow.

Having separate control points for the minimum and maximum control allows a smooth transition between pool control (either high or low) and hinge control for a full range of flows. It also provides the greatest control and sensitivity for allowing the water surface at the pool to be maintained within the tightest tolerances.

The optional steady profile limits table may still be used. As before, it only applies to hinge control.

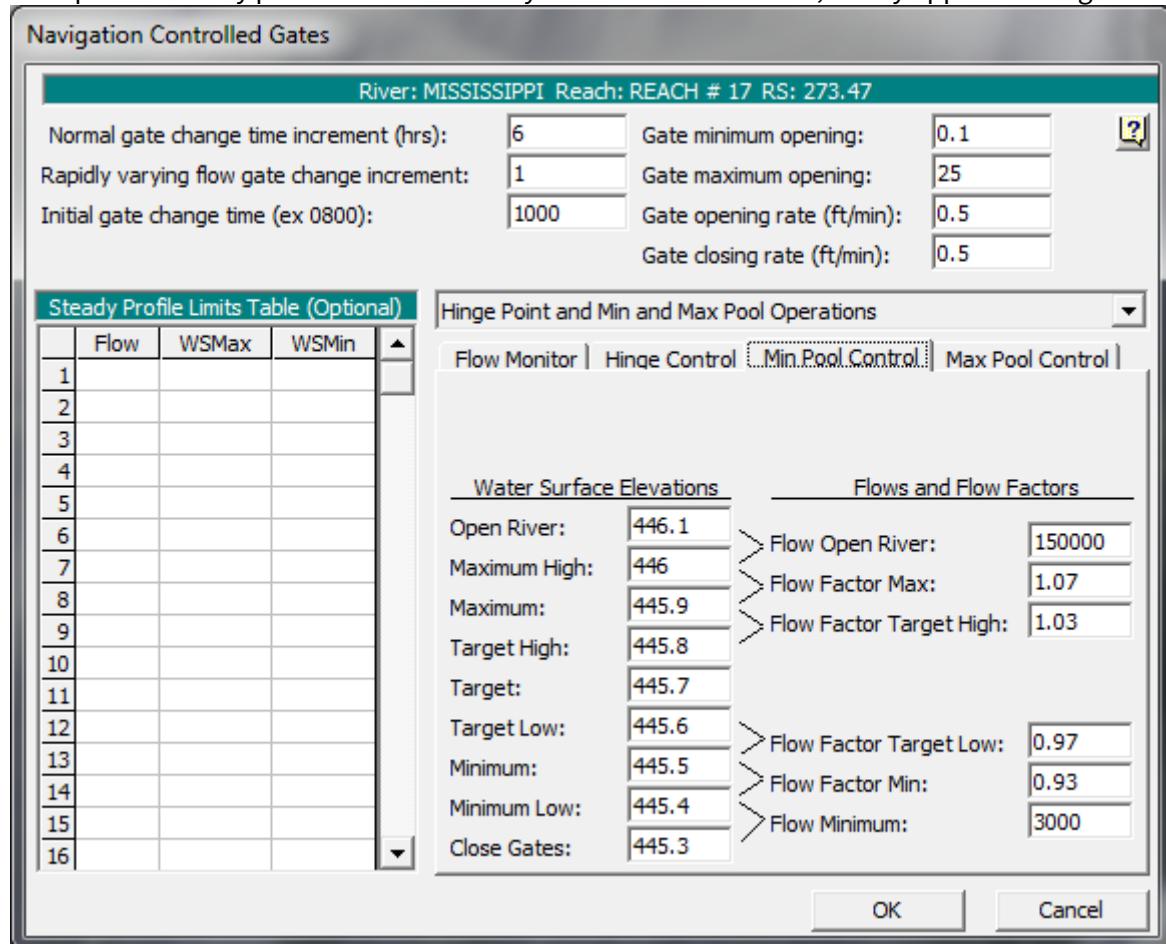


Figure 14 34. Navigation Editor with Hinge and Maximum and Minimum Control (Min Pool Control Shown)

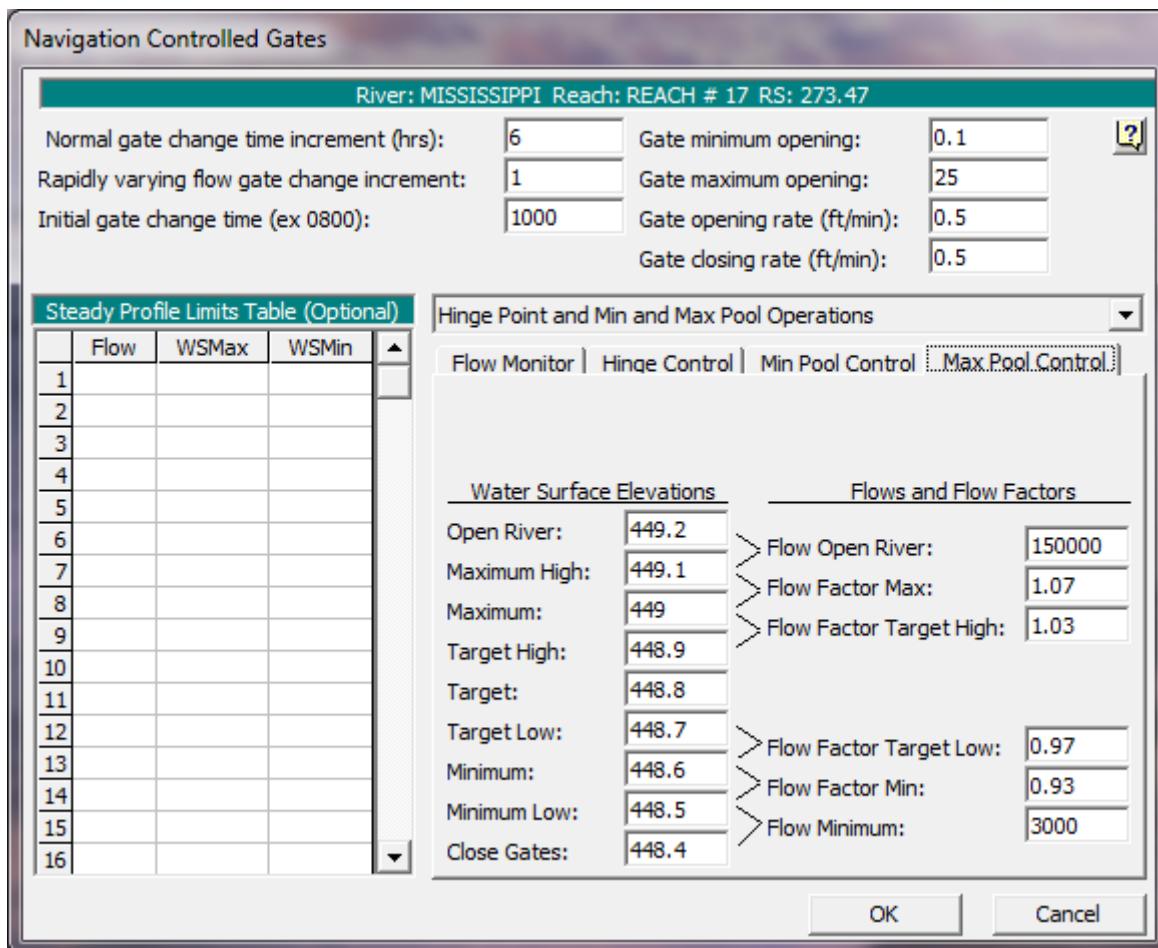


Figure 14 35. Navigation Editor with Hinge and Minimum and Maximum Control (Max Pool Control Shown)

## Modeling Pressurized Pipe Flow

HEC-RAS can be used to model pressurized pipe flow during unsteady flow calculations. This is accomplished by using the Preissmann slot theory applied to the open channel flow equations. To model pressure flow with HEC-RAS, the user must use cross sections with a **Lid** option. The cross section is entered as the bottom half of the pipe and the Lid is entered as the top half of the pipe. Any shape pipe can be modeled, however, the details of the pipe shape will depend on how many points the user puts in for the bottom (cross section) and the top (Lid). An example of adding a lid to a cross section is shown in Figure 14-36.

**NOTE:** This option only works with the 1D Finite Difference solution scheme. This option does not work with the 1D Finite Volume solution scheme.

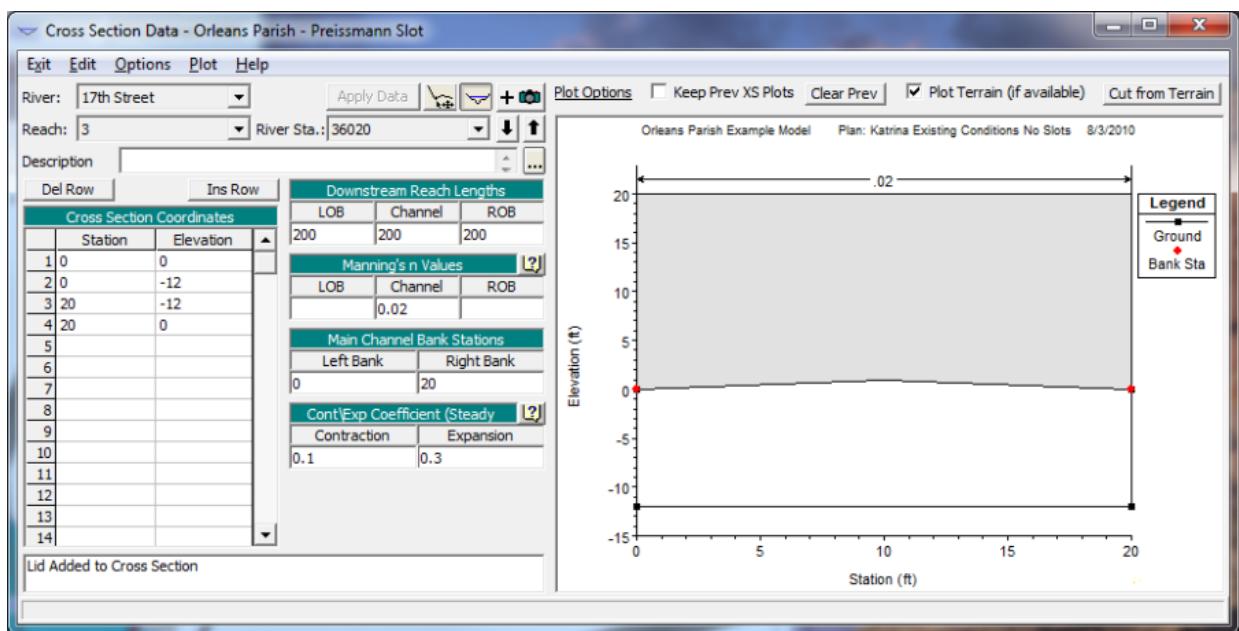


Figure 14 36. Cross Section with the Lid Option entered.

To enter a Lid at a cross section, select **Add a Lid to XS** from the **Options** menu on the Cross Section editor. When this option is selected, a window will appear as shown in Figure 14-37.

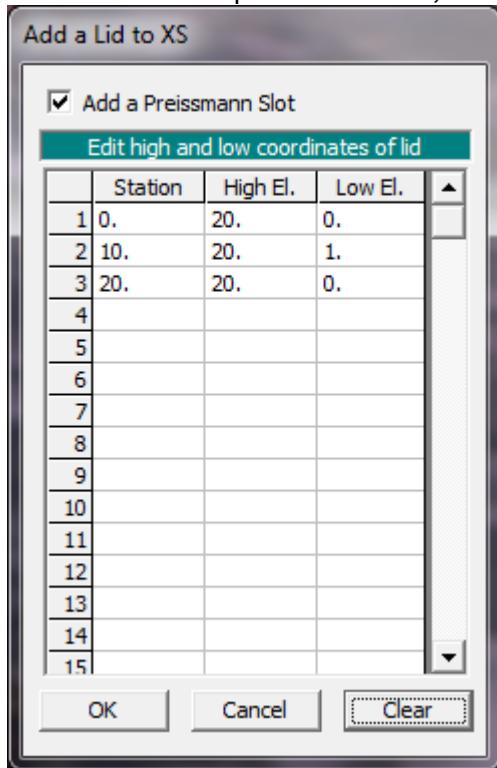


Figure 14 37. Cross Section Lid Editor

Additionally, the user must instruct the program to use the **Preissmann Slot** option for that particular cross section. The Preissmann Slot option can be turned on for an individual cross section from the Cross Section Lid Editor by checking the box at the top of the editor. The user can also bring

up a table that will show all of the locations where cross sections with lids exist. This table can be viewed by selecting **Preissmann's Slot on Lidded XS's** from the **Tables** menu on the Geometric Data Editor. When this option is selected, a window will appear as shown in Figure 14-38.

Check for a Preissmann's Slot to be added to XS's with a lid					
	River	Reach	RS	Add Preissmann Slot	
1	17th Street	3	36020	<input checked="" type="checkbox"/>	
2	17th Street	3	35819.8*	<input checked="" type="checkbox"/>	
3	17th Street	3	35619.6*	<input checked="" type="checkbox"/>	
4	17th Street	3	35419.4*	<input checked="" type="checkbox"/>	
5	17th Street	3	35219.2*	<input checked="" type="checkbox"/>	
6	17th Street	3	35019.*	<input checked="" type="checkbox"/>	
7	17th Street	3	34818.8*	<input checked="" type="checkbox"/>	
8	17th Street	3	34618.6*	<input checked="" type="checkbox"/>	
9	17th Street	3	34418.4*	<input checked="" type="checkbox"/>	
10	17th Street	3	34218.2*	<input checked="" type="checkbox"/>	
11	17th Street	3	34018.*	<input checked="" type="checkbox"/>	
12	17th Street	3	33817.8*	<input checked="" type="checkbox"/>	
13	17th Street	3	33617.6*	<input checked="" type="checkbox"/>	
14	17th Street	3	33417.4*	<input checked="" type="checkbox"/>	
15	17th Street	3	33217.2*	<input checked="" type="checkbox"/>	
16	17th Street	3	33017.*	<input checked="" type="checkbox"/>	
17	17th Street	3	32816.8*	<input checked="" type="checkbox"/>	
18	17th Street	3	32616.6*	<input checked="" type="checkbox"/>	

Figure 14 38. Preissmann Slot Table for Cross Sections with Lid's.

The Preissmann Slot table will show all cross section locations that contain lids. The user can turn on or off the Preissmann slot option by simply checking the box next to the desired cross section location. All of the check boxes can be turned on or off simultaneously by clicking on the **Add Preissmann Slot** column heading at the top of the table.

In general, lids can be added to any cross section in the HEC-RAS model. Several cross sections in succession with lids can be used to represent a pipe. Multiple interconnected pipes can be modeled. Lidded cross sections can be used around stream junctions to represent pressurized junctions. However, HEC-RAS does not compute minor losses at junctions, bends, or where pipes change size. This is currently a limitation in modeling pressurized pipe flow with HEC-RAS. Lateral flows can be modeled by either using lateral structures with culverts, or by directly inputting hydrographs as lateral flow boundary conditions. The lateral structure option can be used to mimic drop inlets connecting the surface flow to the pipe. An example of a pressurized pipe with lateral structures connected to the surface is shown in Figure 14-39.

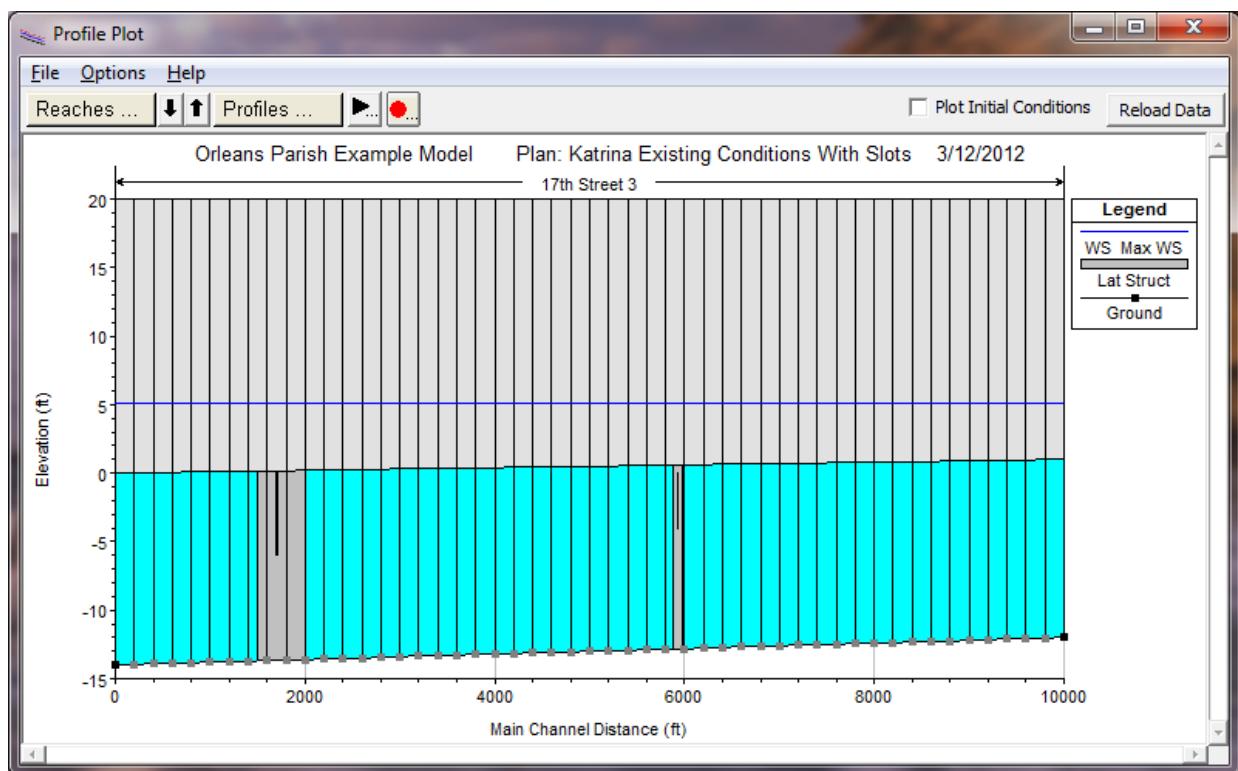


Figure 14 39. Example Pressurized Pipe modeled with Cross Sections and Lids.

For the computational details of how the Preissmann slot option works, please see the section on modeling pressurized pipes in the Hydraulic Reference Manual.

## User Defined Rules for Hydraulic Structures and Pumps

The operating procedures for determining and controlling the releases from reservoirs and other types of hydraulic structures can be quite complex. HEC-RAS allows flexibility in modeling and controlling the operations of hydraulic structures through the use of rules (Figure 14-40).

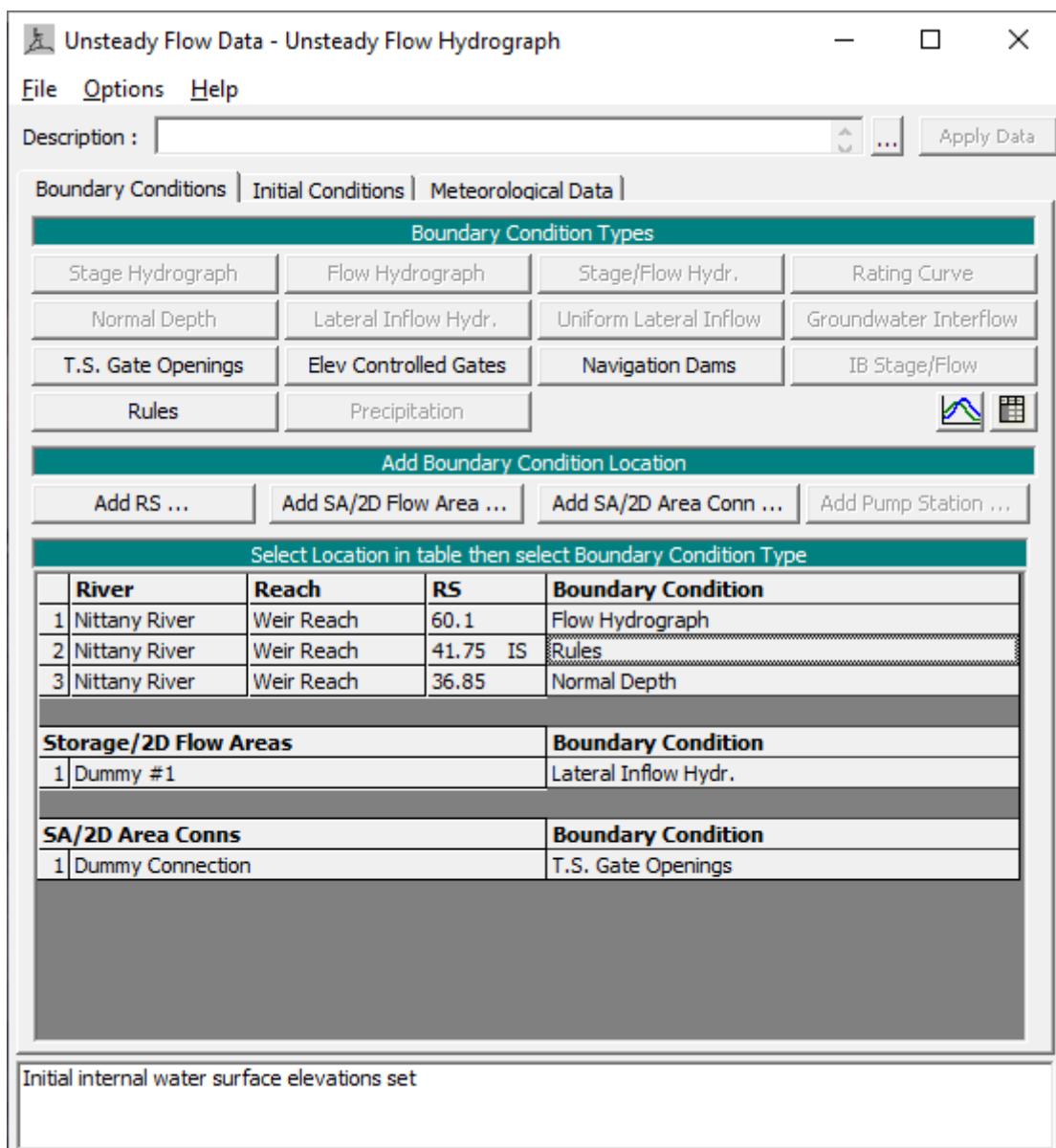


Figure 14 40. Selecting Rules from the Unsteady Flow Data Editor

The rules can be used to operate the height of the gate openings. Alternately, the rules can directly control (or constrain) the flow despite the gate openings (or even without gates at all). Examples of variables that could be used to control releases from a hydraulic structure are: current flows and water surfaces at the structure, current flows and stages at downstream or upstream cross section locations, time considerations (winter, morning, etc), and/or previously computed values (accumulated outflows, running averages, etc). These variables can be combined with math operations and conditional operations to produce sophisticated controls. Rule operations in HEC-RAS are available for inline hydraulic structures, lateral hydraulic structures, and storage area connections.

The rules can also be used at pumping stations. The basic operation for pumping stations, such as the water surface at which a pump turns on, is entered on the geometry editor. Rules can be used to

modify these operations, for instance by setting a maximum pump flow. Or the rules can entirely replace the geometry based operations by specifying when and how the pumps turn on and off.



Sorry, the widget is not supported in this export.  
But you can reach it using the following URL:  
<http://youtube.com/watch?v=KMbV-cexP7w>

## Entering Rule Operations

Rules for controlling hydraulic structures can be entered after an inline structure, lateral structure, or storage area connection has been added to the project. From the Unsteady Flow Data editor, add or select the given structure and then click on the **Rules** button (figure 14-40). This will bring up the Rule Operations editor as shown in figure 14-41. In the Gate Parameters table near the top of the editor, some initial information can be entered for any gate groups that are in the hydraulic structure.

Gate Parameters					
Location	Open Rate (ft/min)	Close Rate (ft/min)	Max Opening	Min Opening	Initial Opening
1 Drop Gates	0.1	0.1	4		2

Summary of Variable Initializations:		
User Variable	Description	Initial Value
1		

Rule Operations	
row	Operation
1	'Gate current opening' = Inline Structures:Gate.Opening(Nittany River,Weir Reach,41.75,Drop Gates,Value at current time step)
2	'Gate current flow' = Inline Structures:Gate.Flow(Nittany River,Weir Reach,41.75,Drop Gates,Value at current time step)
- 3	If ('Gate current flow' < 500) Then
4	Gate.Opening = 'Gate current opening' + 0.1
5	Elseif ('Gate current flow' > 750) Then
6	Gate.Opening = 'Gate current opening'-0.1
7	End If

Figure 14 41. Gate Rule Operations Editor

The Open and Close Rate controls how fast the gates can move. So if, for example, a rule operation required the gate to open one additional foot and the gate opening rate was 0.1 ft/min and the user had selected a one minute time step, it would take ten time steps for the gate to reach the new opening height. The Open and/or Close Rate can be left blank, which means the gate can move to any new setting in a single time step.

The Max and Min Opening will constrain the maximum and minimum gate opening settings. Building on the above example of opening the gate one additional foot, if the gate was at 3.5 feet and the maximum was set to 4 feet (even though the gate was 6 feet tall), over a five minute period, the gate would open to 4 feet and then stop. If the Max is left blank, then the gate maximum opening is limited only by the height of the gate. If the Min is left blank, then the minimum opening is fully closed (i.e. 0.0).

The Initial Opening provides the first setting for the gate. This opening height will be used during the initial backwater computation. The gate will be left at this setting until it is changed by a rule operation. The Initial Opening is required for all gate groups, if any, in the hydraulic structure and may not be left blank.

At the top of the editor, the user has the option of entering a description of the rule set. This can be a useful tool for documentation especially if the user has multiple plans with different rule operations.

### Rule Sets

A group of rules for one hydraulic structure is referred to as a rule set. At the start of each time step, each rule set is evaluated to check for changes to the operation of the given hydraulic structure. Rule operations are performed from the first (top) rule to the last (bottom) rule. By default, each rule operation is evaluated once. However, branching operations (If/Then/Else, etc) can cause some rule operations to be skipped. No looping or jumping to prior rule operations is allowed. That is (during a given time step), a rule operation may not be performed more than once.

Note: A rule set is only called once during a time step, even if the program iterates during that given time step. (Whatever rules are "in force" at the start of the time step will apply during all of the iterations). At some point, a user option may be added to RAS to let the rule set be called for every iteration. This would allow the rule set to use a more "current" water surface and/or respond to stability problems.

When the Rule Operations editor is opened, the rules for that hydraulic structure are displayed beneath the Rule Operations heading. The rule set shown in figure 14-41 has 7 types of operations.

In this example, operation #1 gets the current gate opening. Operation #2 gets the current flow going through the gate. Operation #3 checks if the flow is less than 500 cfs. If it is, then operation #4 sets the gate opening to the current opening + 0.1 feet. After operation #4, control would jump to after the End If (operation #7). However, since there are no more operations after the End If, the rule set would be done for this time step.

If the flow is greater than or equal to 500 cfs, then operation #3 is false. In this case, control would jump to operation #5. Operation #5 checks if the flow is greater than 750 cfs. If it is, then operation #6 will close the gate 0.1 feet. In either case, the rule set would again be finished for this time step.

### Operation Rules

To add, delete, or edit rule operations, click on the **Enter/Edit Rule Operations...** button at the bottom of the Rule Operations editor. This will bring up the Operation Rules editor as shown in figure 14-42.

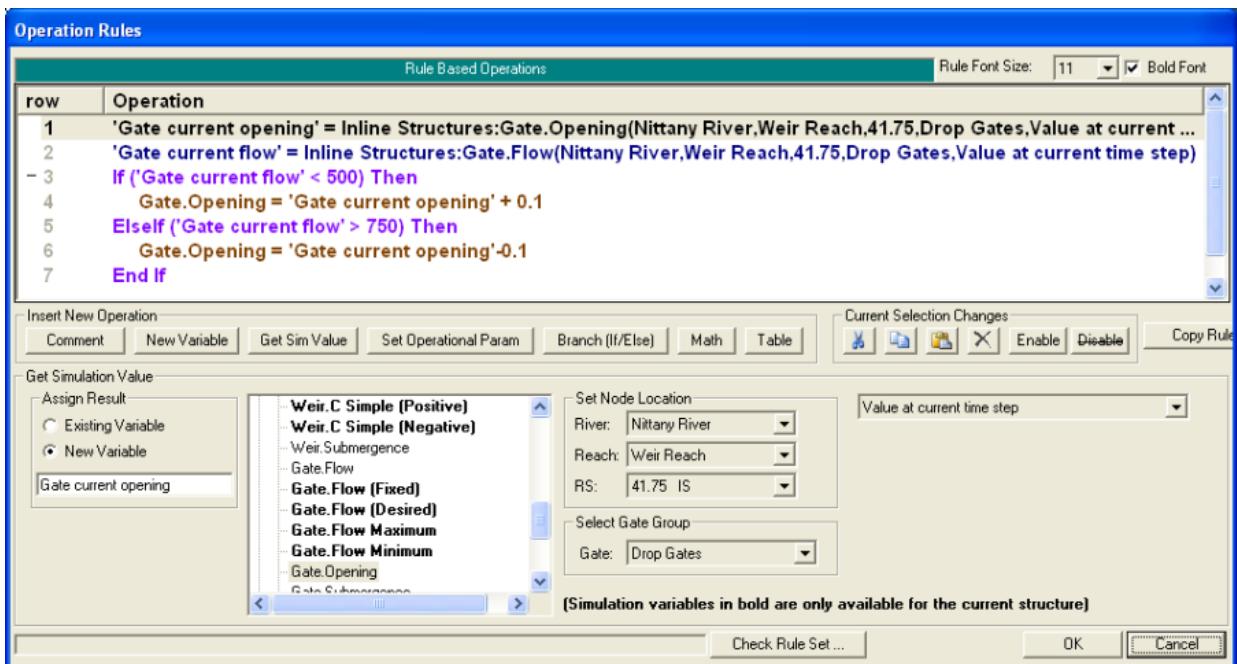


Figure 14 42. Operation Rules Editor

Seven different types of operations can be added by clicking one of the buttons under the Insert New Operation field. A brief overview is given immediately below and this is followed by a detailed description of each rule operation type.

## Operation Types

- **Comment.** Provides a user entered line of text (for documentation only).
- **New Variable.** Allows the user to create a variable and give it a custom name.
- **Get Simulation Value.** A variable is set equal to a given value in the simulation, such as the flow at a cross section or the time of day.
- **Set Operation Parameter.** Changes the operation of the hydraulic structure, for example, adjusting the gate height or setting a maximum discharge.
- **Branch (If/Else).** Controls which operations are executed on the basis of an If-Then test (e.g., do different gate operation checks based on seasonal considerations).
- **Math.** Performs math operations such as summing flows or averaging water surfaces.
- **Table.** This operation allows the user to enter a table and perform table lookups to get a value.

### Comment

Clicking the **Comment** button allows the user to enter a line of text. This "operation" is not used during the computations. Rather, it is intended to make the rule set operating procedure easier to understand by allowing the user to document the rules inside of the rule set (see figure 14-43). Note: because RAS uses a comma as an internal delimiter, it will not allow a "," to be part of a comment line.

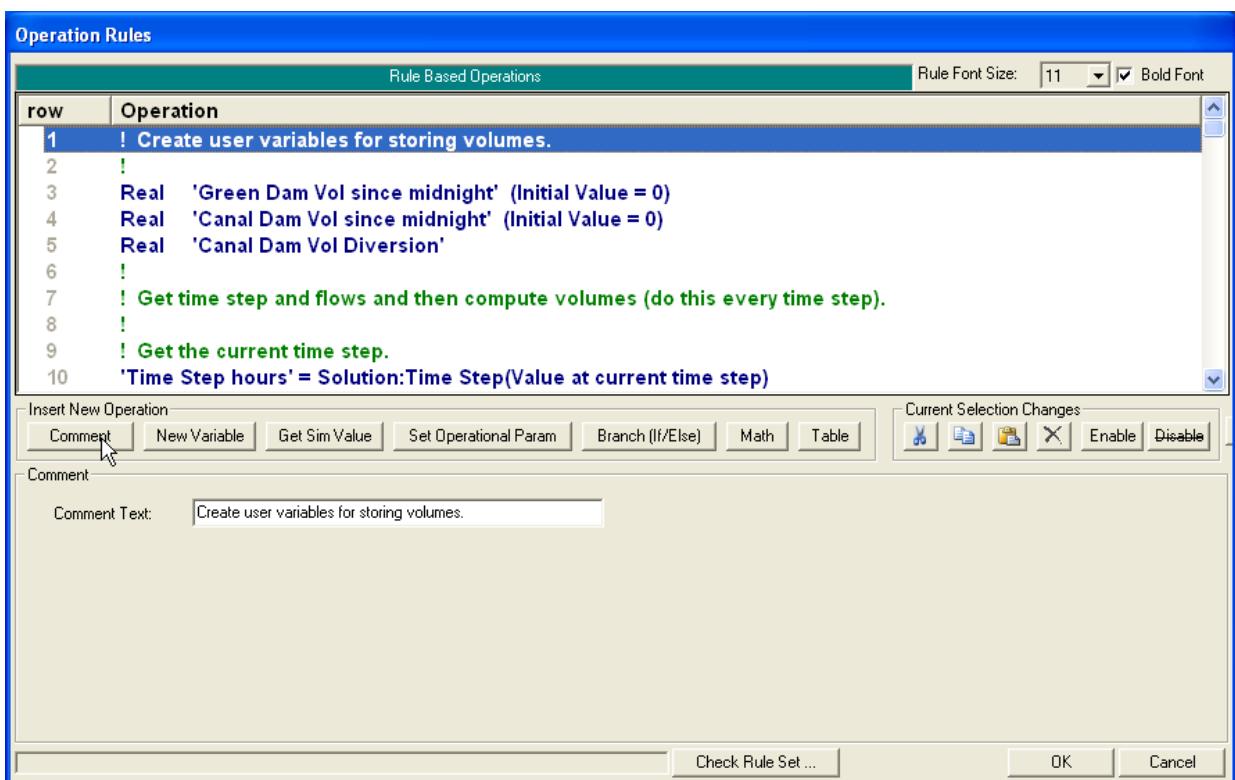


Figure 14 43. Operation Rules Editor with comment line shown

### New Variable

The **New Variable** button brings up the editor as shown in figure 14-44. The name of the variable must be entered in the User Variable Name field. The name must be unique. That is, it can't be the same as any other variable name in the given rule set. A duplicate name will cause a run time check error, as discussed below.

By default, the variable type is real (which includes fractional numbers such as 11.35). The alternative type is integer (counting numbers such as -2, 0, 1, 5, 10, etc). If the user selects integer, the value of the variable will always be an integer. So if the current value of a user integer is 4 and a math operation (see below) adds 1.7 to it, the final value will be rounded to the nearest integer (in this case 6).

The user may enter an initial value for the variable (by default the value is zero). *The variable is only initialized to this value at the start of the simulation.* (To initialize a variable every timestep, use a Math operation.) It will equal this value until (or unless) it is changed by another rule. For example, if the user variable, "Test Case" has an initial value of 3 and at the start of the fourth time step it is changed (by another rule) to a value of 6. At the fifth time step, it will equal 6 (it is not "reinitialized" to 3) and will continue to equal 6 until/unless it is changed again.

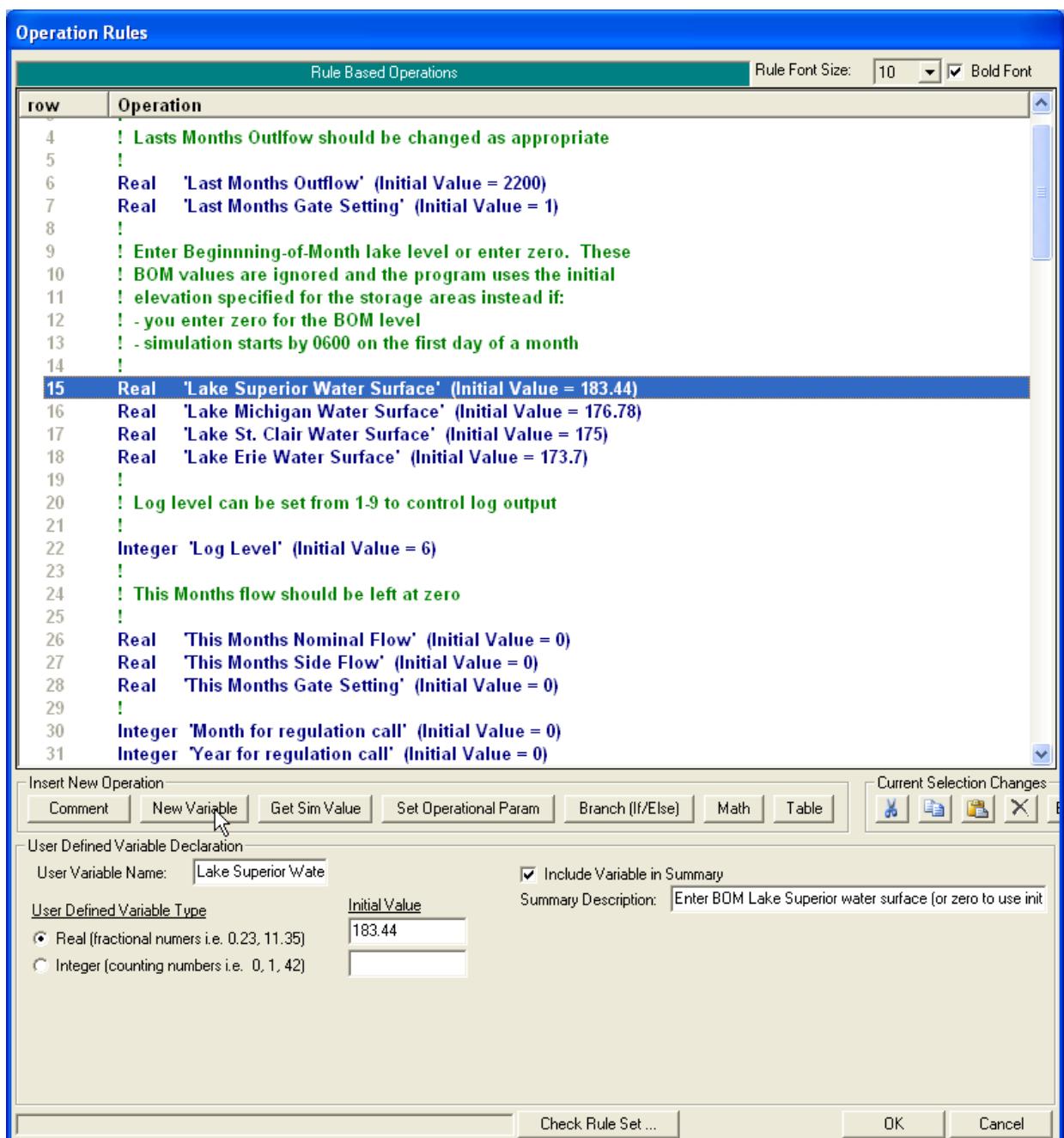


Figure 14 44. Operation Rules Editor with New Variable operation shown

If the user checks the optional "Include Variable in Summary," then the variable will be listed on the main Rule Operation editor as shown in figure 14-45. The initial value can then be entered or changed directly on the Rule Operations editor.

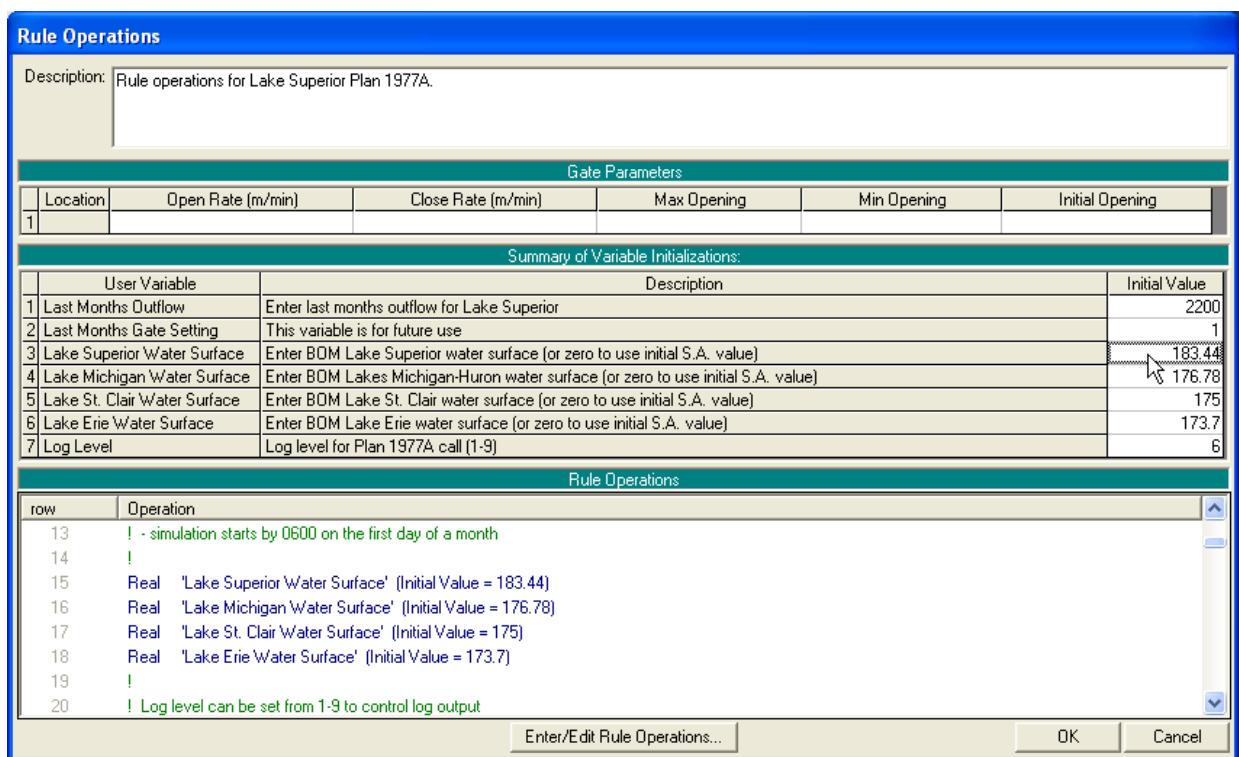


Figure 14 45. Rule Operations Editor with Summary of Variable Initializations

### Get Simulation Value

The **Get Simulation Value** operation provides information about the current state of the model. In the example shown in figure 14-46, the operation is getting the day of the month at the beginning of the time step and putting it into a new variable called "Day Beg time step." For example, if the simulation time window went from 01Jan2000 to 03Jan2000 and the run was about halfway through, the "Day Beg time step" variable would be set to 2.

This is another way to create a "new variable" (a variable does not have to be created with the **New Variable** button). However, variables created in this manner cannot be integers (they may only be real types), they cannot be assigned an initial value (or rather, the initial value is always zero), and they cannot be included in the Variable Summary.

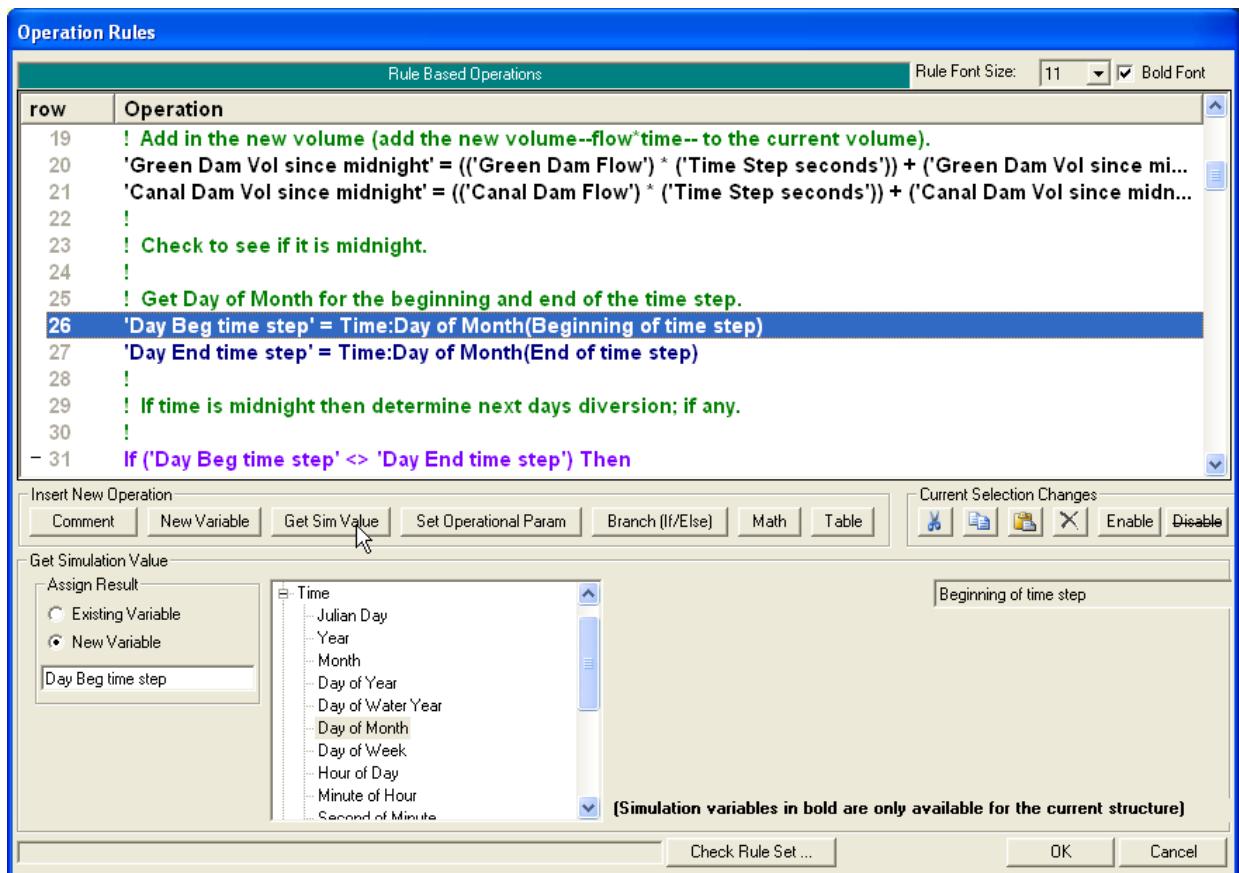


Figure 14 46. Operation Rules Editor with Get Simulation operation shown

If the user selected to change the Assign Result to Existing Variable, then a drop down menu would appear as shown in figure 14-47. Selecting one of the previously defined variables would put the result (day of the month in this example) into that variable instead of creating a new one.

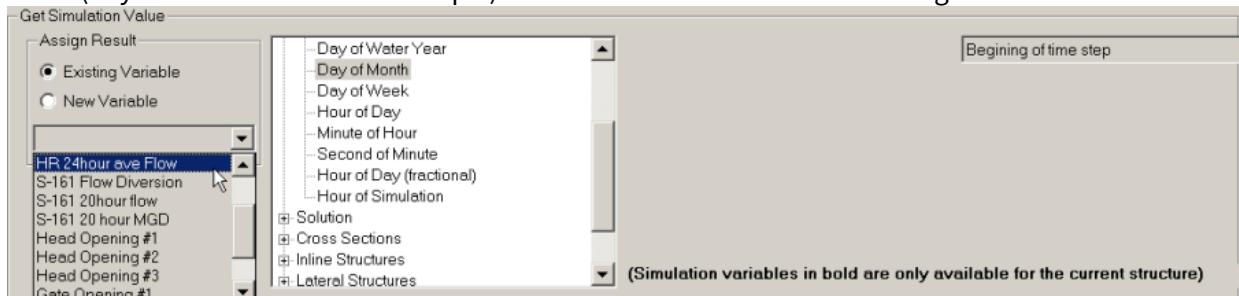


Figure 14 47. Get Simulation Value assigning to an existing variable

**Note:** on renaming New/Existing variables. A new variable (whether it is from a user variable, get simulation, math, or table operation) can be renamed by typing in a new name. **However, any references to that variable will not be automatically renamed!** A reference to a non-existent variable will result in a run time check error. The user will have to manually change all references to that variable (whether on the Assign Result to an Existing Variable or using an Existing Variable in an Expression, see below). This is also covered in the discussion of **Check Rule Set**, below.

There are currently eight categories of simulation variables (more may be added later). These are Time, Solution, Cross Section, Inline Structure, Lateral Structure, Storage Areas, Storage Area Connectors, and Pump Stations. Clicking on the "+" will expand the list for that category. A complete list and definition of each variable is given at the end of this section of the manual.

For all the variables under Time, the user can select to use the time at the beginning of the time step (default), the end of the time step, or the previous time step. For example, assume the time step was 30 minutes long and the program had just finished the time step that ended at 12:15 (the program had just gone from 11:45 to 12:15 and was getting ready to go from 12:15 to 12:45). The minute of the hour at the beginning of time step would be 15. The hour of the day (fractional) for the beginning of the time step, end of time step, and previous time step would be 12.25, 12.75, and 11.75, respectively. The hour of the day at both the beginning and end of the time step would be 12. The previous hour of the day would be 11.

To get a water surface or flow at a normal cross section, expand the Cross Section list and highlight either the Flow or WS Elevation field as shown in figure 14-45. This will also display the standard node selector to allow the user to select the river, reach and river station for the desired cross section (also shown in figure 14-48).

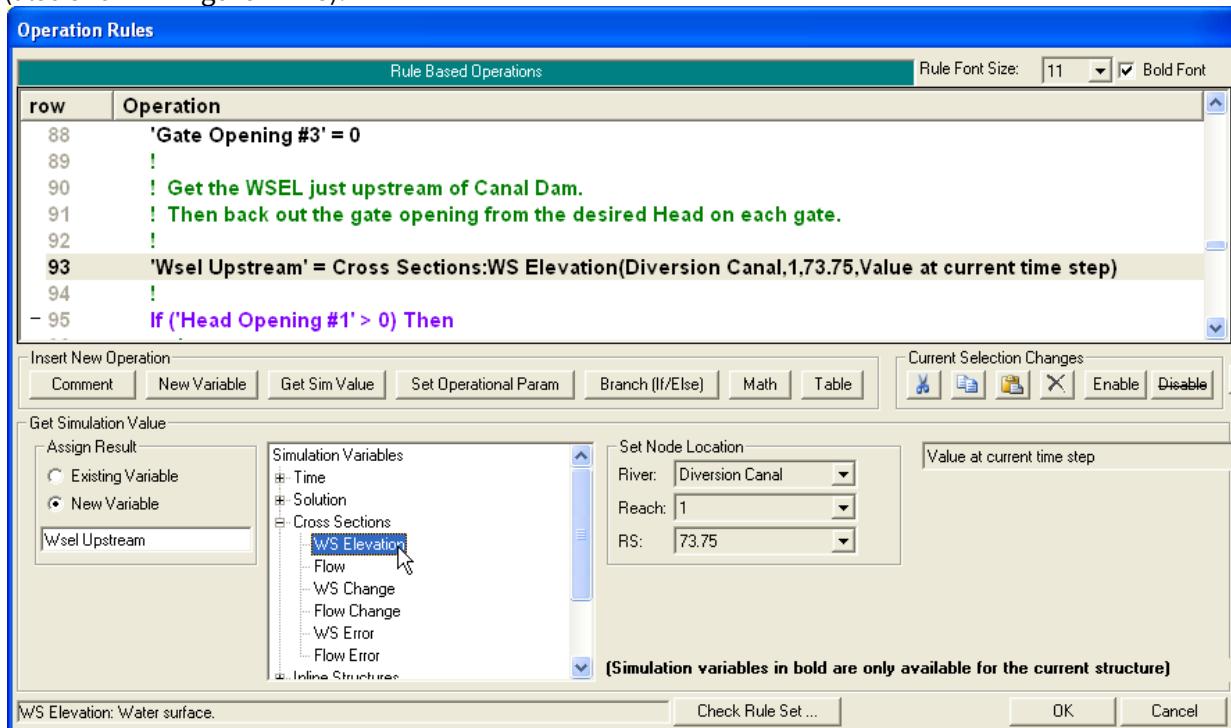


Figure 14 48. Selecting water surface at a cross section

At the far right is another drop down menu that allows additional choices for when and how the simulation value (water surface in this example) is computed, see figure 14-49. The default is the **Value at the current time step**. In the time example above, this would be the water surface at 12:15 for river station 73.75. Alternately, the user could select the **Value at previous time step**, which would be the water surface at 11:45 in the above example.

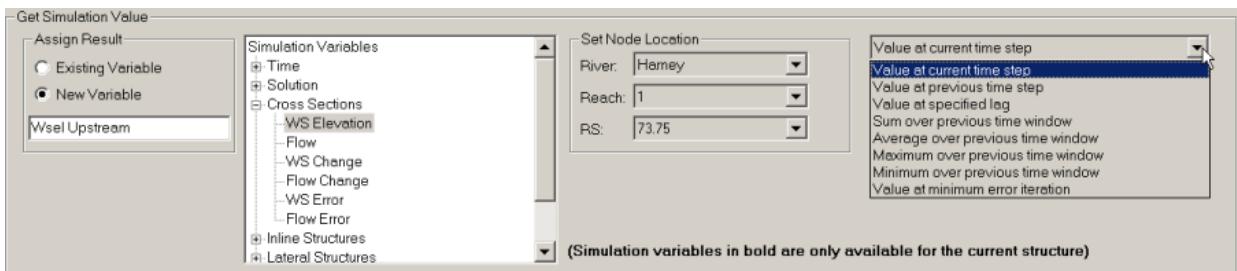


Figure 14 49. Changing how simulation values are computed

The remaining choices provide various options for the value further back in time. Selecting **Value at specified lag**, will display a window where the lag time in hours is specified. So continuing to build on the same example, a lag time of 1.5 hours would get the water surface at the given cross section at 10:45 (12:15 minus 1.5 hours equals 10:45). For the next four options, the user must specify a starting and ending lookback time. Selecting **Sum over previous time window**, the user could enter 1.5 for the starting time and 0 for the ending time (for the specified time window to end at the current time step, the ending time should be 0). So if the user selected flow, this would sum the flow for the previous 1.5 hours. In other words, it would return the volume of water passing the given node for the previous 1.5 hours (since the value takes into account the length of each time step, this is more technically an integral instead of a sum). Or the user could select water surface, select **Average over the previous time window**, enter 2 hours for starting time and 1 hour for ending time, see figure 14-50. This would return the average water surface at the cross section between 10:15 and 11:15. The next two options will return either the maximum or minimum value over the given user specified time window (e.g. the highest water surface).

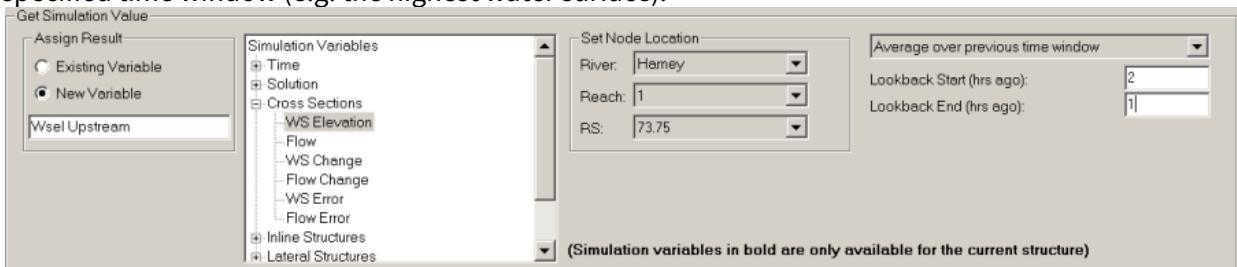


Figure 14 50. One hour average water surface starting two hours ago

Under the inline structure simulation variables as shown in figure 14-51, some of the choices are shown in bold font. The variables shown in bold are user settable operational parameters (Figure 14-52) for the current hydraulic structure (which happens to be an inline structure). These variables are also provided under the Get Simulation Value. This provides a way to check what an operational parameter has been set to (if it has been set at all). So, for instance, if a maximum flow had been set for the structure, then the **Structure – Flow Maximum** variable would return the value that this had been set to. The variables listed in bold are only available for the current inline structure—this is the hydraulic structure that this particular rule set is attached to.)

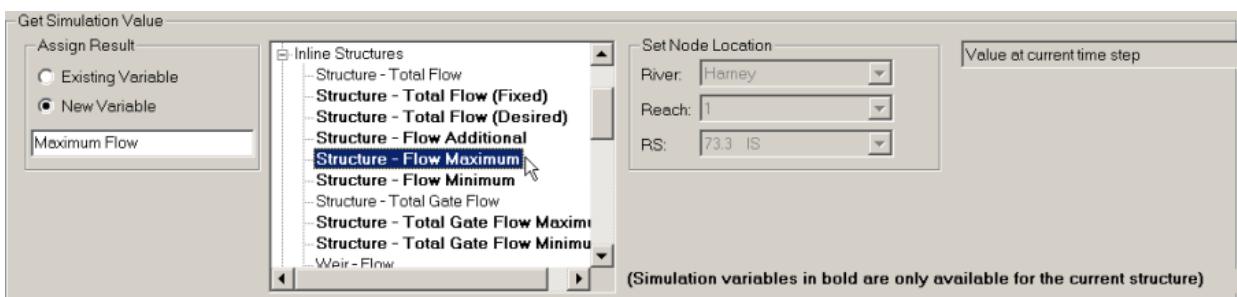


Figure 14 51. Getting an Operational Parameter

### Set Operational Parameter (for hydraulic structures)

Clicking the **Set Operational Param** button brings up the editor that allows a change to be made to the hydraulic structure operations (i.e. adjusting a gate opening). Changes can only be made for the hydraulic structure or pumping station that the rule set is attached to. This is the hydraulic structure (inline, lateral, or storage area connection) that was selected on the Unsteady Flow Data editor.

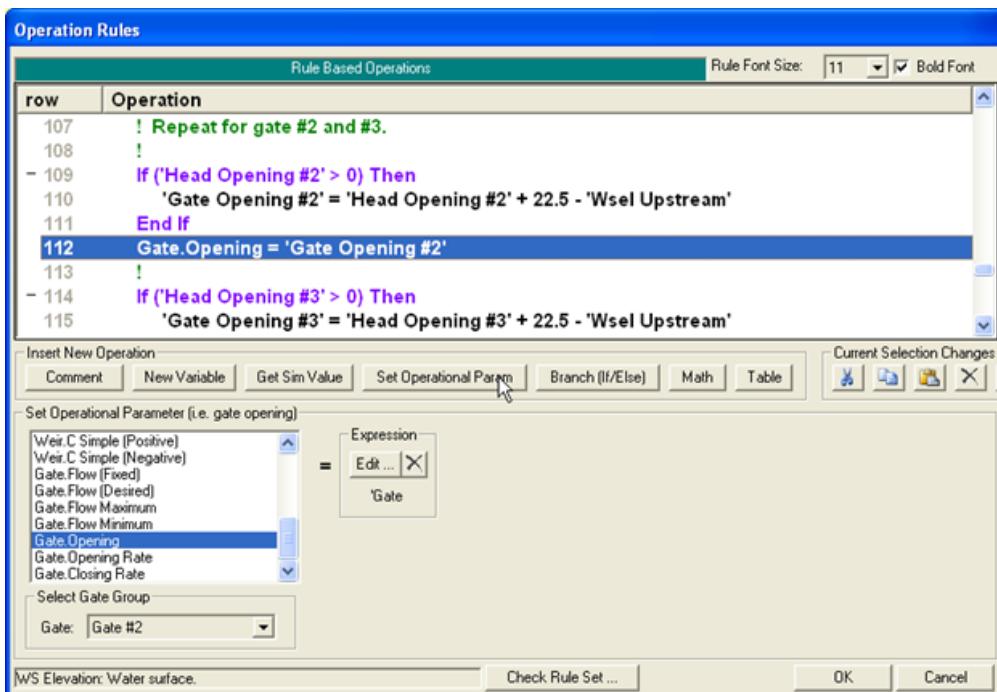


Figure 14-52. Setting an Operational Parameter

Figure 14-52 shows a change being made to the gate opening height (the gate will be moved to the new gate setting based on the Opening/Closing rate and maximum and minimum values, if any). If a gate variable is used, the appropriate gate group must be selected from the drop menu at the bottom. The new value is set equal to the value of the expression. (The expression might be a constant, such as the number “5”, a user variable as shown in Figure 14-52, or a simple math operation. See Math operations, below, for a detailed description of using expressions). The Opening Rate or Closing Rate changes the rate at which the gate can open or close. This overrides any value that the user may have entered on the Rule Operations editor.

**Gate - Flow (Fixed)** sets the gate flow to the given value. However, setting or changing the Flow (Fixed) value does not affect the gate opening height (the given amount of flow will be released through this gate group from the reservoir regardless of the gate opening height). The Fixed flow value will be used each time step until the Fixed flow is changed or removed. To remove the Gate - Flow (Fixed) parameter, set the Flow (Fixed) expression to “Not Set” as shown in figure 14-53 (see Math below for editing expressions).

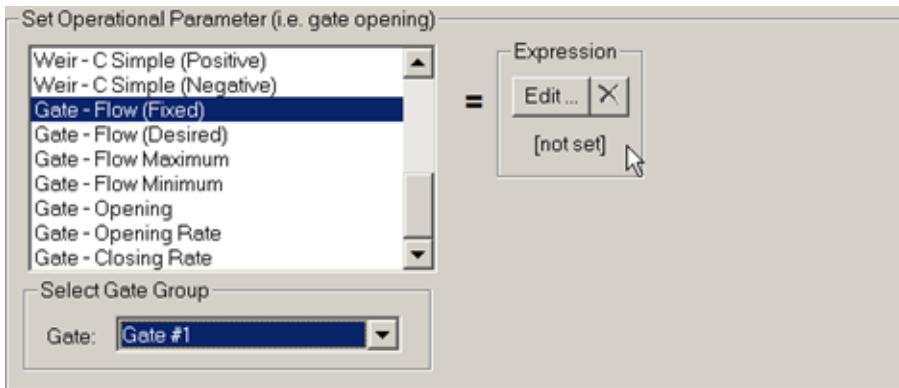
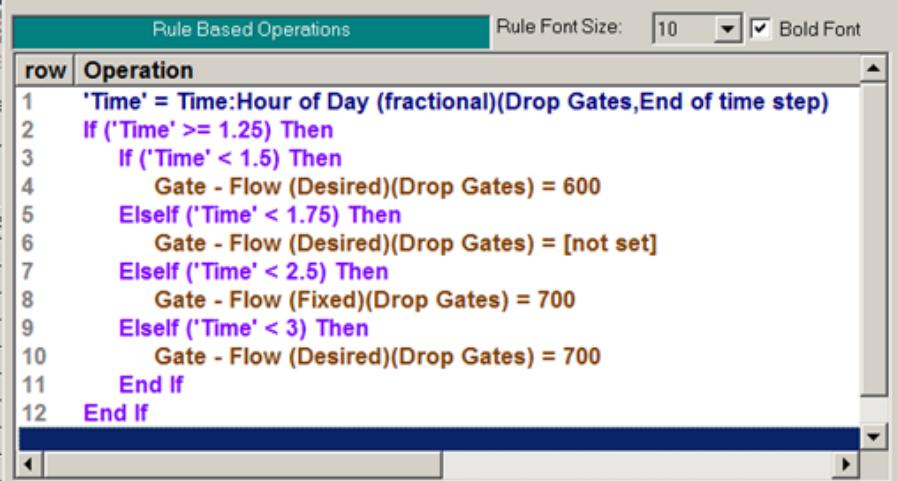


Figure 14-53. Turning off the Gate Fixed Flow

**Gate - Flow Maximum and Minimum** puts limits on the gate flow. For example, if the user sets the flow maximum to 1000 cfs, the program would first compute the flow through the given gate based on the gate opening and water surface(s), if the flow is below 1000, it would use that value. If the computed flow was larger, the program would restrict the flow to the user entered amount (1000 cfs). Note, however, that the Gate Maximum and Minimum flow will not override a Gate Fixed flow value. Gate Maximum and Minimum flow can be removed by using the “Not Set” value.

**Gate - Flow (Desired)** will cause the program to adjust the gate opening in order to give the given, desired, flow (based on the water surfaces and the gate characteristics) through the gate group. Once the gate opening is determined, the program will use this opening height to compute the actual flow. Since determining the gate opening is an inexact, iterative process, the actual computed flow may not perfectly match the desired flow. Note also that the program will not open/close the gates faster than the current Opening/Closing Rate, if any, allows. The program will adjust the gate opening each time step as long as the Gate Flow Desired has a value. This feature can be turned back off by using the “Not Set” value. Having the Desired value on will not prevent the final gate flow from being overwritten by a Maximum, Minimum, or Fixed flow. If the user wishes to force a given gate flow, but also wishes to know the [approximate] gate settings that would result in that flow, then this can be done by setting both the Fixed flow and the Desired flow to the same value (e.g. 3000 cfs).

The rule set shown in figure 14-54 illustrates how these gate features can be used and combined. The resulting output is shown in figure 14-55.



The screenshot shows a software interface titled "Rule Based Operations". At the top right, there are settings for "Rule Font Size" (set to 10) and a "Bold Font" checkbox. The main area is a table with two columns: "row" and "Operation". The "Operation" column contains a series of IF/THEN/ELSE statements in a programming-like syntax:

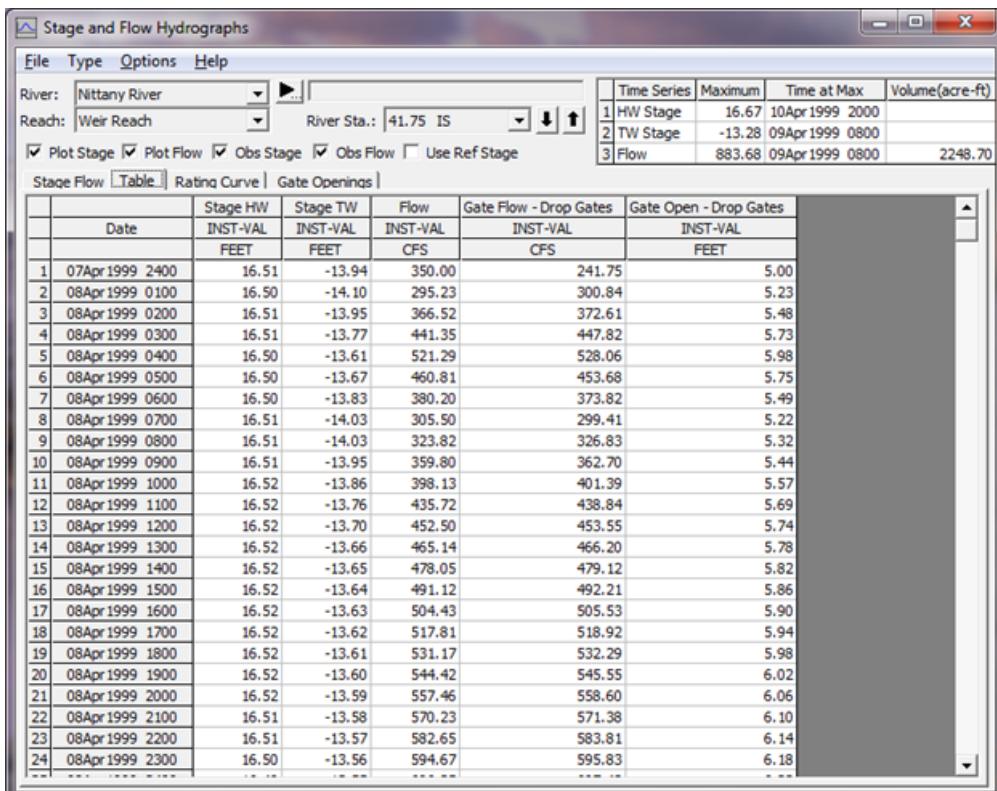
```

1  'Time' = Time:Hour of Day (fractional)(Drop Gates,End of time step)
2  If ('Time' >= 1.25) Then
3    If ('Time' < 1.5) Then
4      Gate - Flow (Desired)(Drop Gates) = 600
5    ElseIf ('Time' < 1.75) Then
6      Gate - Flow (Desired)(Drop Gates) = [not set]
7    ElseIf ('Time' < 2.5) Then
8      Gate - Flow (Fixed)(Drop Gates) = 700
9    ElseIf ('Time' < 3) Then
10      Gate - Flow (Desired)(Drop Gates) = 700
11  End If
12 End If

```

Figure 14-54. Desired and Fixed Flow Gate Operations

The initial gate opening is 6.0 feet. The If/Then test on row #2 is false until the time reaches 1:15 (this is 1.25 in fractional hours). At that point, row #4 is executed and the Desired Flow is set to 600 cfs. The gates open (at the user set opening rate) until the flow is approximately 600 cfs. After 1:30, the Desired Flow is turned off (“not set”). The gates then remain at that, current, gate height (6.1295 feet). At 1:45 row #8 is executed and the flow is fixed at 700 cfs (but the gate opening height is not changed). At 2:30, the Desired Flow is turned back on by setting it to 700 cfs (same as the fixed flow). This causes the gates to adjust. However, the actual release remains exactly 700 cfs because the Fixed Flow is still set to 700.



### Figure 14-55. Output from Desired and Fixed Flow Gate Operations

In addition to the gate control operations for individual gate groups, the user can set limits on all of the gates groups combined:

**Structure - Total Gate Flow** sets the flow for all of the gate groups. Instead of summing the flow from each gate group (and regardless of whether the gate group flow is “natural” or “fixed”), this flow is used instead.

**Structure - Total Gate Flow Maximum** sets a maximum flow for all of the gate groups. It will not override Structure - Total Gate Flow.

**Structure - Total Gate Flow Minimum** sets a minimum flow for all of the gate groups. It will not override Structure - Total Gate Flow.

The next category of structure operation parameters are for weirs:

**Weir - Flow** fixes the amount of flow over the weir.

**Weir - Flow Maximum** sets the maximum flow over the weir. It will not override Weir – Flow.

**Weir - Flow Minimum** sets the minimum flow over the weir. It will not override Weir – Flow.

**Weir - Weir Coefficient** sets the weir coefficient for the weir. (Tip: this allows a straight forward way to adjust the weir coefficient based on the depth and/or velocity of flow over the weir).

**Weir - Minimum Elev for Weir Flow** changes the minimum weir elevation that is required before the program will compute flow for the weir.

**Weir - C Simple (Positive)** sets the linear routing coefficient for positive flow (linear routing weirs only).

**Weir - C Simple (Negative)** sets the linear routing coefficient for negative flow (linear routing weirs only).

The final category of structure operation parameters are for the overall structure:

**Structure - Total Flow (Fixed)** forces the given flow for the inline structure. This flow is used regardless of the flow from the gates and/or weir.

**Structure - Flow Maximum** sets a maximum flow for the inline structure. It will not override the structure Fixed flow.

**Structure - Flow Minimum** sets a minimum flow for the inline structure. It will not override the structure Fixed flow.

**Structure - Flow Additional** will add in the additional given flow to the inline structure. It will not override the structure Maximum, Minimum, or Fixed flow.

**Structure - Total Flow (Desired)** computes gate settings to provide the total given flow for the inline structure. It works in a similar manner to Gate - Flow (Desired). However, it will open or close any/all of the gate groups to get the correct flow. (To increase flow, gate groups are opened in a left to right manner. To decrease flow, gate groups are closed from right to left.) Weir flow (and Flow Additional) is included in the desired flow (if the desired flow is 2000 cfs and the weir flow is 500 cfs,

the gates will be adjusted to get 1500 cfs of flow). Structure - Total Flow (Desired) will not override Structure - Total Flow (Fixed). However, it will still adjust the gate group settings.

**Structure - Stage (Fixed)** will force the given water surface immediately upstream of the inline structure. This option is very similar to the stage part of the Internal Boundary (IB) Stage and Flow Hydrograph option from the Unsteady Flow Data editor (see chapter 8). When the Unsteady Solver determines all of the flows and water surfaces at all the cross sections during the solution of a given time step, it will compute the amount of flow at the inline structure that is required in order to produce the given water surface. It is easy to generate instability and/or hydraulically unrealistic flows when using this option. For instance, a large increase in the fixed stage over a single time step could potentially cause very small or even negative flows through the inline structure and/or cause the model to go unstable. A large drop in the fixed stage may generate a flow that is physically greater than the structure is capable of passing. *To avoid these problems when changing the given water surface, it is recommended that the fixed stage is gradually adjusted over a period of time.* In the sample data sets that are included in the HEC-RAS installation, there is an example that illustrates this.

It should be noted that the Fixed Stage option is not compatible with the other set operational parameters. The flow through the inline structure is completely controlled by the requirement to generate the correct, given water surface. Therefore, any maximum, minimum, and/or fixed flow rules, that would otherwise be in effect, are ignored. Additionally, when the Fixed Stage rule is being used, HEC-RAS will automatically adjust the gates (and will also ignore any set operations for the gate openings). The gates are adjusted in order to approximate the gate opening that would be needed to accommodate the computed flow. Effectively, a Total Flow (Desired) rule is put into effect where the desired flow is set equal to the flow that is being forced through the structure by the Fixes Stage option. This will generally allow the Fixed Stage option to be turned off (using the “not set” feature) and normal gate flow to be resumed without causing instability.

#### Set Operational Parameter (for pumping stations)

Clicking the **Set Operational Param** button for pumping station brings up the editor shown below (figure 14-56).

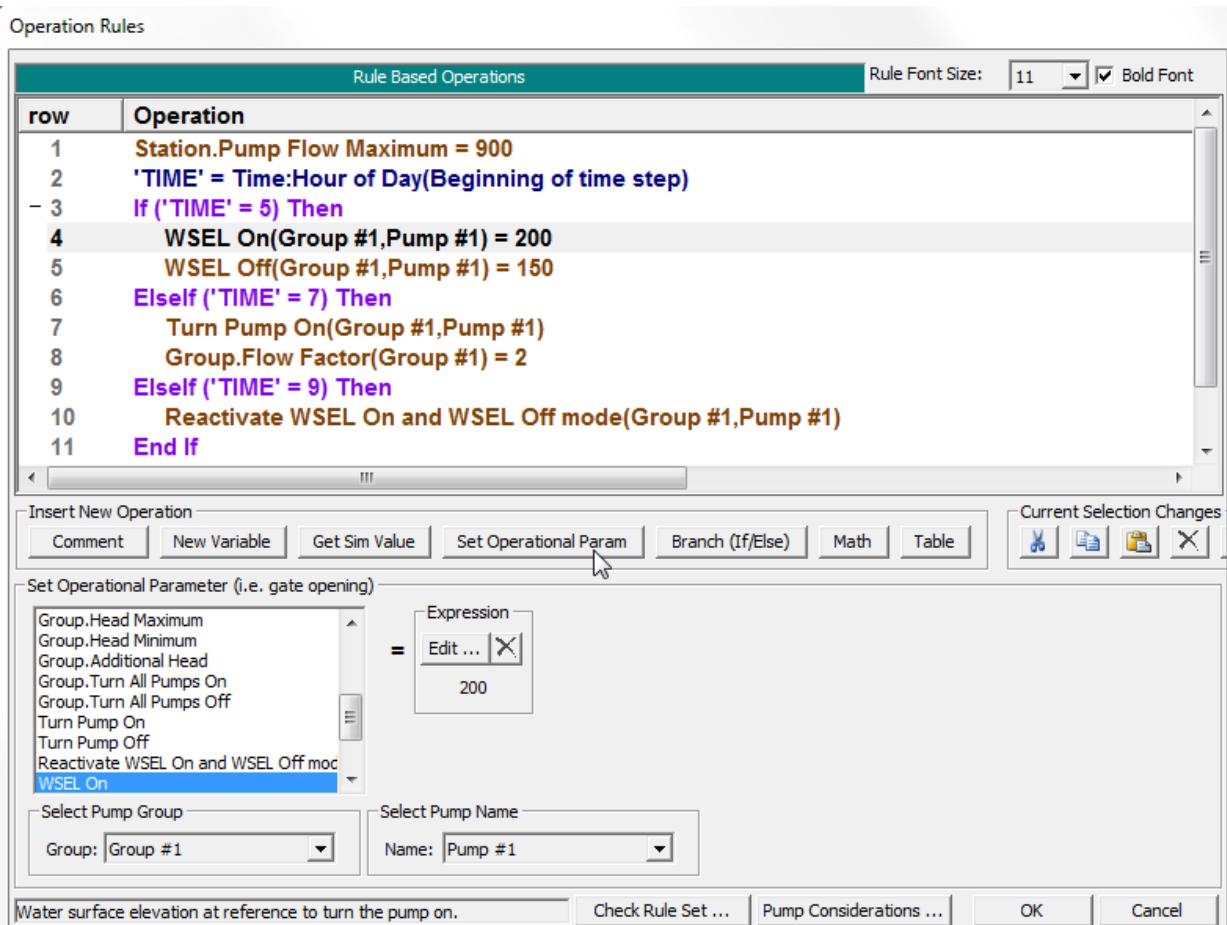


Figure 14 56. Setting an Operational Parameter for Pumping Stations

The operation of rules to control a pumping station is generally similar to the operation of rules to control a hydraulic structure, but, there is at least one notable difference. If a Rules Boundary Condition is selected for a hydraulic structure, then all of the control of that structure (e.g., gate openings) is handled by the Rules entered from the Operation Rules Editor.

However, this does not have to be the case for a pumping station. When a Rule Set is entered for a pumping station, the Rule Operation could be used in order to modify the control of the pumping station without entirely superseding the information that was entered as part of the geometry. For example, a pumping station may have a set of rules that determine a maximum amount that can be pumped, but the rules do not specify when the pumps should be turned on and off. This results in a hybrid control situation. Whether the pumps are pumping or not is determined from the data entered as part of the geometry. The amount of pump flow is also determined from the pump efficiency curve from geometry information, but this amount is not allowed to exceed the maximum set by the rules.

Alternately, the user can override the geometry control by the use of rule operations. But even after the rules have superseded the geometry, the rules can still "hand control" back to the geometry control. This brings up an important concept about the control of a pumping station.

When using rules, there are two fundamentally different modes of operation for turning individual pumps on and off.

Initially, the pumps are turned on and off based on the WSEL On and WSEL Off elevations that have

been entered in the Geometry Pump Station Data editor. This is referred to as the WSEL On and Off mode of operation. (That is, this mode is active.) The pump rules can be used to modify the water surface elevations (the water surface elevations entered from the geometry) that the pumps turn on and off. This does not, in itself, change the WSEL On and Off mode. If the mode was active, it stays active.

However, a pump can also be turned on or off by directly using a rule. Anytime a rule is used to turn a pump on or off, then the WSEL On and Off mode is automatically deactivated for that pump. When the WSEL On and Off mode is deactivated, the On/Off water surfaces elevations (that were entered as part of the geometry) will no longer control that specific pump. (The pump can continue to be turned on and off using rule operations, of course.) If it is desired to return control to the On/Off water surfaces on the geometry editor, then the WSEL On and Off mode must be made active. This is done by using the Reactivate WSEL On and Off mode rule (see below).

**Station – Pump Flow** sets the pumping station flow to the given value.

**Station – Pump Flow Maximum** sets a maximum flow for the pumping station.

**Station – Pump Flow Minimum** sets a minimum flow for the pumping station.

**Station – Turn All Pumps On** starts turning all the pumps on (this will make the WSEL On and Off mode inactivate for all pumps).

**Station – Turn All Pumps Off** starts turning all the pumps off (this will make the WSEL On and Off mode inactivate for all pumps).

**Group – Pumps Flow** sets the group pump flow to the given value.

**Group – Pumps Flow Maximum** sets the maximum group pump flow to the given value.

**Group – Startup Time** sets the time to ramp on the pumps in this group to the given value.

**Group – Shutdown Time** sets the time to ramp off the pumps in this group to the given value.

**Group – Flow Factor** sets a multiplier for the pump flow in this group to the given value.

**Group – Head Maximum** sets a maximum head this group will pump against.

**Group – Head Minimum** sets a minimum head this group will pump against.

**Group – Additional Head** sets an additional head this group will pump against.

**Group – Turn All Pumps On** starts turning all the pumps in this group on (this will make the WSEL On and Off mode inactivate for pumps in this group).

**Group – Turn All Pumps Off** starts turning all the pumps in this group off (this will make the WSEL On and Off mode inactivate for pumps in this group).

**Pump – Flow** sets the pump's flow to the given value.

**Pump – Flow Maximum** sets the pump's maximum flow to the given value.

**Pump – Flow Minimum** sets the pump's minimum flow to the given value.

**Turn Pump On** starts turning the pump on (this will make the WSEL On and Off mode inactivate for this pump).

**Turn Pump Off** starts turning the pump off (this will make the WSEL On and Off mode inactivate for this pump).

**Reactivate WSEL On and WSEL Off mode** this will make the WSEL On and Off mode activate for this pump. The pump will turn on and off based on the water surface elevations entered on the Geometry Pump Station Data editor.

**WSEL On** sets the water surface elevation to turn this pump on (when the WSEL On and Off mode is active for this pump).

**WSEL Off** sets the water surface elevation to turn this pump off (when the WSEL On and Off mode is active for this pump).

Branch (If/Else):

The branching **Branch(If/Else)** operation allows for decision making based on the value of two (or four) expressions. Figure 14-57 shows a simple example. If the gate flow is less than 500 cfs, then the program will go from row #4 to the next operation at row #5. Otherwise, it will skip down to the first row after the End If (row #7). Note that the editor automatically indents the operations between the If and the End If.

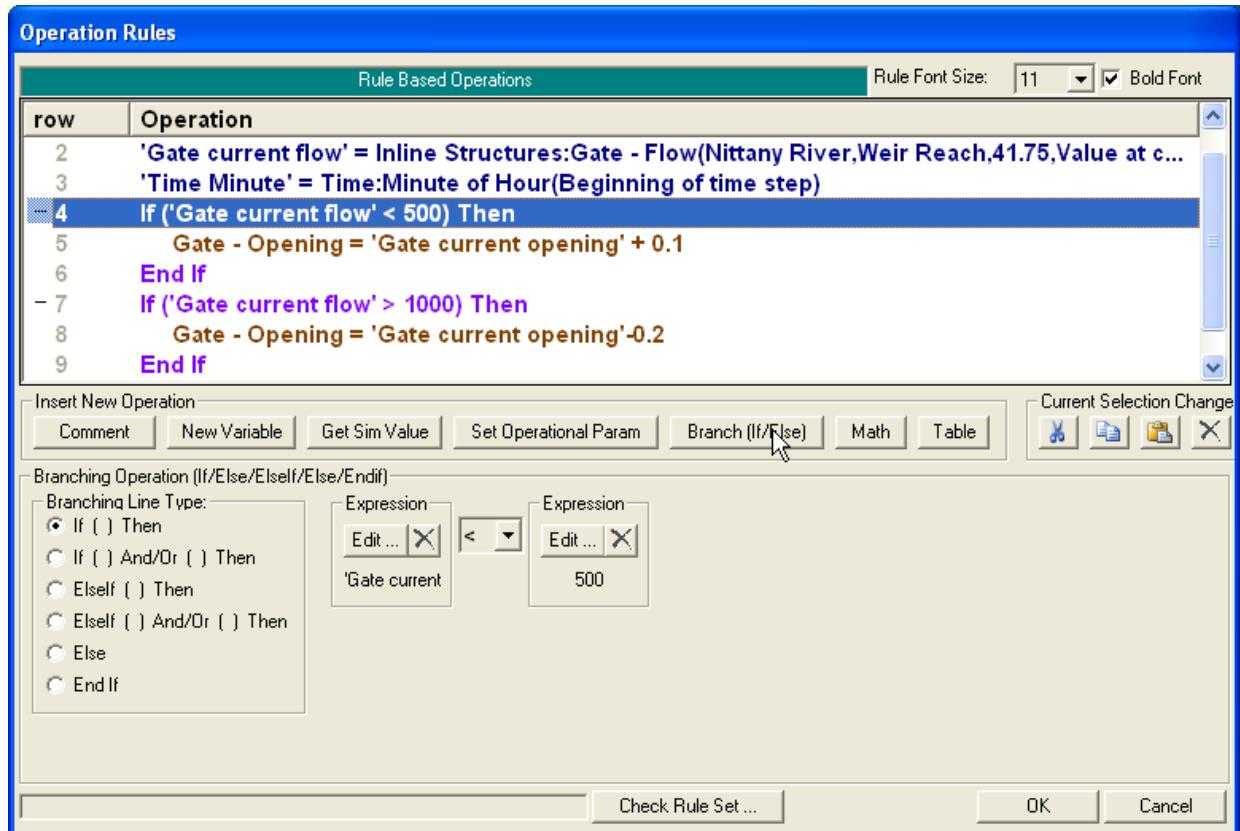


Figure 14 57. Branching Operations

Clicking the **Branch(If/Else)** button brings up a blank (i.e. not set) If/Then as shown in figure 14-58. The user must define a value for both expressions. Going back to figure 14-57, the first expression is the flow through the gate and the second expressions is the constant 500. The user must also choose

a comparison test from the drop down menu between the two expressions. In figure 14-57 (row #4), the comparison is less than. So the If/Then test is true when the first expression is a smaller number than the second (i.e., the flow is under 500 cfs). The comparisons that the user can choose from are: less than, less than or equal to, greater than, greater than or equal to, equal to, or greater than or less than (i.e. not equal to).

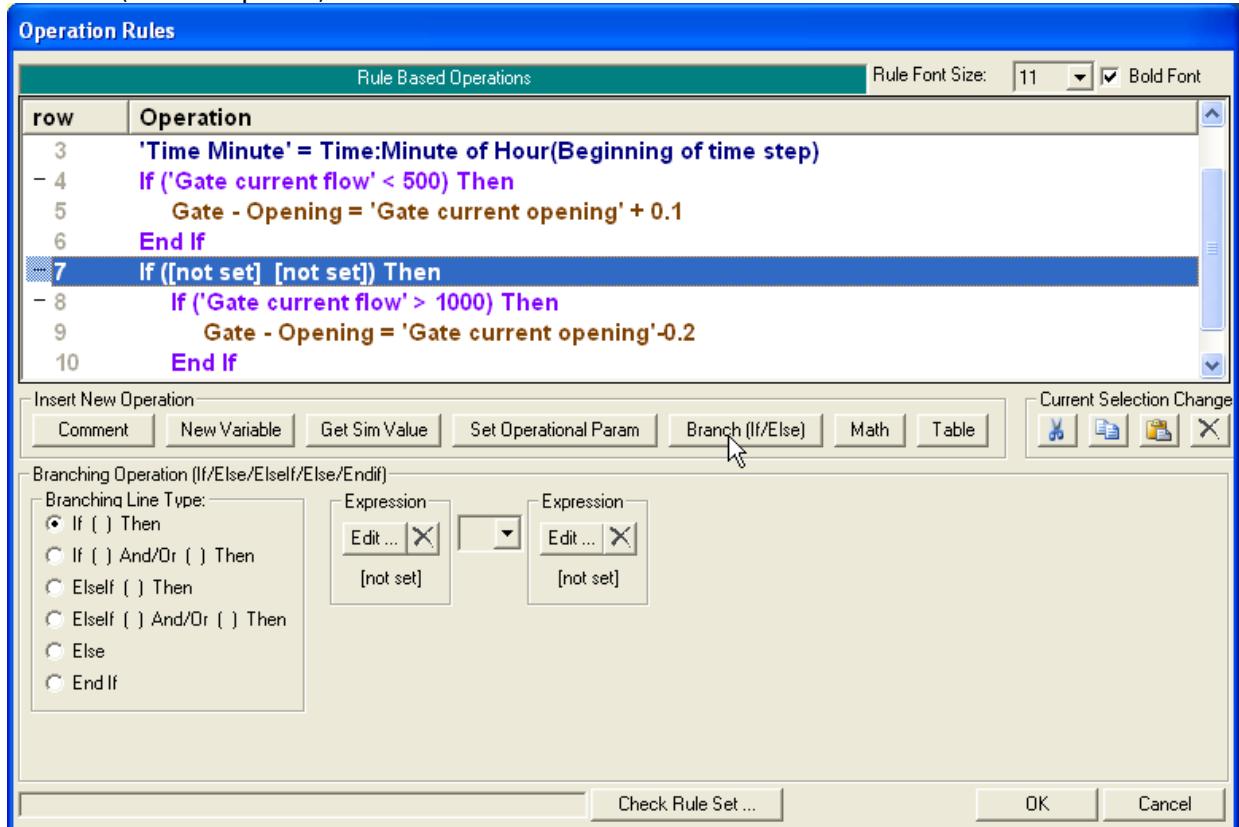


Figure 14 58. Creating a (blank) If/Then Operation

The user must add the End if that is associated with each If/Then. This is done by clicking on the **Branch(If/Else)** which brings up another blank If/Then as shown in figure 14-59. For the **Branching Line Type**, select **End If** as shown in figure 14-60.

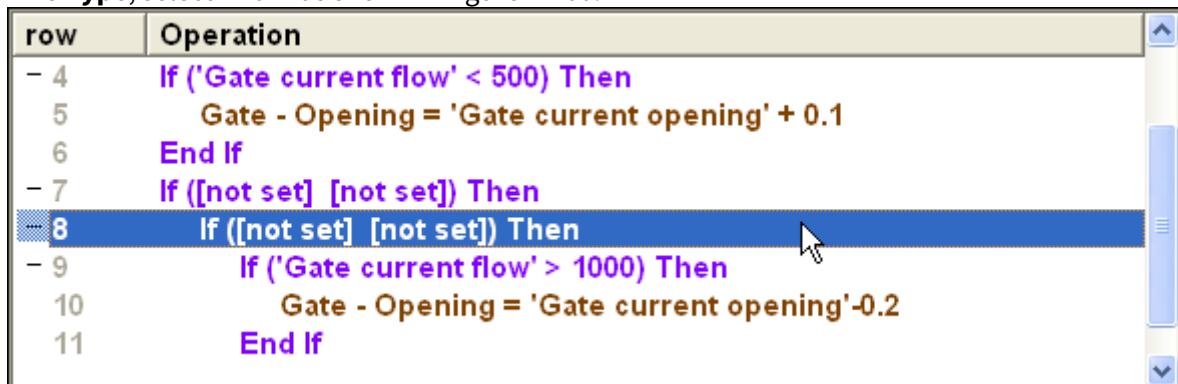


Figure 14 59. Adding another blank If/Then Operation (first step in adding an End if)

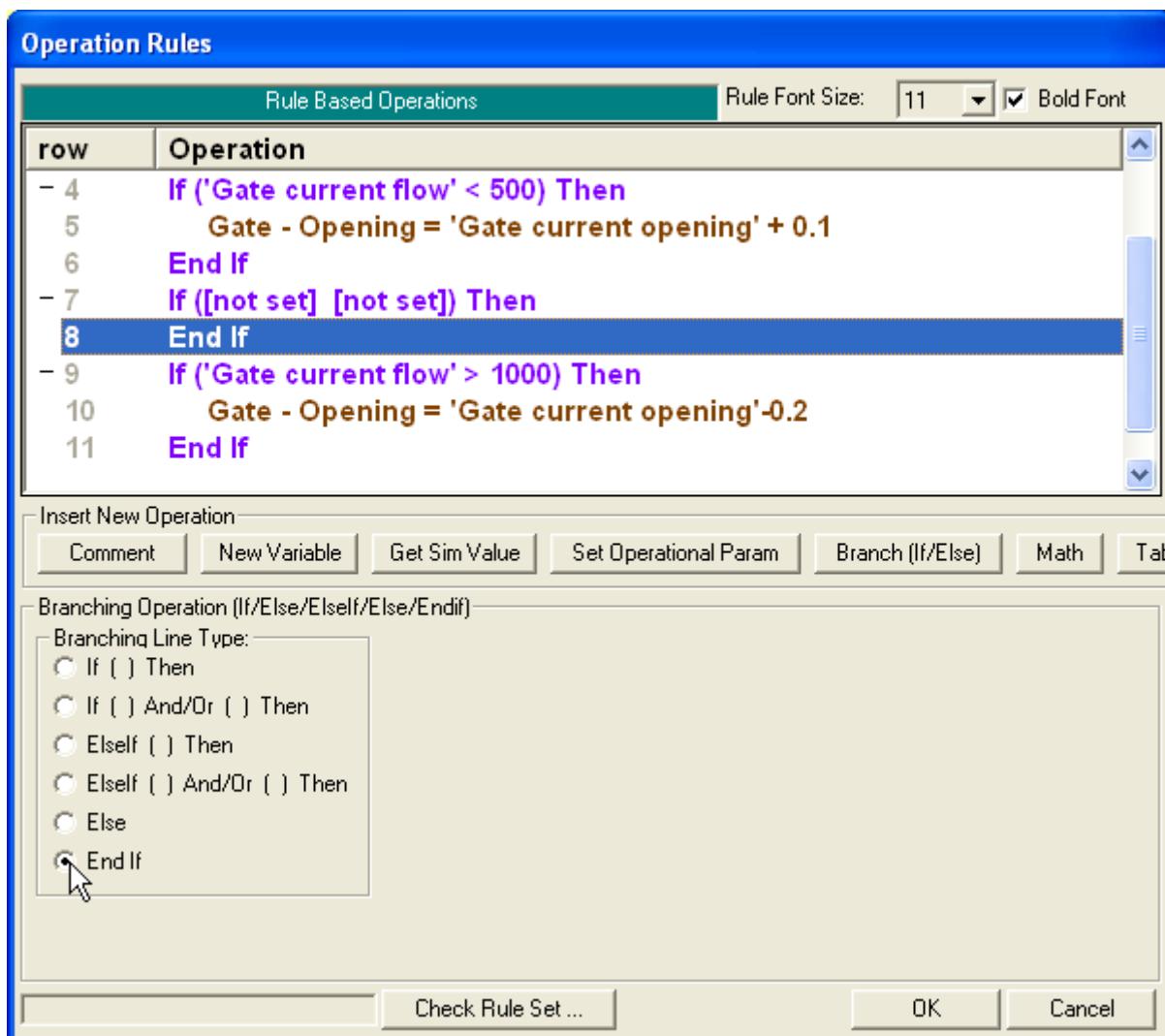


Figure 14 60. Changing the Branching Line Type to an End if

**Important! There must be one, and only one, End If for each If/Then.**

The rule set as shown in figure 14-59 is not valid because it has three If/Then operations but only two End Ifs. Notice how the last operation, row #10, is indented. If the last row is indented, then the rule set is missing at least one (if not more than) End if.

**Although not required, it is highly recommended that the user create the End If operation immediately after adding the If/Then operation.**

When the If/Then is added (figure 14-59), all the remaining operations are indented, which can look confusing. Figures 14-59 and 14-60 show the steps in adding the End if. To add the End If, the **Branch(If/Then)** is clicked again, which adds another If/Then that causes the remaining operations to be indented even further (figure 14-58). However, once the rule operation is changed from an If/Then to an End If (figure 14-59) the remaining operations return to their appropriate location. With the desired If/Then and End If in place, the If/Then operation can be defined and additional rule operations can be inserted between the If/Then and the End If as desired. The program will allow the operations to be added in any order. So the user could, of course, create the If/Then and then add

the additional operations, before finally creating the End If. However, up until the End If is added, the indentation on the display is liable to cause confusion.

**Warning! When an If/Then rule operation is deleted, the user must also delete the appropriate End If.**

Just as it is possible to have more If/Then operations than End If operations, it is also possible to have too many End If operations. There is an erroneous End If in row #7 of figure 14-61 (if there is an End If that does not have an If/Then, it will be displayed in red). If a rule set has a large number of operations with complex, nested If/Then operations, it may be worthwhile to note which End If corresponds to which If/Then before beginning to delete either rule operation.

Rule Based Operations		Rule Font Size:	11	<input checked="" type="checkbox"/> Bold Font
row	Operation			
3	'Time Minute' = Time:Minute of Hour(Beginning of time step)			
- 4	If ('Gate current flow' < 500) Then			
5	Gate - Opening = 'Gate current opening' + 0.1			
6	End If			
7	End If			
- 8	If ('Gate current flow' > 1000) Then			
9	Gate - Opening = 'Gate current opening'-0.2			
10	End If			

Figure 14 61. Erroneous "End If" is displayed in red

Instead of simple If/Then, a two part if test can be done by selecting the If And/Or Then option under the Branching Line Type. This requires the user to define four expressions and select a logical operator, figure 14-62. In the first part, the first two expressions are compared (using a greater than for the example in figure 14-62). In the second part, the third and fourth expressions are compared (using a less than or equal to, in the example). The final step is testing the two parts with the logical operator. For figure 14-62, And has been selected from the drop down menu between the two sets of expressions. If both the first part and the second part are true, then the overall If test is true. If either, or both, the first or second part are false, then the overall If test will be false. The two parts can also be tested with an Or logical operator by changing the selection on the drop down menu. If this is done, the test will be true if either the first or the second part is true. Only if both parts are false will the If test then be false.

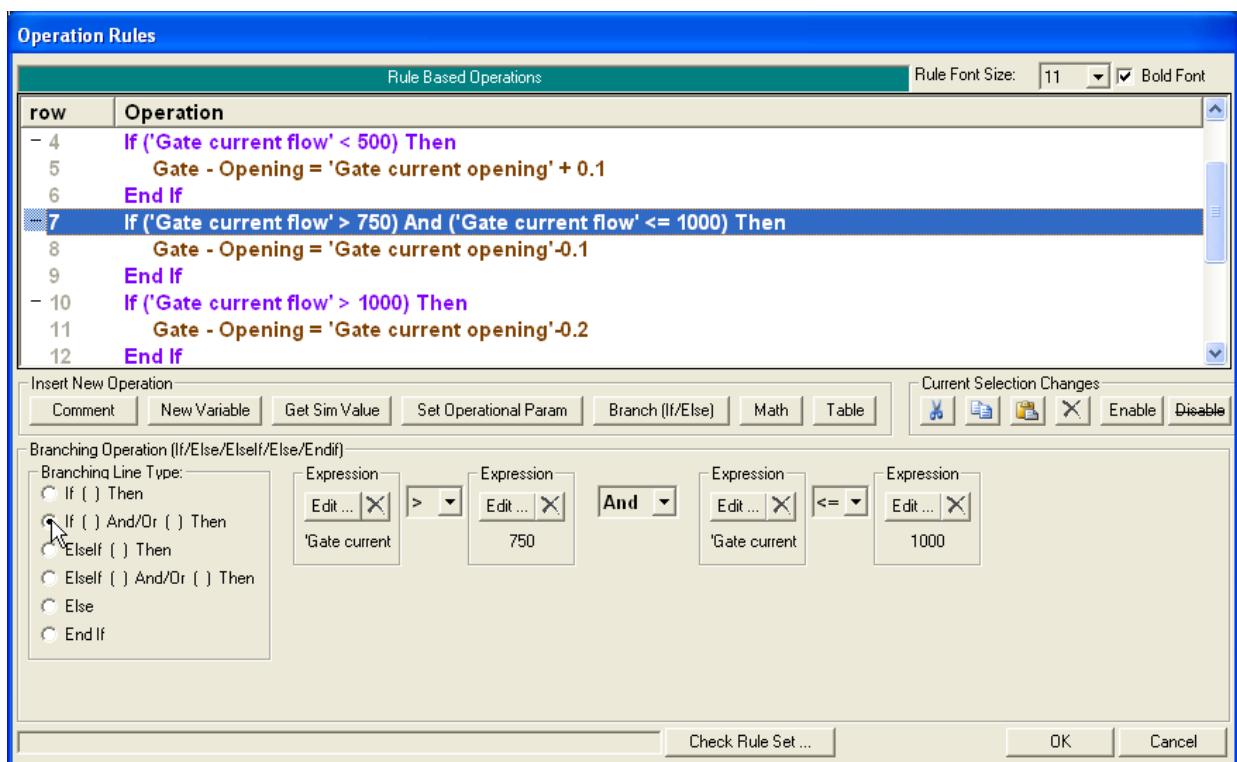


Figure 14 62. Two part If/Then test

If/Then operations can be nested. Figure 14-63 shows an example where the check for the gate adjustment is made at the top of the hour and half past the hour. If the first If/Then is false (not an appropriate time), then control will jump to after the corresponding End If (as shown by the level of indentation) at row 12. Continuing the example, If the first If/Then is true (time to make flow check) then the second If/Then will be evaluated (row #5). Control will go to row #6 or row #8 depending on whether the second If/Then is true or false.

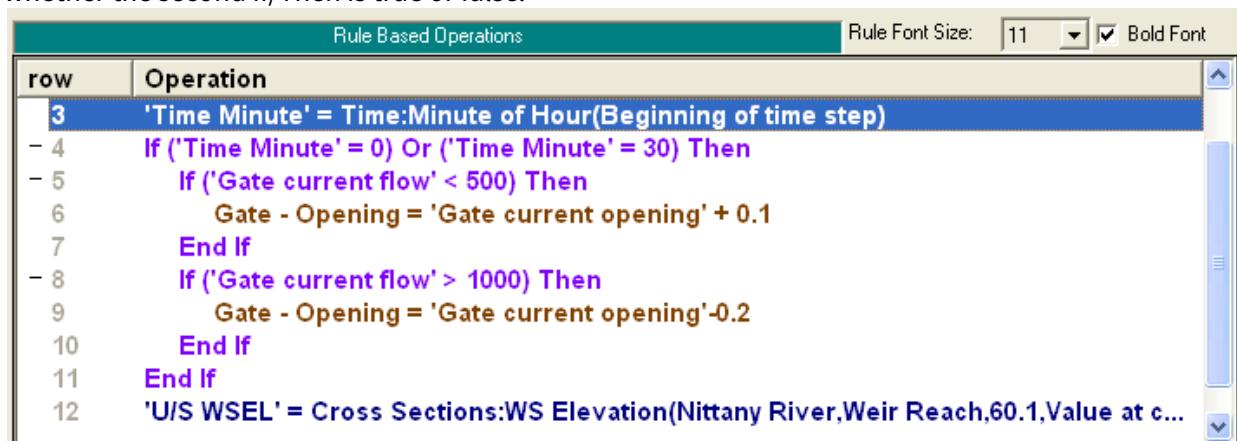


Figure 14 63. Nested If/Then test

After an If/Then and corresponding End If have been added, an Else can be added as shown in figure 14-64. When the original If is false, control will go to the first line after the Else (row #12 in the example). When the original If is true, the operations between the If and the Else will be performed. Once control reaches the Else, it will jump to the End If (after row #10 control will jump to row #14).

row	Operation
- 4	If ('Time Minute' = 0) Or ('Time Minute' = 30) Then
- 5	If ('Gate current flow' < 500) Then
6	Gate - Opening = 'Gate current opening' + 0.1
7	End If
- 8	If ('Gate current flow' > 1000) Then
9	Gate - Opening = 'Gate current opening'-0.2
10	End If
11	Else
12	'U/S WSEL' = Cross Sections:WS Elevation(Nittany River,Weir Reach,60.1,Value a...
13	End If

Figure 14 64. Else Operation

Instead of a simple Else, another option is an Elseif. In this case, there is a second conditional. The operations after the Elseif will only be performed if the initial If is false and the second If (that is, the Elseif) is true. Additional Elseifs can be added as shown in figure 14-65. An Else can also be combined with the Elseif(s). However, there can only be one Else and it must come after the Elseif(s). Therefore, after a simple Else operation, there may not be any more Elseif or Else operations. (The limitations on Elseifs/Else only apply to branching types at the same level of indentation, that is, in the context of the given If/Then End If. There may still be other "nested" conditionals with their own Elseifs and Else operations).

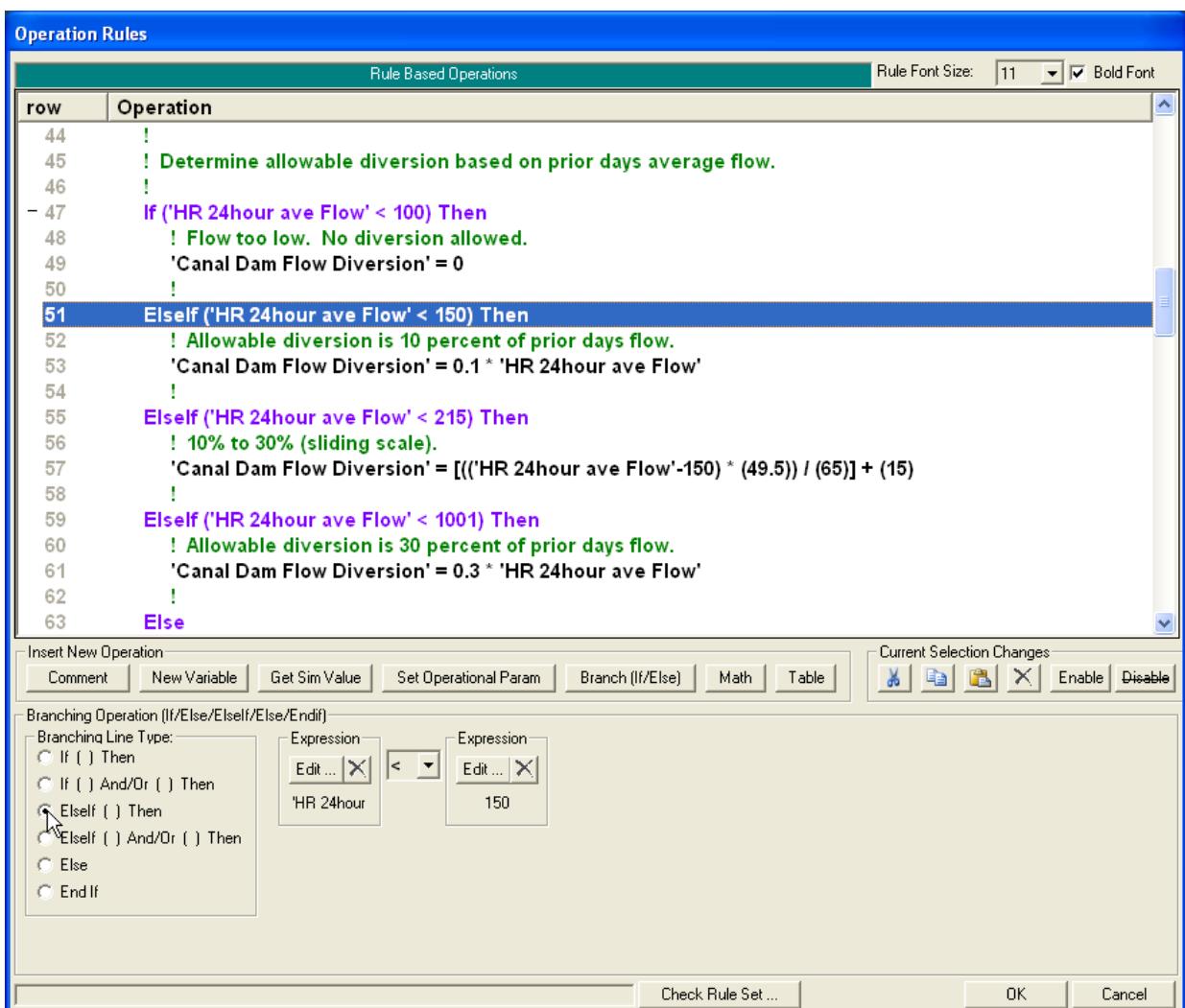


Figure 14 65. Elseif Operations

The display of the rule operations between an If/Then and the corresponding End If may be "collapsed." Note the "" at the beginning of each If/Then rule. Clicking the "" will change it to a "" and the display of the rules between the If/Then to End If will collapse as shown in figure 14-66. These rules are still in effect (collapsing rules does not change their operation). This option merely changes the display, and it is intended to make large rule sets easier to understand and manage. Clicking the "" will expand the rules back to their original form. Note: all of the operations under the **Current Selection Changes** (cut, copy, paste, etc), see below, function normally even on collapsed regions. In the above example where the collapsed region is highlighted, clicking the **Delete** button would delete rules 47 through 66.

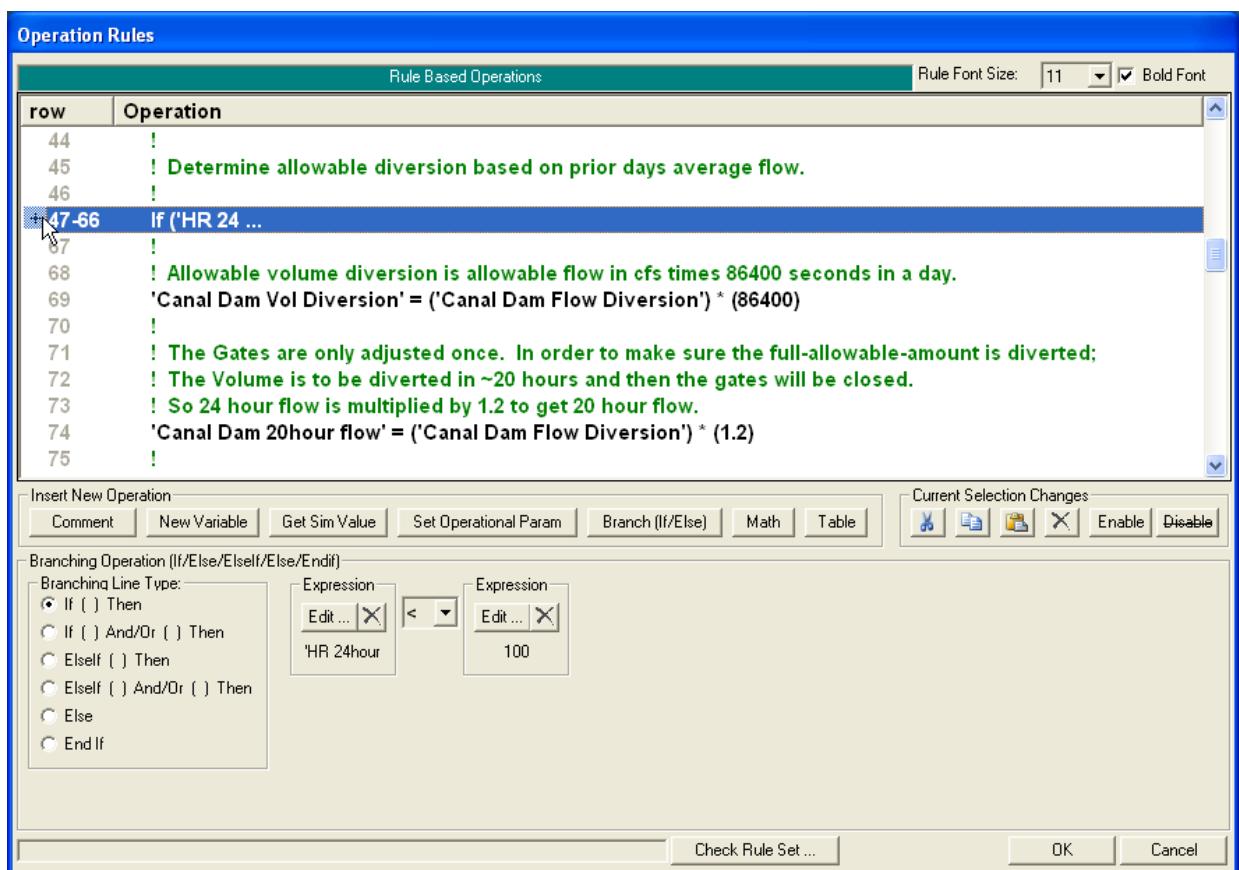


Figure 14 66. Collapsed If-Then

#### Math

Clicking the **Math** operation button creates a [blank] math operation as shown in figure 14-67. The result of the math operation can be assigned to either a new variable or an existing variable (in the same way that a get simulation variable can be assigned, as above).

The math operation itself is composed of up to four different “expressions.” Each expression that is defined will return a real number. Expressions should be defined from left to right. So if a math operation is composed of two expressions, the left two expressions should be defined and the right two expressions should be left as “[not set]” (i.e. they should be left blank). If more than one expression is defined, then the user must choose an algebraic connector from the drop down menu between them. The choices are: addition, subtraction, multiplication, and division.

The value of each individual expression is determined and then the remaining algebraic operations are performed from left to right. So if the math operation has the three expressions as shown in figure 14-68, the first two expressions are added together and that sum is then divided by the third expression.

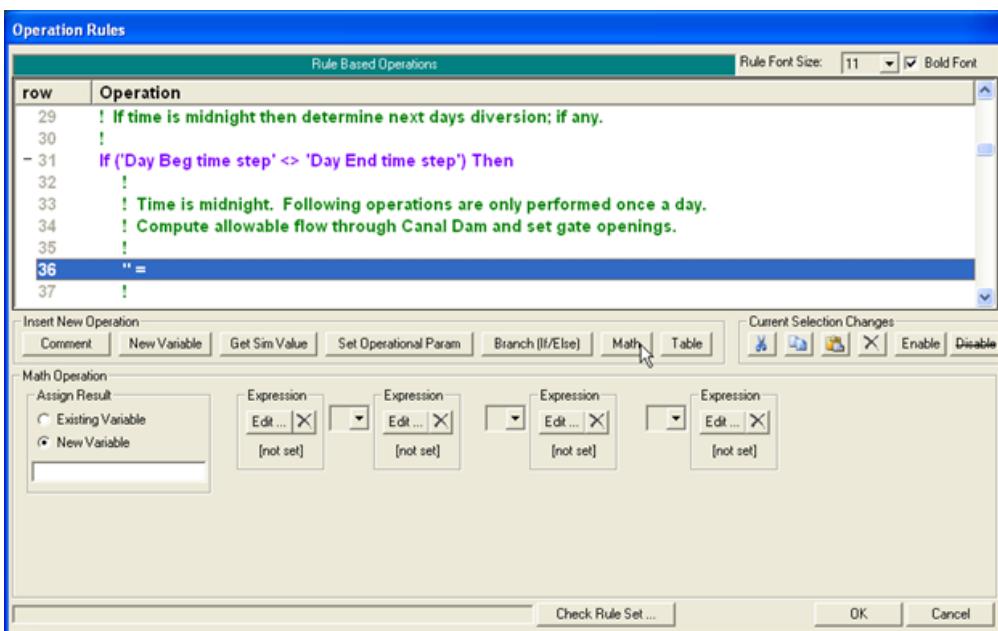


Figure 14-67. Blank Math Operation

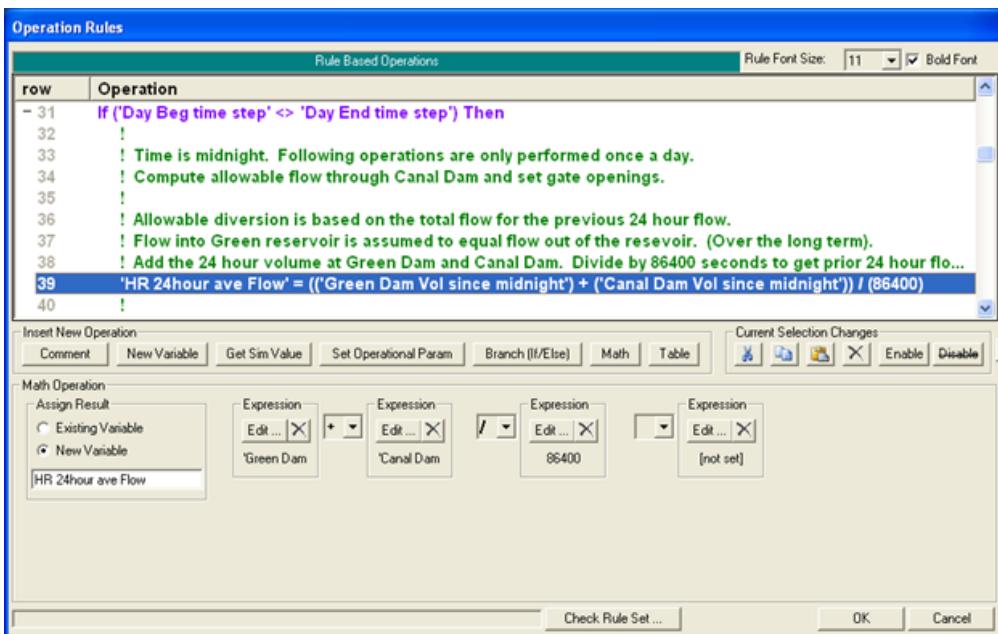


Figure 14-68. Three Expression Math Operation

**Expression.** To define an expression, click on the **Edit** button to bring up the Edit Rule Expression editor as shown in figure 14-69. If no values have been entered (or if the **Clear Expression** button has been clicked), then the current expression will be shown as "[not set]."

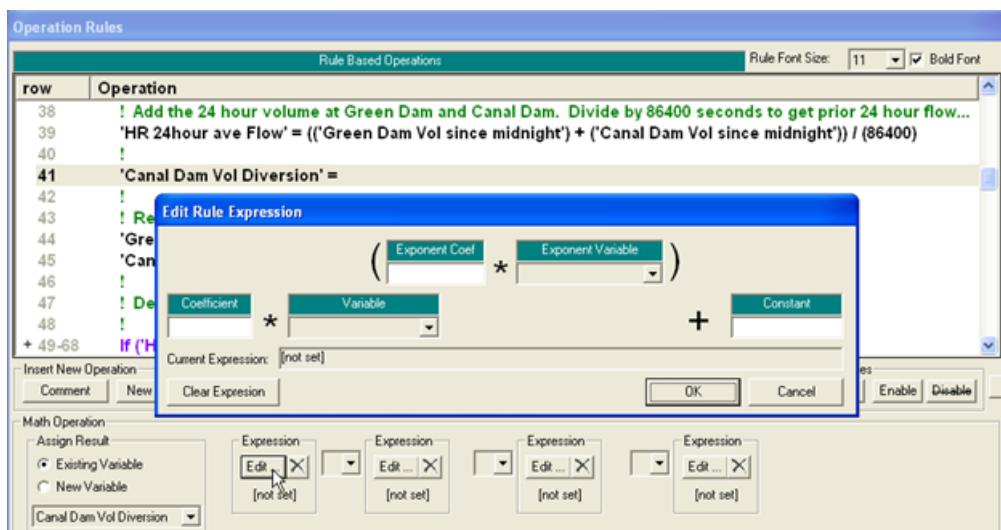


Figure 14-69. Blank Rule Expression

Up to five fields in the expression editor can be defined. Any that are not defined are ignored. The simplest expression is to enter a single number in the Constant field as shown in figure 14-70. In this example, this expression will always have a value of 5.

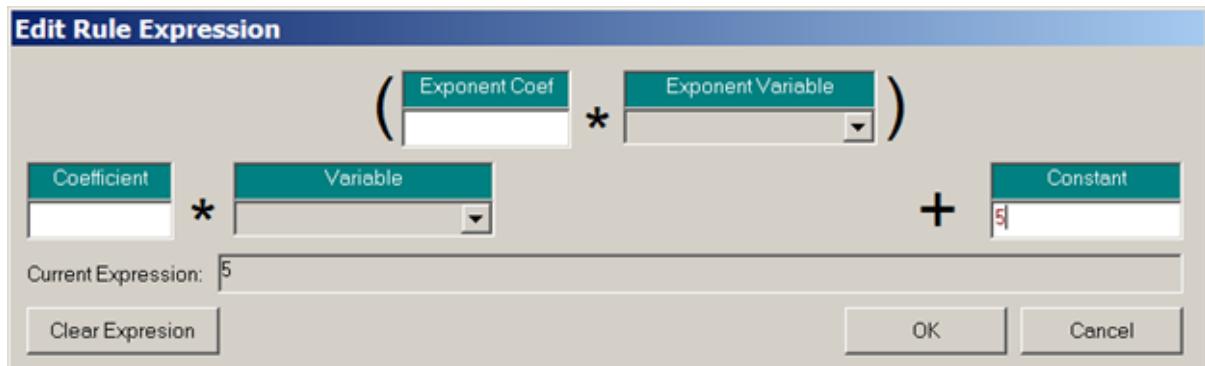


Figure 14-70. Rule Expression set to a constant

Another simple example is shown in figure 14-71. Here a preexisting variable has been selected from the drop down menu. This expression will return the current value of this variable. An optional coefficient can be added in front of the selected variable and a value may also still be added under the constant field.

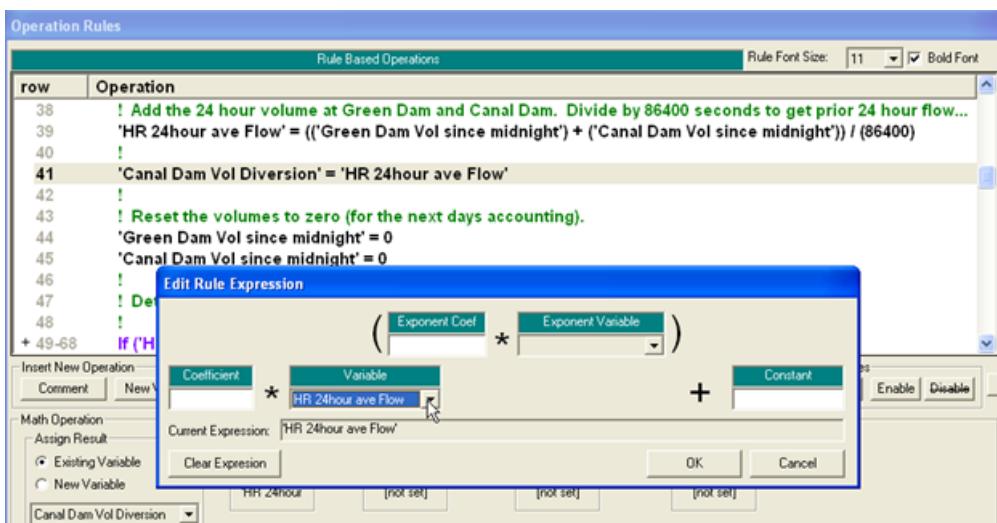


Figure 14-71. Rule Expression set to an existing variable

The Variable can also be raised to an exponent by entering a value in either or both of the fields inside of the parenthetical. If only the Exponent Coefficient or the Exponent Variable is defined, then the variable is raised to the given value of the Exponent Coefficient or Exponent Variable, see figure 14-72. If both are defined, then the Exponent Variable is multiplied by the Exponent Coefficient and the given Variable is raised to the resulting product.

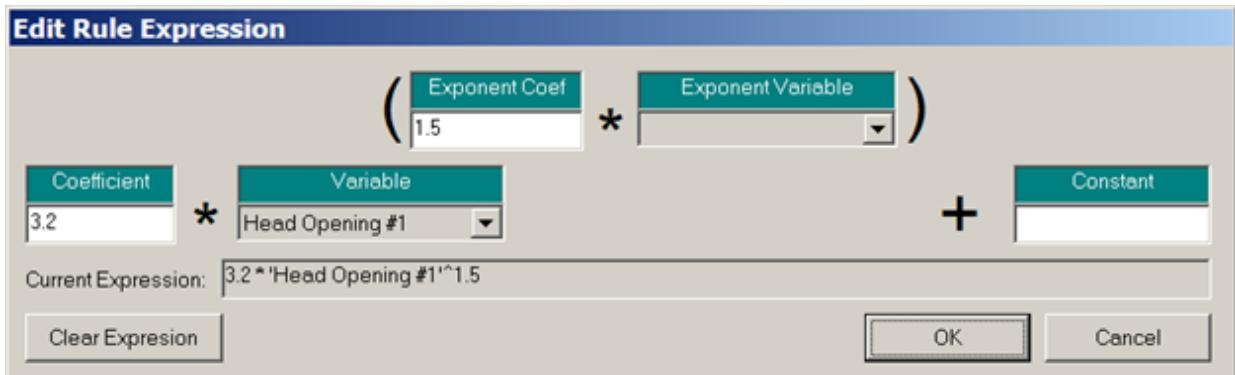


Figure 14-72. Variable raised to an exponent multiplied by a coefficient

Note: each expression is always determined before operations between expressions are performed.

## Table

The final operation type is a table lookup. Clicking the **Table** operation button creates a table operation as shown in figure 14-73. The result of the table lookup can be assigned to a new variable or an existing variable. The table can be either one or two dimensional.

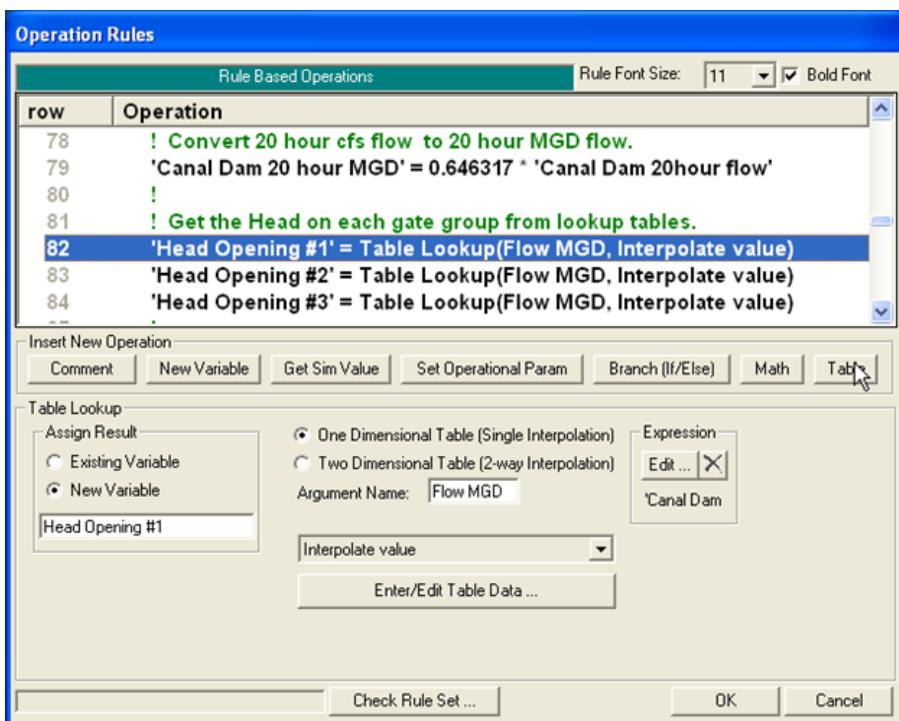


Figure 14-73. Table lookup operations

Figure 14-73 shows a one dimensional table operation. The table data can be entered (and/or viewed) by clicking on the **Enter/Edit Table Data...** button. This brings up the Rule Table editor as shown in figure 14-74.

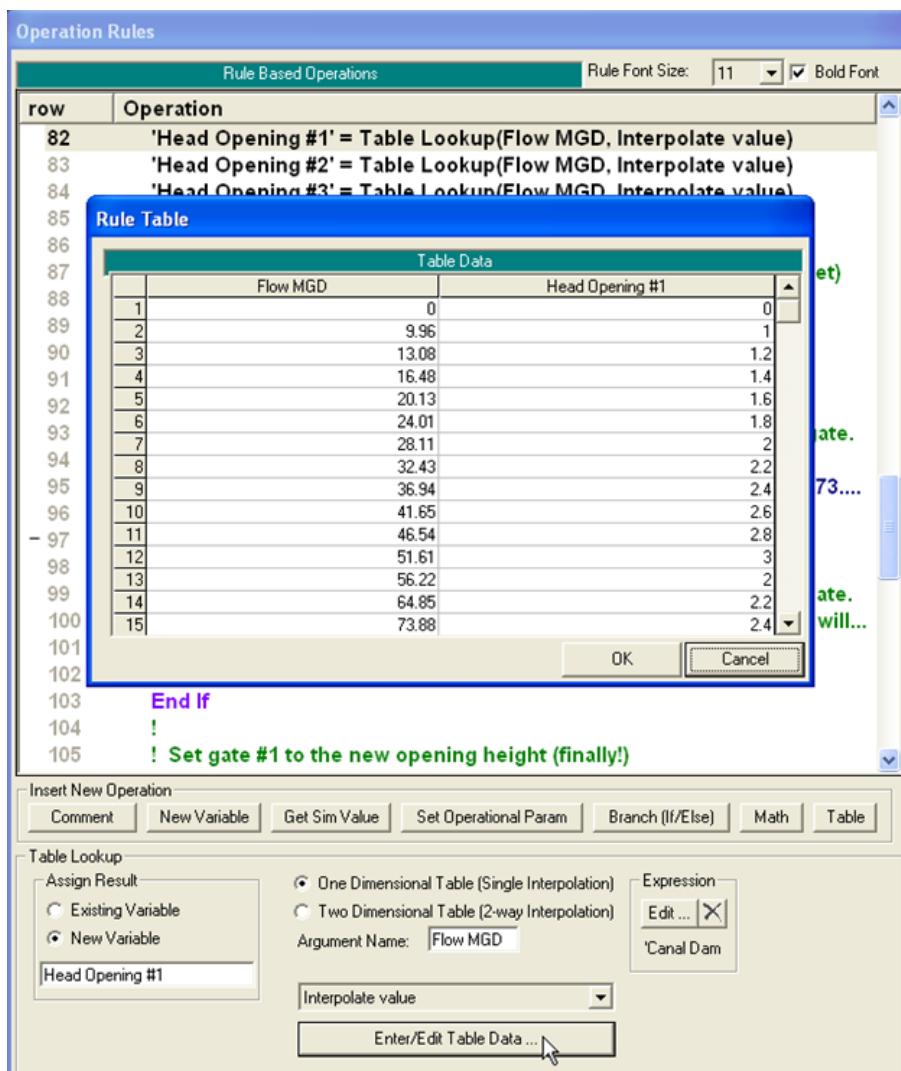


Figure 14-74. Rule Table Editor

When the table operation is performed, the program will determine the value of the Expression, which starts out 'Canal Dam' in the example in figures 14-73 and 14-74. The location of the Expression value is determined in the left hand column of the table (figure 14-74) and the corresponding lookup value is determined from the right hand column. In the above figure, if the value of the expression happens to equal 13.08, the result of the table lookup would be to assign the value 1.2 to the variable "Head Opening #1."

The Argument Name ("Flow MGD" in the above example) is used as the heading for the left column. This is only used as a label. (Alternately, the program could have used the numeric formula in the given expression as the heading label, but this could be rather long and awkward.) This label is only used as a heading in the Rule Table editor (it is not a user selectable variable).

The right hand column is labeled with the assignment result. In this example, the result of the table lookup is being assigned to a new variable called "Head Opening #1."

By default, the lookup will interpolate between values. So in the above example, if the expression equaled 14.78, the lookup would return 1.3. This can be changed by the drop down menu that is just above the **Enter/Edit Table Data**. There are three other choices. “Nearest index value” will move up or down to the nearest value (14.7 would return 1.2 and 14.8 would return 1.4, in the above table). “Index <=value” and “Index >=” will go down or up to the next value in the table. These other options can be useful for forcing exact gate settings. For instance, if it was desired that the gates only be opened to the nearest tenth of a foot, values in tenths (e.g., 3.0’, 3.1’, 3.2’, etc) could be entered in a table and “Nearest index value” selected. The result of the table lookup could then be used to set the gate.

*Tip: Another possibility for forcing exact gate settings is to use an integer user variable. Assume that the gate can be opened in hundredths of a foot (e.g., 3.00’, 3.01’, 3.02’, etc.). These could be [tediously] entered into a table. Alternately, the approximate gate opening could be determined, say for example, 3.028 feet. This value could be multiplied by 100 to get 302.8. This value, 302.8 could be assigned to an integer user variable which would result in 303. Finally, this could be divided back by a 100 (assigning the result back to a real variable) to get 3.03 that could then be used to set a gate opening.*

**NOTE: By default the software will extrapolate at both ends of the user entered tables.**  
**Meaning, if a value is requested above the last value of the Table, the software will take the last two points in the table, project a straight line, and then extrapolate to get a value. The same thing is done at the lower end of the table. If a value is requested that is below the lowest point in the table, the first two values are used to create a straight line, and a value is extrapolated below the table. Therefore if you want to control the lower end of the table, you will need to put in an extra row that will encompass all possible values requested.**

Instead of a one dimensional table, the other option is a two dimensional table as shown in figure 14-75. The editor now has two expressions and two argument Names (the top argument name corresponds with the left expression and the bottom argument name corresponds with the right expression). Clicking **Enter/Edit Table Data** brings up an expanded Rule Table as also shown in figure 14-75. As before, the left most column corresponds to the value in the first expression. The top row now corresponds to the second expression. The value in the table is determined by two way interpolation (or nearest value depending on the interpolation option). So in the table shown, if the first expression (“Inline Flow”) is equal to 5000 and if the second expression (“Hour”) is equal to 9, then the value from the table lookup would be 400.

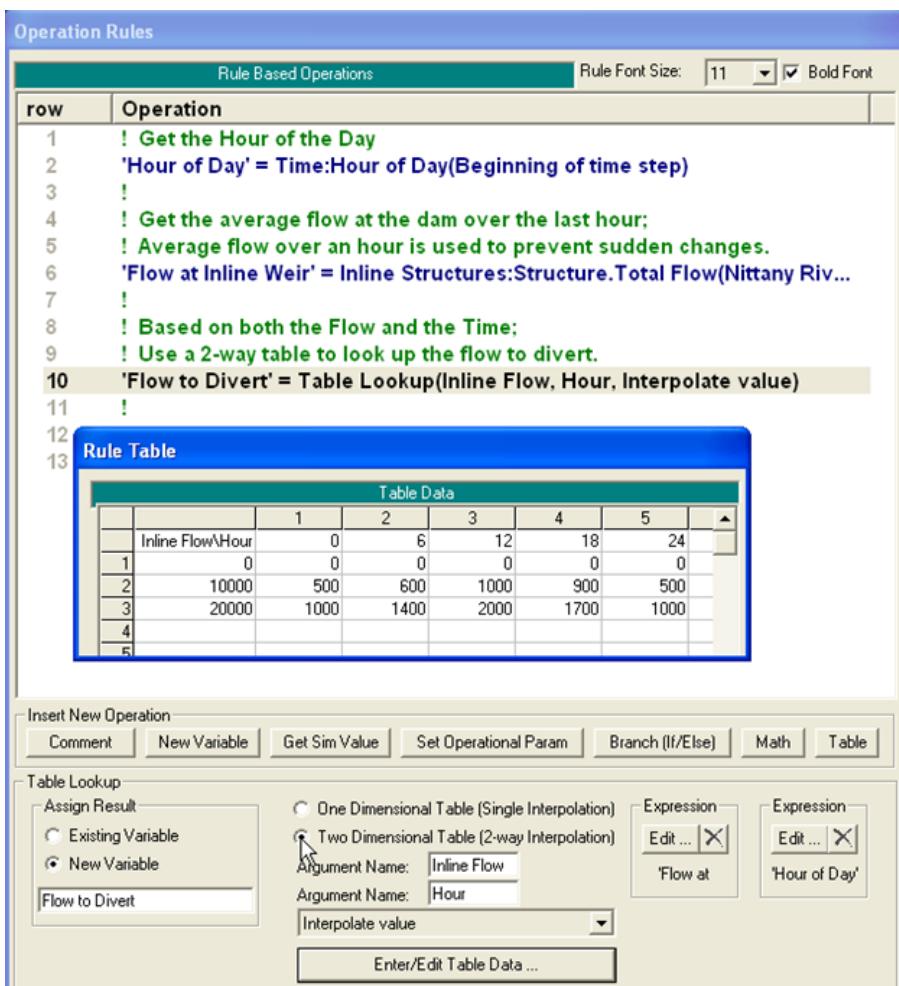


Figure 14-75. Two dimensional Table

**Current Selection Changes.** On the right hand side of the Operation Rule editor (figure 14-76), are six buttons for manipulating the current, highlighted rule (or selection of rules). The **Cut**, **Copy**, **Paste**, and **Delete** buttons operate in a normal, Windows manner. One or more rules may be selected using the keyboard (e.g. **Shift + down arrow**) or the mouse pointer (e.g. **Ctrl + click**) as shown in figure xxx. A copy of the rule(s) can be put on the Clipboard with the **Copy** button and can then be pasted (using the **Paste** button) to another location, as shown in figure 14-80.

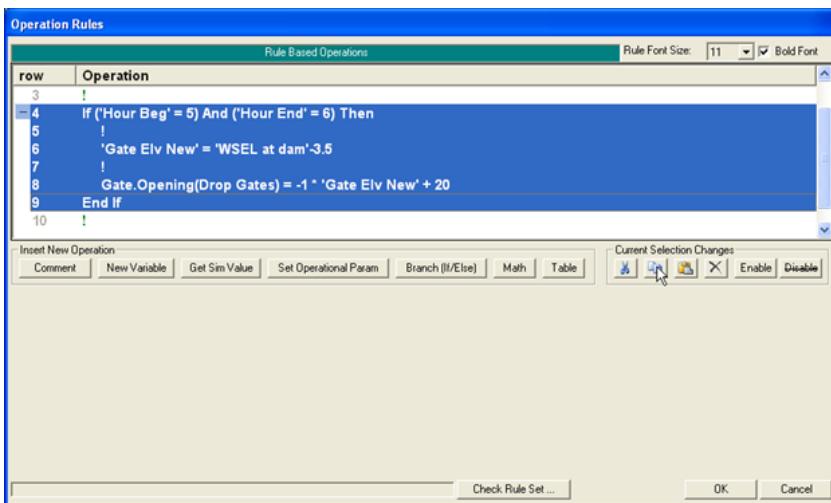


Figure 14-76. Copying Highlighted Rules

The copy function makes an exact duplicate of the selected rules. This can generate potential “errors” that the user will have to correct. For instance, in the above example of using the copy function, rule 6 is a Get operation that assigns the result to the “New Variable” named Gate Elv New. The copy of this rule, rule 13 in figure 14-77, is also assigning the result to the same “New Variable” named Gate Elv New. After copying this rule, the user must change one of the “Gate Elv New” names to something else. Or, if it is intended that the copy use the same variable, the user should change the assign result for the copied rule to “Existing Variable” and then select Gate Elv New from the drop down menu.

Since the copy function uses the standard Windows Clipboard, rules can be pasted into a completely different rule set, or the user can even open up a different plan (or different RAS project) and paste the results. **The user will have to correct any erroneous variable names or references (different cross section river stations, different gate group names, etc.).**

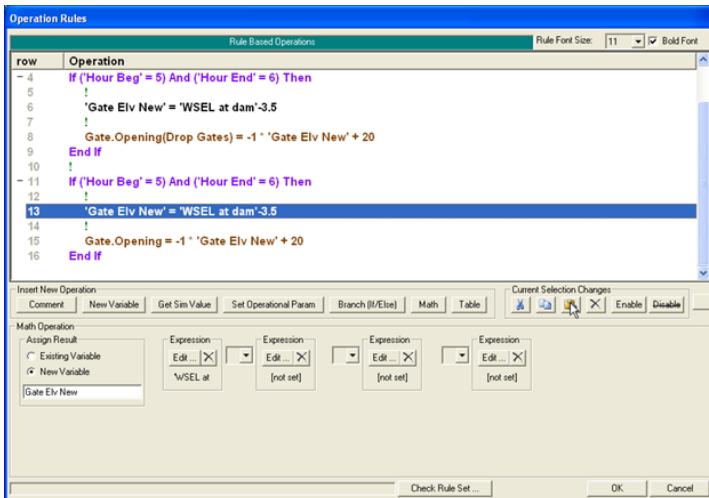


Figure 14-77. Pasting Rules

The **Cut** button will move the highlighted rule(s) to the clipboard. After the rules have been removed by cutting, the **Paste** button can then be used as a “move” operation. The **Delete** button permanently removes the highlighted rules. There is no “undo” operation, so care should be exercised when using the **Delete** button. However, if a mistake is made, the **Cancel** button will cancel all the changes that have been made since the Operation Rules editor was opened. *Tip: frequently saving the changes made in the Operation Rules editor allows the **Cancel** button to be used as an “undo” operation without canceling too much work.*

Note: If a collapsed If/Then-End If block is highlighted, then it will still be subject to copy, paste, and delete/cut, just as it would be in its fully expanded state.

**Tip: The standard Windows shortcut keys: Ctrl + “x”, “c” or “v” may be used instead of clicking on the Cut, Copy, or Paste buttons.**

Checking the **Disable** button is a quick way to temporarily remove the highlighted operations (it will cause the highlighted operations to be displayed as green comment lines with a strikethrough), see figures 14-78 and 14-79. These operations will no longer be performed by the program (be careful disabling Branching Line Types). Clicking the **Enable** button will restore the operations.

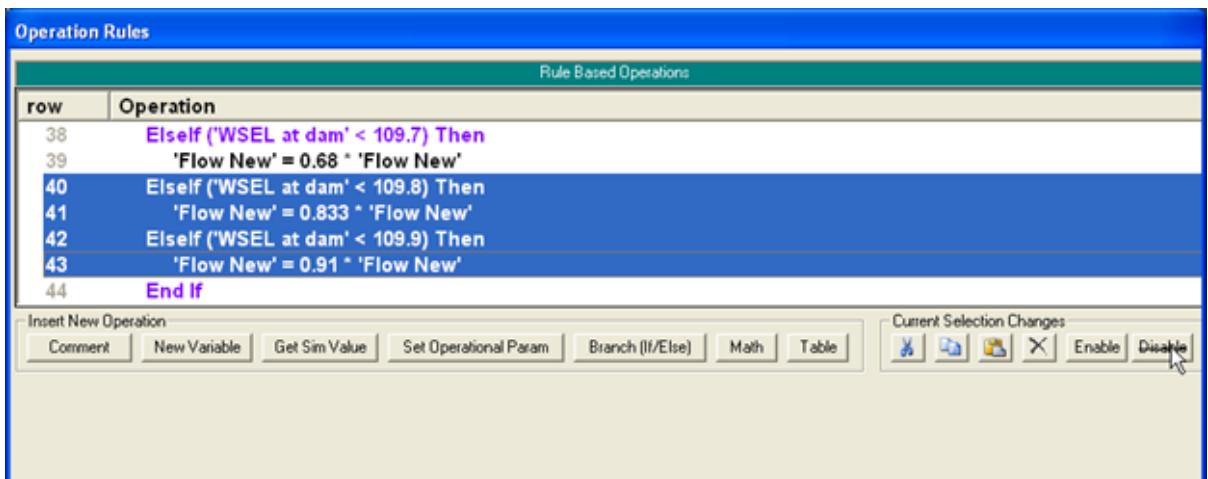


Figure 14-78. Disabling Highlighted Rules

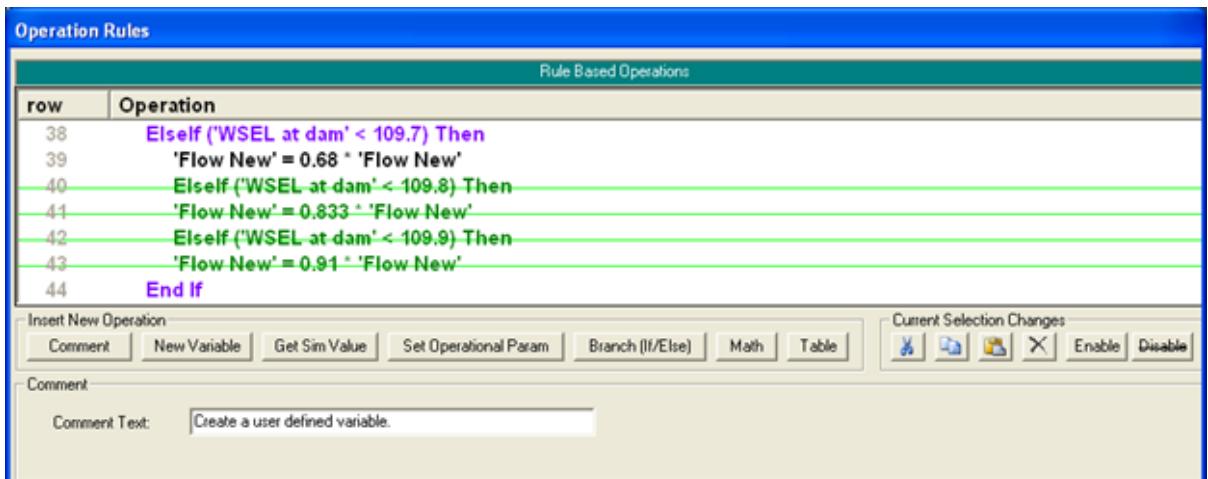


Figure 14-79. Disabled Rules

The **Copy Rules Text to Clipboard** will copy the display text of the entire rule set to the clipboard (figure 14-80). This can then be pasted, for instance, as simple text into Notepad or a Word document report (figure 14-81). This copy is for “display” only and may not be pasted back into a rule set.

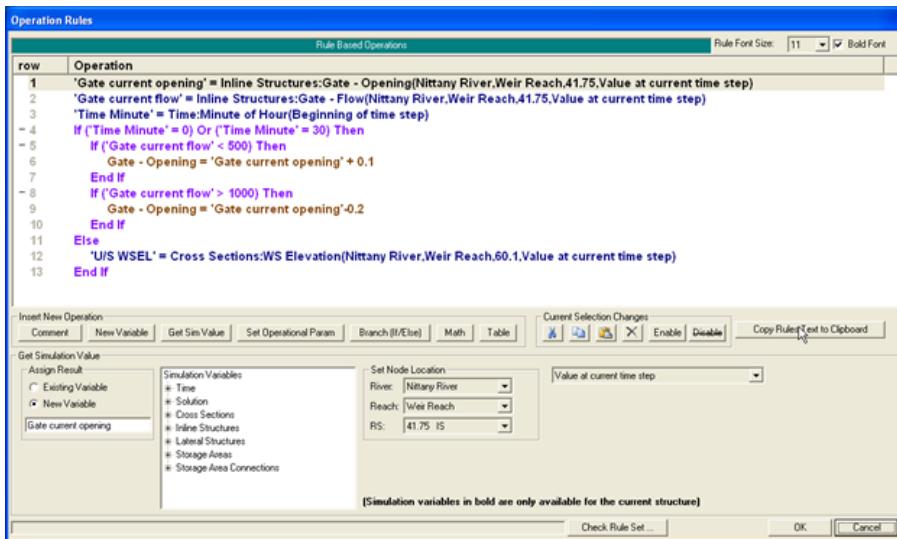


Figure 14-80. Copying Rule Text to the Clipboard

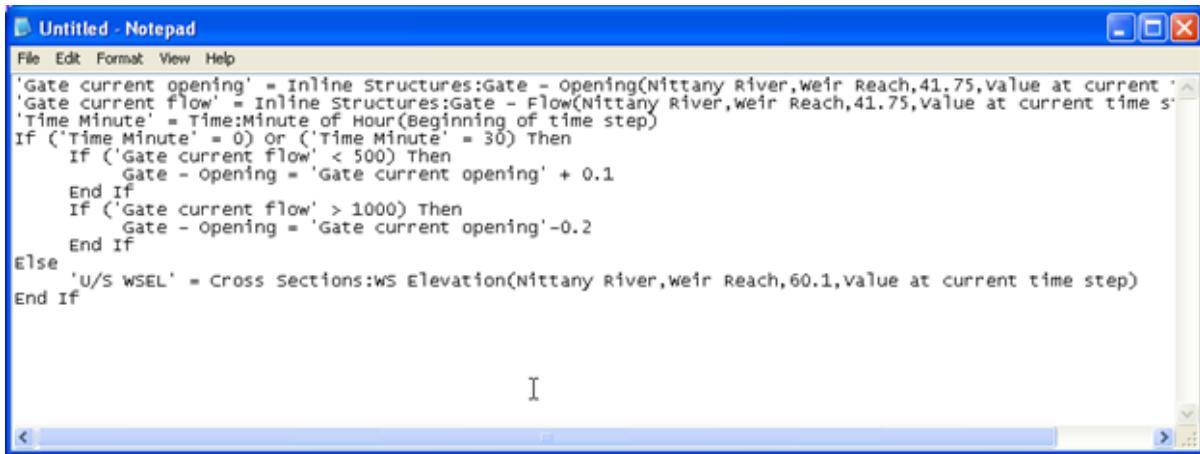


Figure 14-81. Text Pasted into Notepad

Clicking the right mouse button (on a given row) will display a popup editor as shown in figure 14-82. In addition to the functions described above, the Insert New Operation functions are also available in this manner.

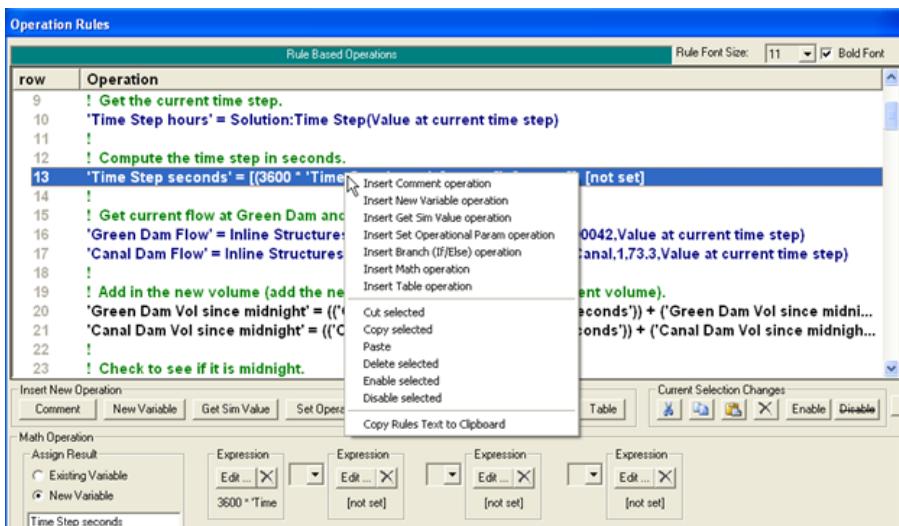


Figure 14-82. Right mouse click on a line

Clicking the **Check Rule Set...** button will cause RAS to check the rule set for common user errors. All of the rule sets in the model will also be checked when an unsteady flow run is launched. The **Check Rule Set...** button is just a convenient way to find and fix rule errors for the given rule set while the Operation Rules editor is opened.

If no errors are found, RAS will display a message stating that no inconsistencies were found. Otherwise, RAS will display a list of the mistakes and the line numbers they occur. An example is shown in figure 14-83. Common problems are: a variable name that has been defined more than once, a reference to a non-existent variable (the variable was renamed or deleted), “unbalanced” If/ Then End If operations, or a reference to a non-existent node (e.g. a river station that has been removed from the project).

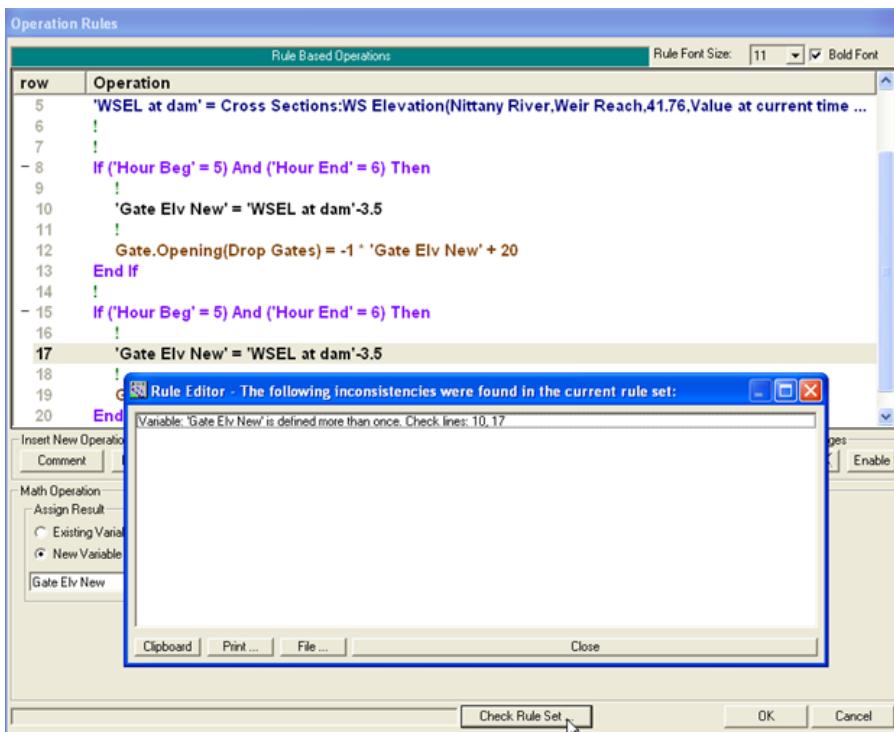


Figure 14-83. Checking the Rule Set

**Font.** In the upper right hand corner of the Operation Rule editor, there is a drop down menu where the user can change the rule operation font size. The font can also be toggled between normal and bold by checking the Bold Font box.

**Detailed Log Output.** If the detailed log output is turned on, then results from each rule set will be sent to the log file during runtime, see figure 14-84. On the left side is the row number of the operation followed by the result of the operation. For instance, the operation at row #19 results in the variable 'Tampa Dam Flow' being set equal to 324.9499. Row #32 is an If/Then test that came back false. For this time step, for this operation, the first and second expressions are both equal to 4. This results in the less than or greater than (i.e. 'not equal to') test being false. Since the test is false (and there is not a corresponding ElseIf or Else), control jumps to after the End If, which happens to be row #121. Row #121 is a two part If/Then test that is also false. It is expected that additional, (tabular and graphical) output from rule set operations will be added to future versions of HEC-RAS. For the current version of RAS, the log output may be the best way to track down user programming mistakes. Rule operations that are valid (as far as RAS is concerned), but do not produce the result desired by the user. For instance, a Get operation that references the wrong cross section node.

```

TampaExamples.bco - Notepad
File Edit Format View Help
Hydrographs written at 22.50hours.

*** Start of New Time Step ***

Rule Set for Harney 1 73.3

0008      Tampa Dam 4 Hour Ave Flow = 280.193
0013          Time Step hours = 0.25
0016          Time Step seconds = 900.
0019          Tampa Dam Flow = 324.9499

0020          S-161 Flow = 39.24155
0023      Tampa Dam Vol since midnight = 8751051.
0024          S-161 Vol since midnight = 1439424.
0029          Day Beg time step = 4.

0030          Day End time step = 4.
0032      False    4. <= 4.
0121      False    1439424. >= 6198242. OR    280.193 < 10.

```

Figure 14-84. Detailed log output

#### Simulation and Operational Variables.

The following is a list of the currently available simulation output variables and operational variables that can be set.

##### Time variables

Julian Day: Days since December 31, 1899 (e.g. 01Jan2000 = 36525).

Year: Year (e.g. 2006).

Month: Month of the year (e.g. August = 8).

Day of Year: e.g. Jan 1 = 1. Feb. 1 = 32. Dec 31 is 365 (non-leap year).

Day of Water Year: e.g. Oct 1 = 1. Sept 30 = 365 (non-leap year).

Day of Month: e.g. 22Jan2000 = 22.

Day of Week: Integer day starting on Sunday. e.g. Sunday = 1, Monday = 2, Saturday = 7.

Hour of Day: Integer hours since midnight (e.g. 01Jan2000 1245 = 12).

Minute of Hour: Integer minutes after hour (e.g. 01Jan2000 1245 = 45).

Second of Minute: Integer seconds after minute (e.g. 01Jan2000 1245:15 = 15).

Hour of Day (fractional): (fractional) Hours since midnight (e.g. 01Jan2000 1245 = 12.75).

Hour of Simulation: (fractional) Hours since simulation started.

##### Solution variables

Time Step: Length of current time step in hours.

Iteration Number: Number of iterations, for given time step (“current time step” will not have relevance until rules are allowed for every iteration, see above, but “previous time step” will return the number of iterations from the last time step).

**WS Error Max:** Maximum error, for given time step, in computed water surface at any cross section (“current time step” will not have relevance until iterations are allowed, see above, but “previous time step” will return the maximum error from the last time step).

**Flow Error Max:** Maximum error, for given time step, in computed flow at any cross section (previous time step only).

**WS SA Error Max:** Maximum error, for given time step, in computed water surface at any storage area (previous time step only).

Cross Sections variables

**WS Elevation:** Water surface.

**Flow:** Flow.

**WS Change:** Change in water surface, for given time step (previous time step only).

**Flow Change:** Change in flow, for given time step (previous time step only).

**WS Error:** Error in water surface, for given time step (previous time step only).

**Flow Error:** Error in flow, for given time step (previous time step only).

**Bed Change:** The change in elevation of the thalweg (only applicable for Unsteady Sediment simulations).

**Sediment Concentration:** The concentration of sediment leaving the cross section (only applicable for Unsteady Sediment simulations).

Inline Structures, Lateral Structures, and Storage Area Connections variables:

**Structure - Total Flow:** Total flow for the inline structure.

**Structure - Total Flow (Fixed):** Force the given flow for the inline structure.

**Structure - Total Flow (Desired):** Compute gate settings to provide the total given flow for the inline structure.

**Structure - Flow Additional:** Add in the additional given flow to the inline structure.

**Structure - Flow Maximum:** Set a maximum flow for the inline structure.

**Structure - Flow Minimum:** Set a minimum flow for the inline structure.

**Structure - Total Gate Flow:** Flow for all of the gate groups.

**Structure - Total Gate Flow Maximum:** Set a maximum flow for all of the gate groups.

**Structure - Total Gate Flow Minimum:** Set a minimum flow for all of the gate groups.

**Structure - Stage (Fixed):** Sets the given stage. Computes the flow through the inline structure that is required to produce the given stage.

**Weir - Flow:** Flow over the weir.

**Weir - Flow Maximum:** Set a maximum flow over the weir.

Weir - Flow Minimum: Set a minimum flow over the weir.

Weir - Weir Coefficient: Weir coefficient for the weir.

Weir - Minimum Elev for Weir Flow: Minimum weir elevation for flow for the weir (water surfaces below this elevation will not produce weir flow).

Weir - C Simple (Positive): Linear routing coefficient for positive flow (linear routing weirs only).

Weir - C Simple (Negative): Linear routing coefficient for negative flow (linear routing weirs only).

Weir - Submergence: Fractional submergence for the given weir (e.g. 0.97).

Gate - Flow: Flow through the gate group.

Gate - Flow (Fixed): Force the given flow for the gate group.

Gate - Flow (Desired): Compute gate setting to provide the given flow for the gate group.

Gate - Flow Maximum: Set a maximum flow through the gate group.

Gate - Flow Minimum: Set a minimum flow through the gate group.

Gate - Opening: The [current] gate opening height for the gate group.

Gate – Opening (target position): Gate target opening height set by user/rule. If the user has specified a new gate opening (by using the rules), but this gate opening has not yet been reached because of constraints in how fast the gate can open or close, then this variable will be the new, target opening. If no target position has been specified, then this variable will be the current gate opening height.

Gate - Submergence: (fractional) Gate submergence for the gate group (e.g. 0.88).

Gate - Opening Rate: Gate opening rate for the gate group.

Gate - Closing Rate: Gate closing rate for the gate group.

Lake Superior (Plan 1977A): This get simulation value will determine the nominal monthly outflow for Lake Superior as specified by the Plan 1977A regulations. This computation is based on the value of user defined variables that must be in a specific order. The first twelve rule operations (excluding comment lines) must be defined in the order show in figure 14-41.

Storage Areas variables

WS Elevation: Water surface elevation for the given storage area.

Net Inflow: Net inflow for the given storage area (e.g. Total Inflow - Total Outflow).

Total Inflow: Total inflow for the given storage area (gross inflow, ignores outflow).

Total Outflow: Total outflow for the given storage area (gross outflow, ignores inflow).

Area: Current surface area of storage area.

Volume: Current volume of storage area.

Pump Stations variables:

Station – Pump Flow: Total pump flow for the pump station.

Station – Pump Flow (transition complete): What the total pump flow would be (at current heads) when all pumps have finished ramping on or ramping off. If there are no pumps that are currently ramping on or off, then this variable will equal Station – Pump Flow.

Station – Pump Flow Maximum: Set a maximum flow for the pumping station.

Station – Pump Flow Minimum: Set a minimum flow for the pumping station.

Station – Turn All Pumps On: Starts turning all the pumps on.

Station – Turn All Pumps Off: Starts turning all the pumps off.

Station – WSEL Inlet: Water surface at pump inlet.

Station – WSEL Outlet: Water surface at pump outlet.

Station – WSEL Reference: Water surface at the reference node (as defined on the Geometry Pump Station Data editor).

Group – Pump Flow: Total pump flow for the pump group.

Group – Pump Flow (transition complete): What the total pump flow would be (at current heads) when all pumps in this group have finished ramping on or ramping off. If there are no pumps that are currently ramping on or off, then this variable will equal Group – Pump Flow.

Group – Pump Flow Maximum: Set a maximum flow for the pump group.

Group – Startup Time: Sets the time to ramp on the pumps in this group.

Group – Shutdown Time: Sets the time to ramp off the pumps in this group.

Group – Flow Factor: Sets a multiplier for the pump flow in this group.

Group – Head Maximum: Sets a maximum head this group will pump against.

Group – Head Minimum: Sets a minimum head this group will pump against.

Group – Additional Head: Sets an additional head this group will pump against.

Group – Turn All Pumps On: Starts turning all the pumps in this group on.

Group – Turn All Pumps Off: Starts turning all the pumps in this group off.

Group – Number of Pumps On: The number of pumps in this group that are on. Any pumps in the process of ramping off will not be included.

Pumps On (fraction): A decimal fraction that represent whether this pump is on or off or transitioning. This value is 0.0 when the pump is fully off, it is 1.0 when fully on, and it is a fraction between 0.0 and 1.0 when it is ramping on or off.

Pump – Flow: Pump flow.

Pump – Flow (transition complete): What the pump flow would be (at current heads) when this pump has finished ramping on or ramping off. If the pump is not currently ramping on or off, then this variable will equal Pump – Flow.

Pump – Flow Maximum: Set a maximum flow for the pump.

Pump – Flow Minimum: Set a minimum flow for the pump.

Turn Pump On: Starts turning the pump on.

Turn Pump Off: Starts turning the pump off.

Reactivate WSEL On and WSEL Off mode: This will make the WSEL On and Off mode activate for this pump.

WSEL On: Sets the water surface elevation to turn this pump on (when the WSEL On and Off mode is active for this pump).

WSEL Off: Sets the water surface elevation to turn this pump off (when the WSEL On and Off mode is active for this pump).

## Automated Calibration of Manning's n Values for Unsteady Flow

In order to assist engineers in the calibration of unsteady flow models, we have added an automated Manning's n value calibration feature. This Option requires observed stage time series data (flow time series is optional) in order to be used. Manning's n values are calibrated on a reach basis. One or more reaches can be calibrated within the same run. Hydrographs are broken into flow zones from low to high in order to allow Manning's n values to vary with flow rate. The results from using this feature are a set of flow versus roughness factors for each reach that it is applied too.

This automated calibration feature can be applied in either a "Global" or "Sequential" optimization mode. The "Global" mode optimizes Manning's n values for all reaches at the same time. While the "Sequential" mode optimizes Manning's n values for one reach at a time, working from upstream to downstream. Manning's n values are optimized (adjusted) for each flow zone until the maximum flow zone error is less than a user entered tolerance, or until a maximum number of iterations is reached.

To better understand how this works; let's look at the optimization procedure for a single reach, as shown in Figure 14-85.

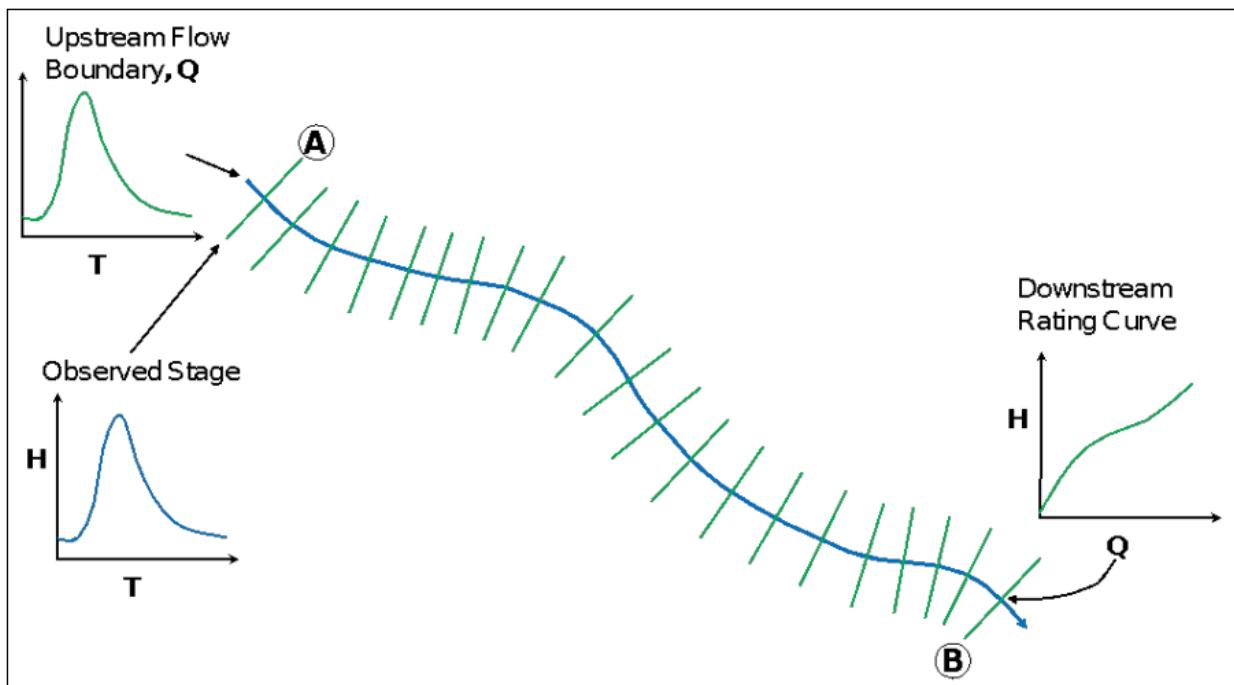


Figure 14.85. Single Reach Example

For the single reach example shown in Figure 14-85 above, there is a flow hydrograph introduced as the upstream boundary condition and a rating curve used as the downstream boundary for the reach. An observed stage hydrograph is attached to the model at the most upstream cross section (labeled as location A). Location A is where the software will compare the computed stages to the observed stages, in order to make decisions on how to change the Manning's n values for the reach. The user must set up a flow versus roughness factor table for the reach to be calibrated (optimized). For this example, let's assume we will have the entire reach set up as a single flow versus roughness factor reach. The user will enter flows versus roughness values into a table. The hydrograph is broken into flow zones from low to high in order to allow Manning's n values to vary with flow rate (Figure 14-86). The number of flow values entered in the flow versus roughness factor table will dictate how the hydrograph is broken up into flow zones for that reach. The roughness factors can be entered simply as values of 1.0 to start out, which assumes no change to the base Manning's n values. The optimization process will adjust the flow roughness factors for each flow zone in order to improve the model calibration.

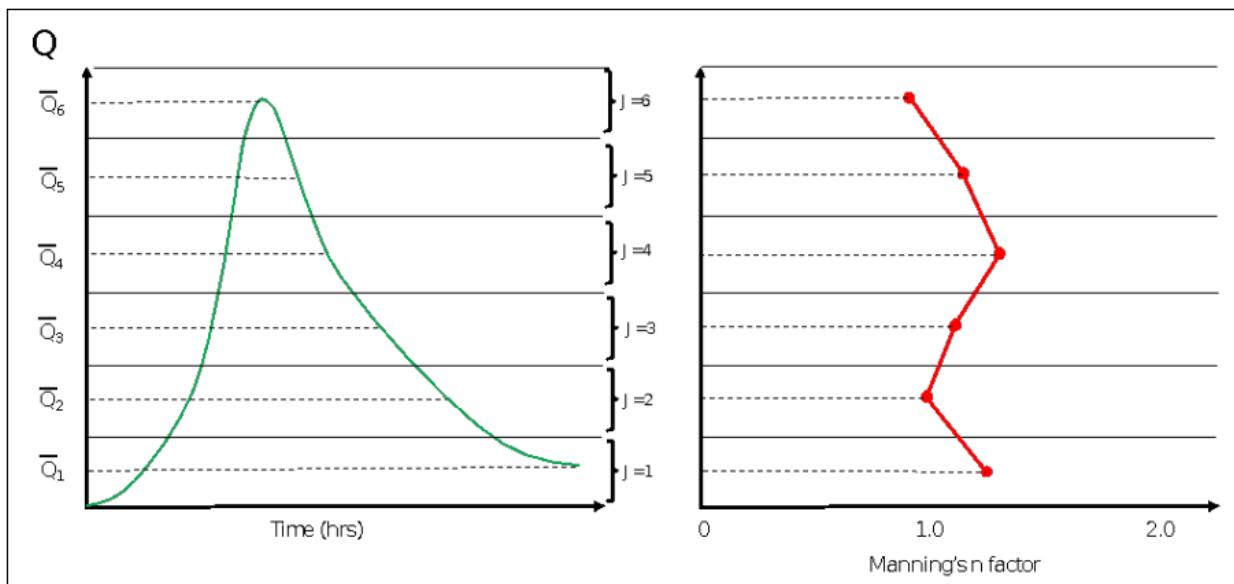


Figure 14-86. Flow Zones Versus Manning's roughness factors.

As shown in Figure 14-86, the hydrograph for this example is broken into 6 flow zones (this is controlled by the user). The software first runs the entire analysis just using the base Manning's  $n$  values (those entered by the user in the Geometric data editor). Then the computed versus observed stages are compared for each flow zone (Figure 14-87). The model adjusts roughness for each flow zone depending on the magnitude and direction of the error in that flow zone. The product of the optimization run is a set of Flow versus roughness factors that best fits the observed data given the information provided.

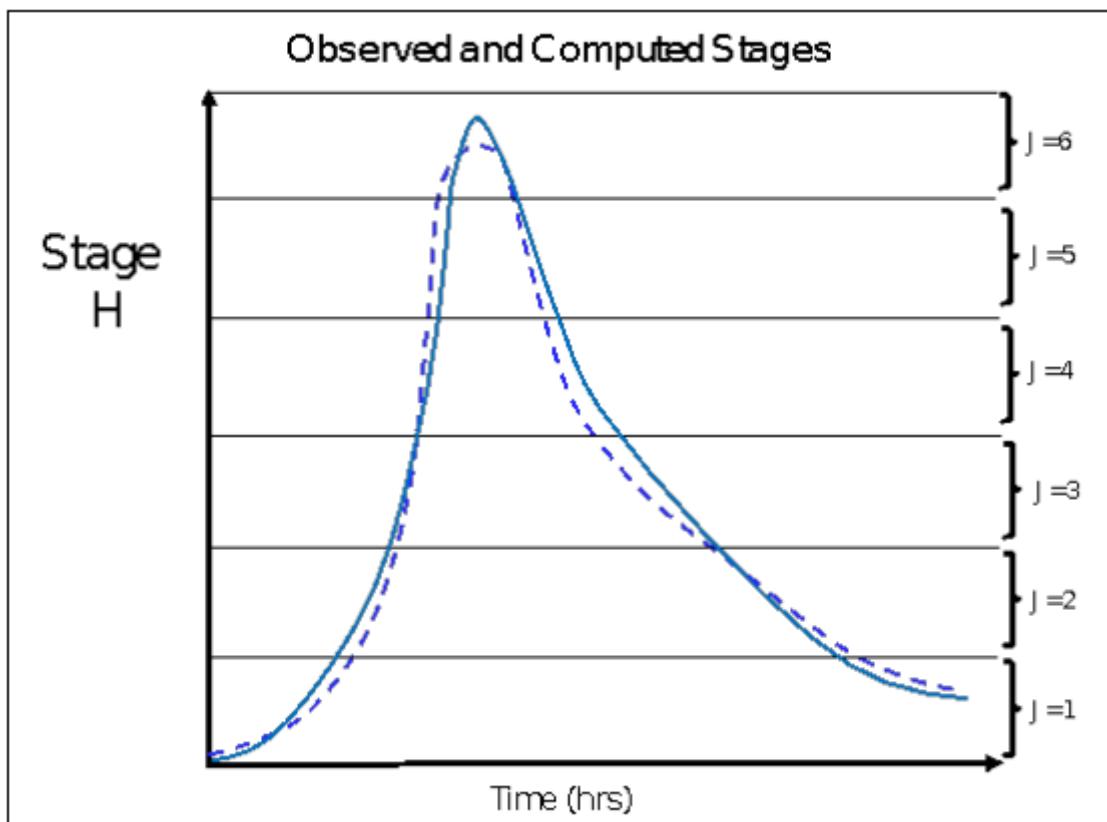


Figure 14-87. Computed and observed stages broken into flow zones for comparison.

### Using the Automated Calibration Feature

To use this feature, users must first enter an initial set of Manning's n values for all of the cross sections in the model. The initial set of Manning's n values should be a reasonable estimate of the main channel and overbank areas based on land use, Ariel photography, and knowledge of the river in general.

The second step in using the automated Manning's n value option is to break up the river system into logical calibration reaches in which a set of flow versus roughness values can be applied. Flow versus roughness factors can be applied to an entire river reach, or multiple flow versus roughness factor sets can be set up within a single river reach. However, each flow versus roughness factor calibration reach will need to be assigned an observed stage hydrograph in order for the automated procedure to perform a comparison of computed versus observed values. An example of how one could break up a river system for this purpose is shown in Figure 14-88.

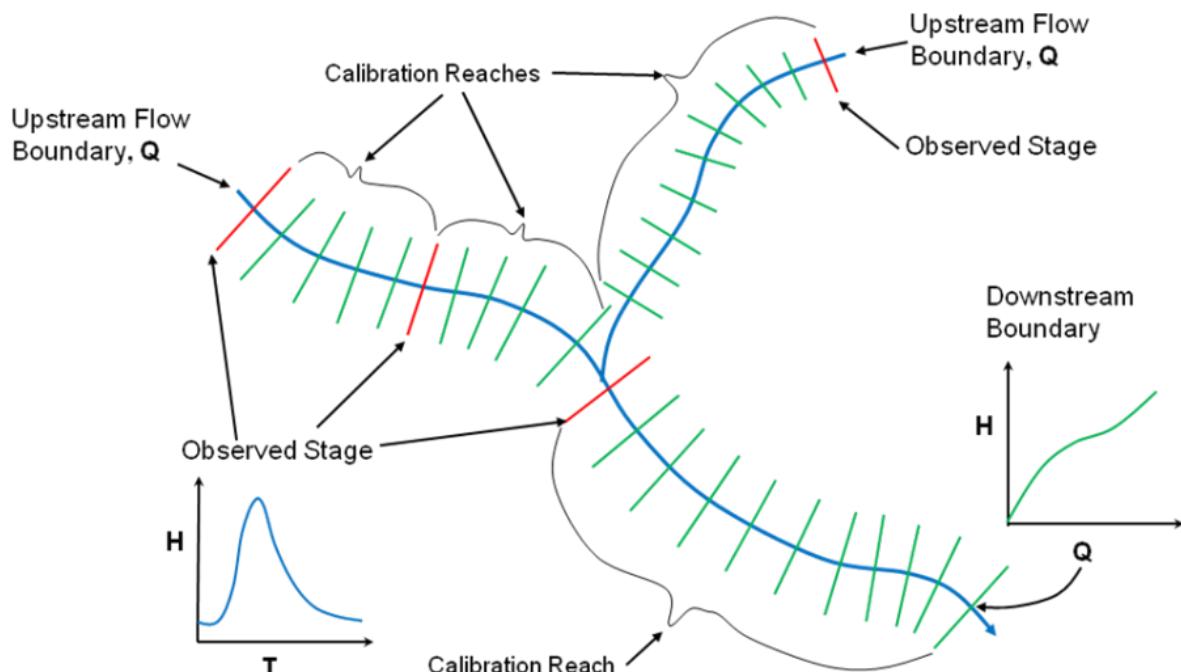


Figure 14-88. Example Calibration Reach Layout.

As shown in Figure 14-88, each calibration reach must have an observed hydrograph location to compare against. It is not an absolute requirement that the observed hydrograph be at the very upstream end of each calibration reach, but it generally makes sense to do it that way, as Manning's  $n$  value changes that occur downstream will affect water surface elevation upstream. However, the observed hydrograph locations can be anywhere (even outside of that particular calibration reach), as long as the roughness changes that occur within the calibration reach will directly impact the computed water surface elevations at the observed hydrograph location.

The next step in performing the automated Manning's  $n$  value calibration is to set up a Flow Versus Roughness table for each calibration reach (Figure 14-89). The initial values entered, will be a placeholder for the automated routine to start from. In general, users should enter a range of flows that encompass the entire range of flows that will be experienced for the river system. However, the initial roughness factors can all be set to 1.0 (which means no change from the base Manning's  $n$  values). To get to the Flow Versus Roughness Factor Editor, the user can select it from the Geometric Data Editor under "Tools", then "Flow Roughness Factors". Flow versus roughness factors can also be set up as part of the Plan file by selecting the "Options" menu from the Unsteady Flow Analysis window, then selecting "Flow Roughness Factors". Shown below is the Flow versus roughness editor, with a starting set of factors of 1.0.

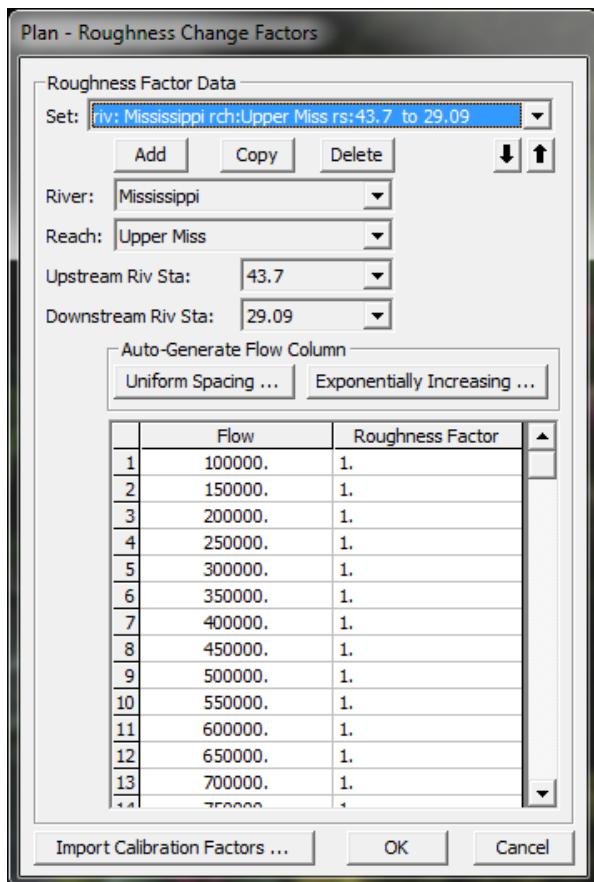


Figure 14 89. Flow versus Roughness Factor Editor.

The next required step to using the Automated Unsteady Flow Manning's n value calibration option is to enter observed stage hydrographs into the Unsteady Flow Data editor (Figure 14-90). Observed stage hydrographs must be assigned to each calibration reach in order to use this automated calibration feature. All observed stage data must be contained within an HEC DSS file and attached to specific cross section location from within the Unsteady Flow Data editor. To attach observed time series data to cross section locations, bring up the Unsteady Flow Data editor and select "Options", then "Observed (Measured) Data, then "Time Series in DSS". The following window will then appear:

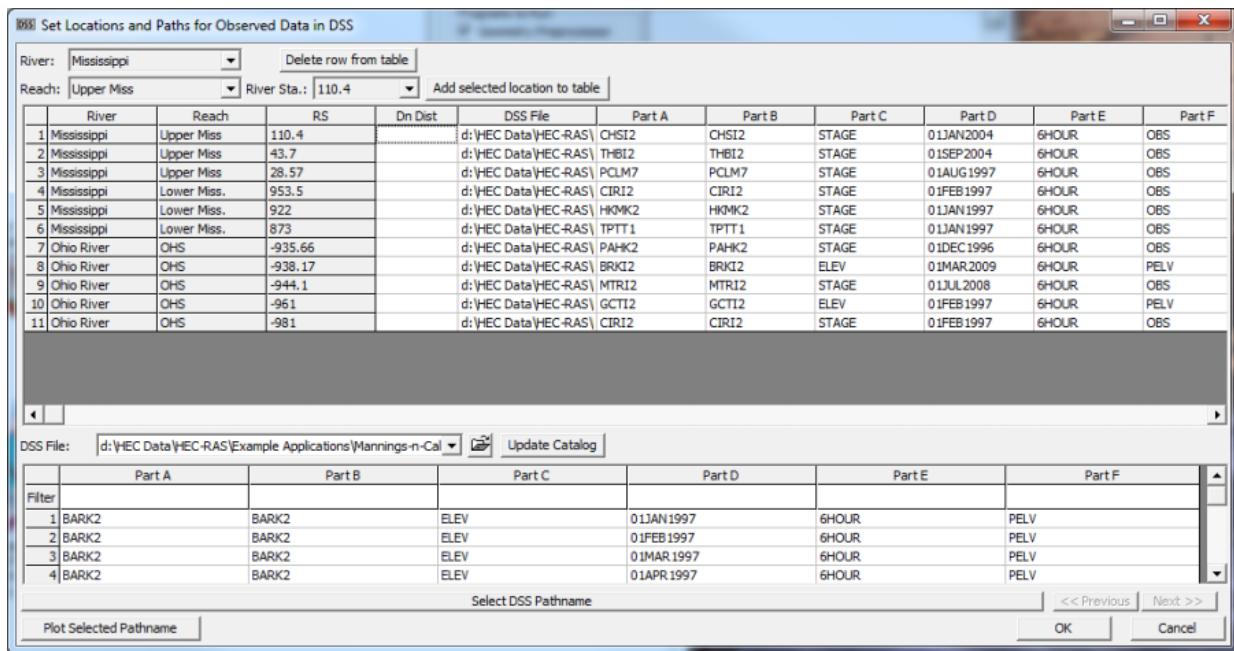


Figure 14 90. Observed Time Series Data Editor

To learn more about how to assign observed time series data to cross section locations, review Chapter 7 of this manual, and the section on Unsteady Flow Data Options.

Once the user has set up flow versus roughness tables (With place holder factors of 1.0), and attached observed stage time series to cross sections, then the Automated Manning's n Value tool can be used. To use this feature select the "Options" menus from the Unsteady Flow Analysis window, then select "Automated Roughness Calibration". The following window will appear:

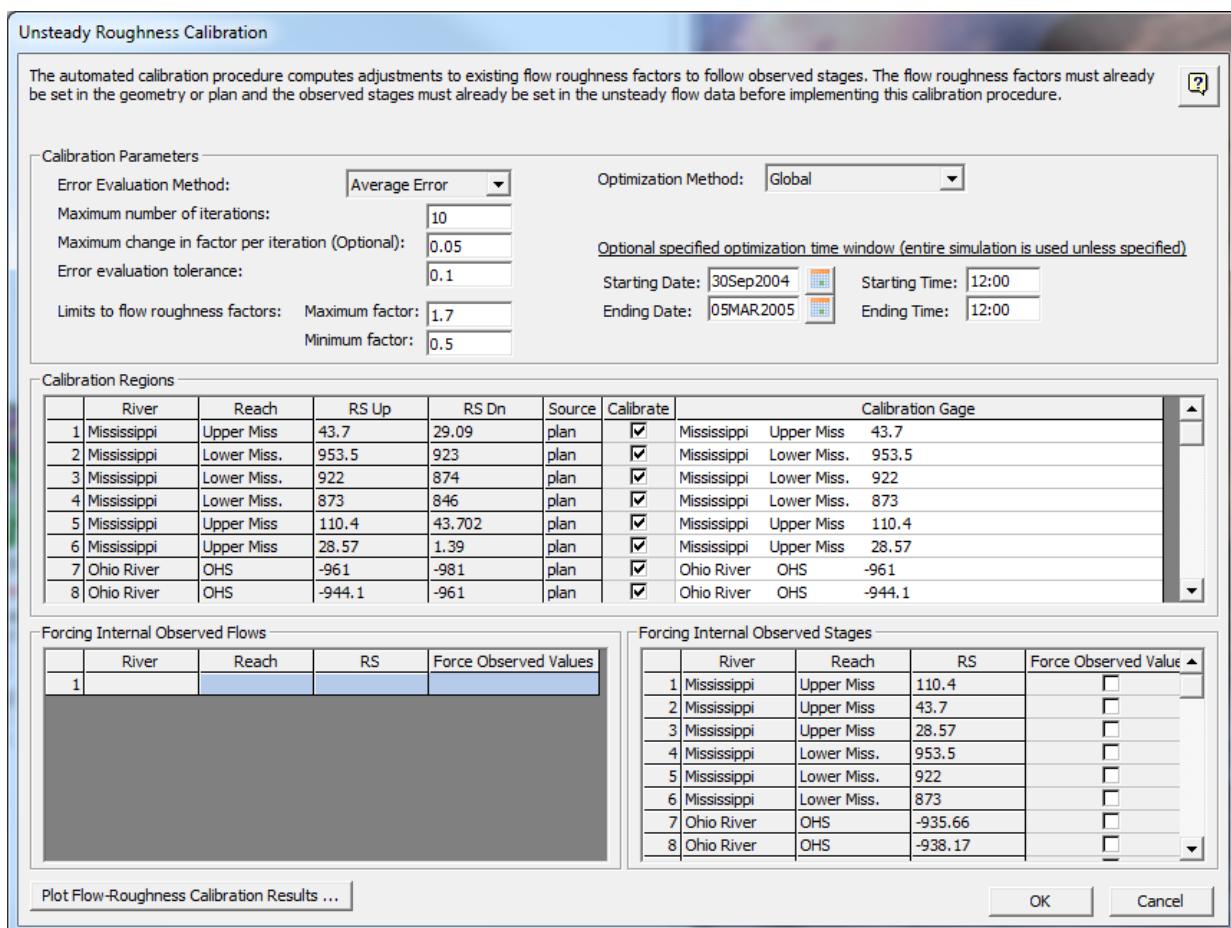


Figure 14.91. Automated Manning's  $n$  Value Editor for Unsteady Flow Analyses.

As shown in Figure 14-91, the Automated Manning's  $n$  Value calibration editor has three areas for data entry. These areas are labeled: Calibration Parameters; Calibration Regions; and Forcing Internal Observed Flows and Stages.

#### Calibration Parameters

The top area, called "**Calibration Parameters**" is required to use this tool. This is where the main information is entered to control the optimization feature. The following information is entered in this part of the editor:

**Error Evaluation Method.** There are two options for the error evaluation method: Average Error (default) and Squared Error. The error is computed separately for each flow band (ranges of flow rates) within the hydrograph. The Average Error is computed by taking the sum of the difference between the computed and observed water surface elevations (for all points within the optimization time window), and then dividing by the number of points. The Average Error equation is as follows:

$$\text{Average Error} = \frac{1}{n} \sum_1^n \text{Comp. WS} - \text{Obs. WS}$$

The squared error method is computed by taking the sum of the computed minus the observed water surface elevations, squaring that value, dividing by the number of points, then taking the square root. The sign of the error (positive or negative) is kept track of separately in order to decide if the n values should be increased or decreased. The use of the Squared Error method will put more weight on the points that have larger differences. The Squared Error equation is as follows:

$$\text{Squared Error} = \sqrt{\frac{1}{n} \sum_{1}^n (\text{Comp. WS} - \text{Obs. WS})^2}$$

**Optimization Method.** There are two optimization methods available within the HEC-RAS Unsteady Flow Roughness Calibration methodology: **Global** (this is the default); and **Sequential**.

The **Global method** optimizes all calibration reaches simultaneously. Manning's n values are modified for all of the calibration reaches by adjusting the flow versus roughness values for each flow band. A full unsteady flow simulation is performed for the entire model for each iteration. The optimization process continues until either the convergence criteria is met or the maximum number of iterations is reached. This method is the preferred method. This is due to the fact that downstream stage changes will affect upstream stages, and upstream flow routing is affected by changes in roughness. So a simultaneous optimization of all reaches will often produce better results for the Manning's n values.

The **Sequential method** optimizes calibration reaches one at a time from upstream to downstream. This method requires more observed data, in that the user must have an observed stage hydrograph at the downstream end of each calibration reach to be used as a downstream boundary condition. This is in addition to the observed stage at the upstream end, which is used for comparing observed and computed stages (see Figure 14-92). A computed or an observed flow can be used at the upstream boundary condition for each reach. The optimization is performed in its entirety for a single upstream reach, then the 2<sup>nd</sup> and subsequent reaches are optimized separately, until all reaches are optimized. While this process does a good job at isolating each reach from any downstream influences/errors (this is due to using the observed stage hydrograph as a downstream boundary for each reach), the method takes much more compute time, and can end up with sets of flow versus roughness factors that do not work as well once they are used in a normal simulation mode, without forcing all the stages in the middle of the system.

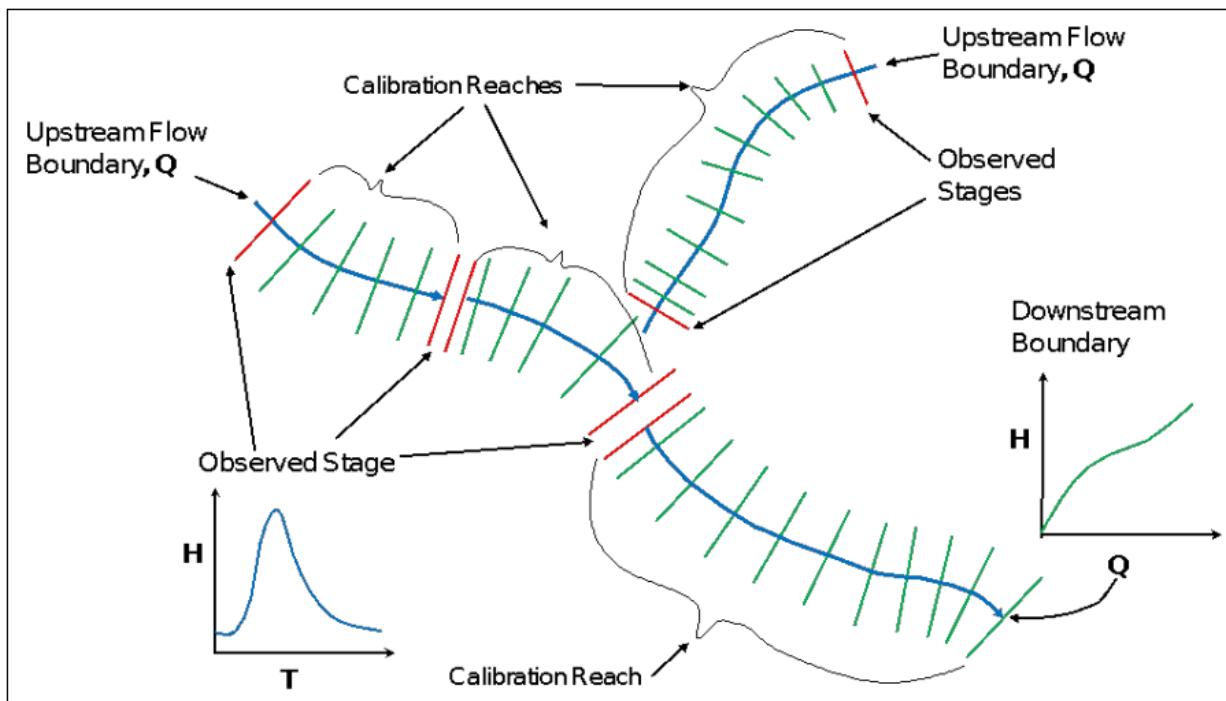


Figure 14.92. Decomposition of Reaches for the Sequential Optimization Method

**Maximum Number of Iterations.** This field is used to set the maximum number of iterations that the optimization process will try in order to adjust the roughness factors to the optimal values. The default value is 10 iterations, however the user can enter anywhere from 1 to 100. For each optimization iteration, the model is run for the entire time window, before evaluating the model error and adjusting the roughness factors. So for example, if a single model run takes 5 minutes, then 10 iterations will take 50 minutes.

**Maximum Change in Factor per iteration (Optional).** This field is used to enter a maximum amount that any roughness factor can change from one iteration to the next. While this field is optional, it can be very useful in ensuring that the optimization method does not make too large of a change at any one location between iterations.

**Error evaluation tolerance.** This field is used to enter the tolerance that is used to compare against the computed error for each flow band. The optimization process will continue until either the maximum flow band error (Average error or squared error) is less than the user entered tolerance in this field, or the maximum number of iterations is reached. If computed water surface elevations are in feet, then this tolerance is in feet. If computed water surface elevations are in meters, then this tolerance is in meters.

**Limits to flow roughness factors (Maximum Factor and Minimum Factor).** These fields are used to enter maximum and minimum values for the flow roughness factors. In other words, during the optimization no flow factor will be allowed to go above the "Maximum Factor", and no roughness factor will be allowed to go below the "Minimum Factor".

**Optional specified optimization time window.** This area is used to enter a starting data and time, and an ending data and time for evaluating the computed versus observed data. This does not change the actual computation window. It only changes the time window in which computed versus observed flows will be compared for computing the model error and evaluating changes in

roughness factors. This is very useful for windowing in around the main hydrograph, such that the error in computed versus observed data is only evaluated in the important region of the event, and not during times of insignificant results. Very often in unsteady flow modeling, the very beginning of the simulation can be off by quite a bit due to bad starting conditions. So not including this beginning model warm-up period can often produce better optimization results.

#### Calibration Regions

The second area, labeled "**Calibration Regions**" displays all of the locations in which flow versus roughness tables have been set up on a reach basis (see Figure 14-91). These are the regions (user define flow versus roughness reaches) in which the calibration option can be applied to. The user can turn on any one region, combinations of regions, or all of the regions, by checking the **Calibrate** column for that region. Additionally, this table is used to assign an observed gage location (Observed stage data) to a calibration region. **Note: Assigning an observed stage hydrograph to each calibration region (calibration reach) is required in order to perform the optimization process.**

#### Forcing Internal Observed Flows and Stages

The third area of the editor, labeled "Forced Internal Observed Flows" and "Forced Internal Observed Stages" is only used if the **Sequential** optimization method is being applied (Figure 14-91). If the "Sequential" Optimization method is turned on, then the user **must** turn on the options to "Force Internal Observed Stages" at the downstream end of every Calibration Region (calibration reach). See the example in Figure 14-92, that shows how a model would be broken into reaches when the "Sequential" optimization method is used. The table labeled "Forcing Internal Observed Flows" is an optional input. This is an optional method, and it requires observed flow time series in addition to the observed stage time series.

For more information on how to use the automated roughness calibration feature for unsteady flow modeling, please see example 24 in the HEC-RAS Applications Guide.