

Making Your HEC-RAS Model Run Faster

Introduction

The purpose of this document is to provide guidance on model modifications that can be made to an HEC-RAS unsteady flow model (1D and/or 2D) in order to improve computational speed, while maintaining model accuracy. Some of the subjects in this document are specific to 1D modeling, some are specific to 2D modeling, and others will affect both types of modeling. The following list are the items contained within this document, and what the user should focus on to improve computational speed and reduce run times:

1. 1D Model Stability (this is the #1 problem with most models)
2. 2D Flow Area Stability
3. Dam and Levee Breaching
4. Boundary Conditions
5. Variable Time Step
6. Too Many Points in Structure Station/Elevation Definitions
7. Too Many 2D cells
8. Pardiso Solver for Large Complex 1D Models
9. Re-Ordering the Reaches for Large Complex 1D Models

Each of these issues are discussed in detail below. Any or all of these issues could be causing your model to be computationally inefficient and the cause of long run times. Working on these issues will improve computational efficiency and reduce model runs times.

1D Model Stability

Model stability problems are often the number one source of long model run times. Improving model stability can dramatically reduce the computational time to run a model. For example, if you have a model that is going to the maximum number of iterations (or even a high number of iterations) for a significant amount of time steps, even if the errors are small, this is significantly slowing down the model computations. Improving the model in the areas where it is having trouble solving the equations (cross sections, 2D cells, storage areas, or hydraulic structures) can dramatically speed up the computations and reduce run times.

Chapter 8 of the HEC-RAS User's Manual has a very detailed discussion on model stability issues for 1D models. If you are having 1D modeling issues, please review that section of the User's Manual very closely, in addition to what is discussed in this document. This document will discuss some of the most common model stability problems that arise for both

1D and 2D models which end up increasing run times. If your model is repeatedly going to the maximum number of iterations, please review these items to help reduce the number of iterations being used in the model.

Cross Section Spacing, Cell size, and Time step

1D cross section spacing, velocity of the flood wave, and computational time step go hand in hand. If you have too large of a time step for the cross section spacing and the flood wave velocity, the model will iterate more. Shown in Figure 1 is a hydrograph plot for an unstable 1D model with too larger of a time step for the cross section spacing and flood wave speed being modelled.

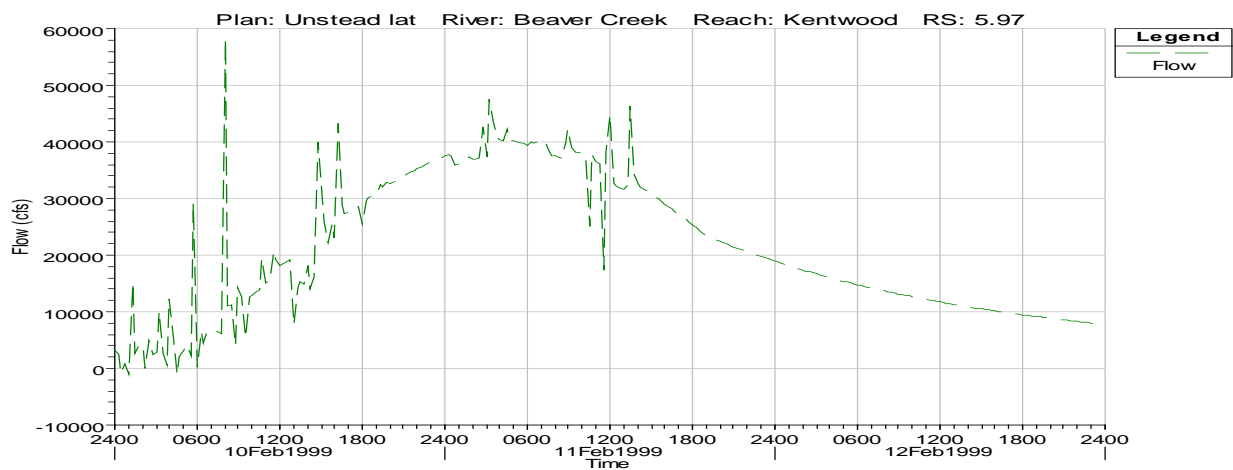


Figure 1. Hydrograph plot for unstable 1D model.

The same thing is true for 2D models. In general, if the time step is too large for the cell size and floodwave speeds, then the model will iterate more or possibly go unstable. Sometimes reducing the time step can speed up the model, even though it has to solve for more time steps (that is as long as it improves the model stability).

In general users should follow the Courant condition criteria for selecting a time step. Meaning you should try to keep the Courant number around 1.0 or lower during the run, if you are having model stability issues. The general Courant condition is:

$$C_r = V_w \frac{\Delta t}{\Delta x} \leq 1.0$$

And therefore:

$$\Delta t \leq \frac{\Delta x}{V_w}$$

Where: V_w = Flood wave speed, which is normally greater than the average velocity.

C_r = Courant Number. A value = 1.0 is optimal.

Δx = Average distance between cross sections, or average cell spacing.

Δt = Computational time step.

If a model is well structured and stable, users can get away with selecting a larger time step that will produce Courant numbers greater than one, as long as the model is stable and still producing accurate answers. Additionally, if the flood wave rises slowly and falls slowly, then larger time steps can be used because the water surface, depth, and velocity are changing slowly with respect to time. Cross section spacing, cell size, and the time step should be tested for sensitivity, stability, and accuracy.

Bad Low Flow Channel Data

Many models do not have good information for the low-flow portion of the channel (i.e. 2-yr flow and lower). Poor channel data, (erratic bed profile, or channel area changing dramatically from one cross section to the next) can cause model instability or significant model iterations at low to moderate flows. It is very common to have missing channel data when using terrain data derived from LiDAR. This is due to the fact that (typically) the LiDAR cannot penetrate through the water to obtain accurate information about the terrain below the water. An example cross section, having the tell-tale “flat” channel bottom, extracted from a LiDAR-based terrain model is shown in Figure 2.

When you have poor low-flow channel data, first try to verify the accuracy of the low flow channel. If it is deemed that the low flow channel is not accurate, consider using the HEC-RAS channel modification routines to cut in a trapezoidal channel that is reasonable, based on all available data sources. Possibilities of source data include: old studies (either HEC-RAS models or even HEC-2 models); FEMA studies; cross sections from old reports; channel invert profiles from older studies; using Google Maps/Earth to make measurements of channel width at the top of the channel and water line; and going out into the field and taking measurements of channel width and depth.

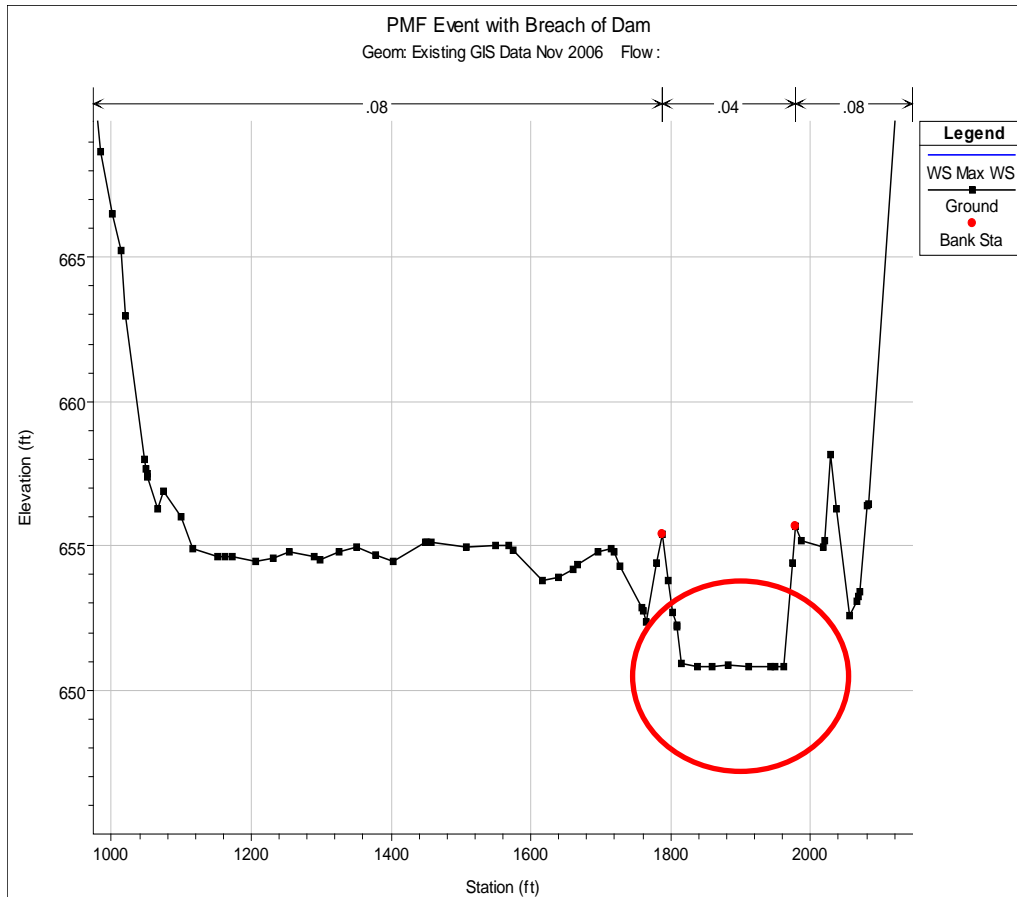


Figure 2. Cross section Extracted from LIDAR based Terrain data.

1D Initial Conditions

The initial conditions for a model are often a source of model stability issues right at the beginning of a run, but can also cause stability problems to persist for some time during the run. For 1D models, this is often a problem with not having the correct flows set up for the 1D reaches, or not having enough base flow. If the initial flows are too low, then pools and riffles will form. Where the model goes from a pool to a riffle, it may rapidly pass through critical depth. This rapid change in water surface, depth, and velocity can cause model instability and/or excessive iterations. To prevent this from happening, increase the initial conditions flows to a high enough flow to drown out these rapid transitions.

An additional problem is that when there are lateral structures connected to the 1D river reaches, and the computed water surface in the river reaches is higher than the lateral structures, the model may end up computing very larger flows going across the lateral structures during the very first actual unsteady-flow time step. This is due to the fact that the program does a steady-flow backwater computation for all of the 1D rivers, for the flows it has been given as the initial conditions. If the computed water surface is higher than the Lateral

Structure, then HEC-RAS wants to send flow over the lateral structure right at the first time step. This problem is made even worse, when the lateral structure is connected to a storage area, and the initial conditions of the storage area is either dry, or at an elevation much lower than the computed water surface in the 1D river on the other side of the Lateral Structure that is connecting them (see Figure 3). The solution to this problem is to either reduce the initial flows, so the computed water surface is below the lateral structures, or set the starting water surface of the storage areas to the same elevation as the water surface in the river on the other side of the Lateral Structure.

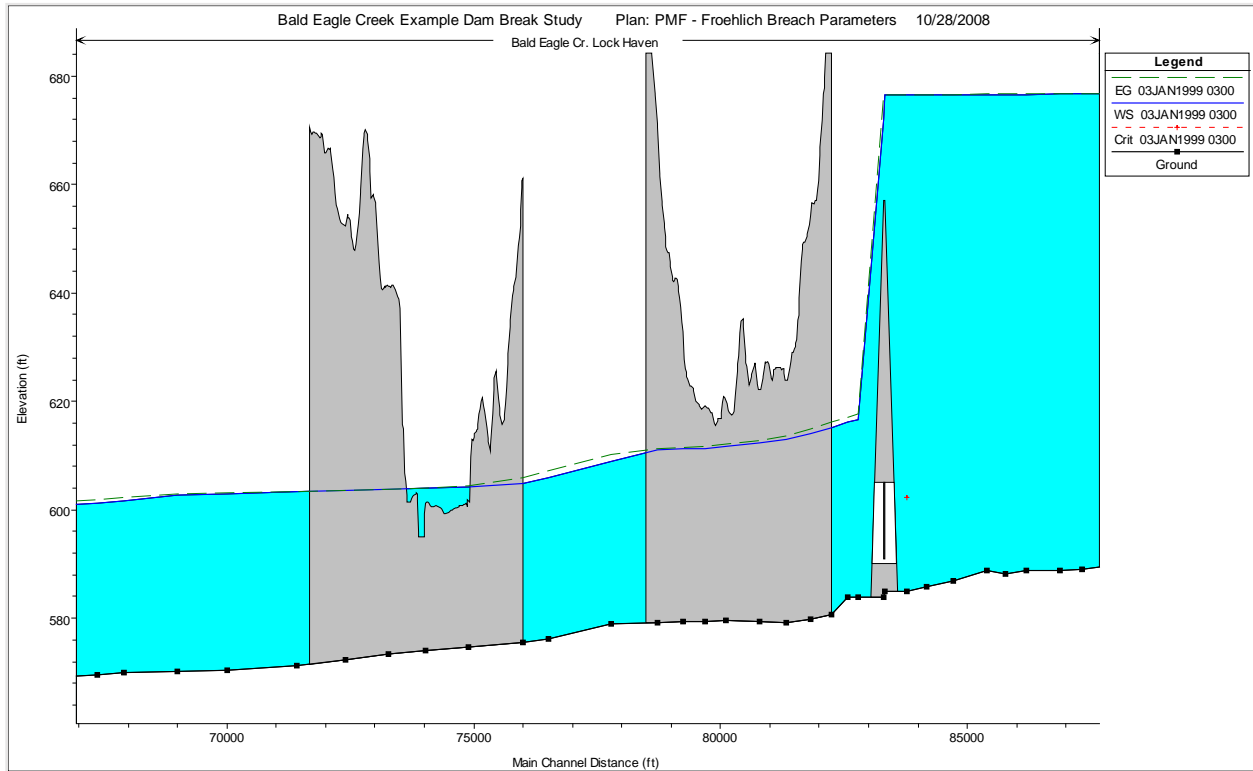


Figure 3. Example of Initial Conditions with Lateral and Inline Structures in the model.

Other things that can help with this problem are to turn on the model warmup option and to create a restart file. You can set model warmup for a specific number of time steps (e.g. 100), and you can also set the computation interval during model warmup to something lower than your base computation interval, which will give the model a better chance of solving when there are many flow and stage discontinuities at the beginning of the simulation. A restart file is the solution to your model for a specific time step that is written to a file which can then be used to set the initial conditions for the entire model. The benefits of a restart file are good stable model conditions that can be used at the beginning of other model runs/events. However, if you change anything in your model, such as adding cross sections, removing cross sections, or adding or deleting 2D cells, then previously created restart files are

no longer valid for that model, and you have to create a new one. Because a change in geometry requires creation of a new restart file, only use restart files where necessary for model stability.

Steep drops and sharp changes in XS Area

Steep drops in the bed profile between two cross sections, or significant changes in flow area between two cross sections can create model stability issues. Significant bed elevation drops from one cross section to the next for low to moderate flows conditions can cause the computed water surface to pass through critical depth (see Figure 4). This extremely sharp change in water surface elevation, depth, velocity, and area can cause the program to iterate a lot and even go unstable.

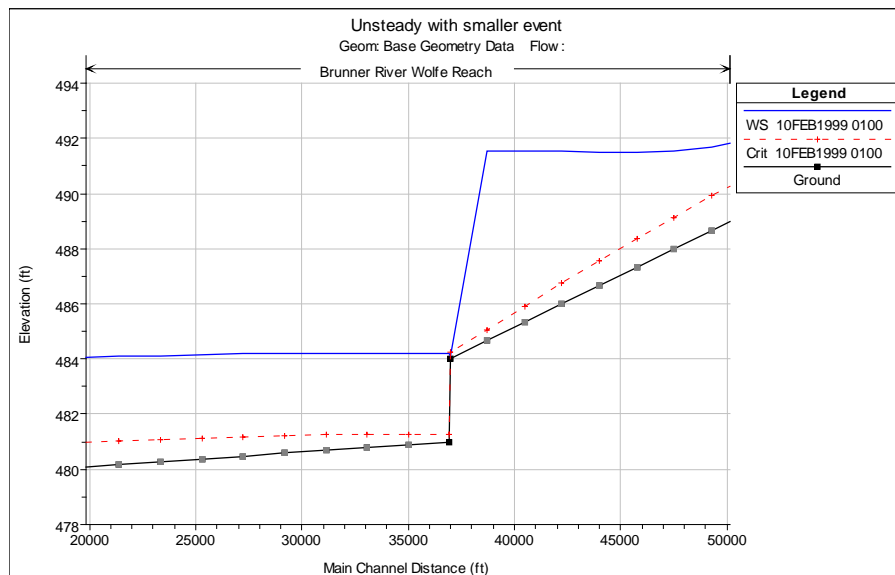


Figure 4. Example of a significant drop in the bed elevation.

Sharp drops in the bed profile should always first be investigated to see if they are real, or just due to poor cross section or terrain information. If the sharp bed does not appear to really exist, then the cross sections and/or the terrain should be modified to smooth out the bed profile. If the sharp drop in elevation is found to be real, then improved modeling approaches should be used. These include: using a higher base flow rate to drown out the drop (if it is only significant during low flows); additional cross section to make the transition more gradual (if the drop is over a long enough distance); modeling the drop with an inline structure between the cross sections (this prevents the unsteady flow equations from being computed over the drop, instead using the weir equation over the structure; the solution is shown in Figure 5); or adding a rating curve to the cross section at the top of the drop (this also prevents

the unsteady flow equations from being computed through the drop, and the rating curve is used to get the water surface for any flow).

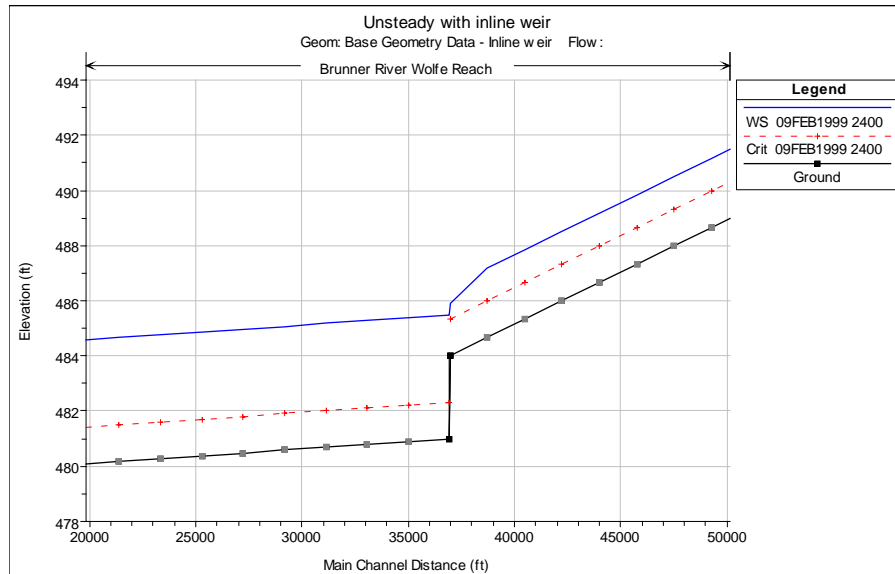


Figure 5. Sharp drop in bed modelled with an Inline Structure.

Extremely sharp changes in flow area from one cross section to the next can also create model instabilities and cause the program to iterate a lot. In general, the solution to this issue is to add more cross sections, if possible, to smooth out the transition over a range of cross sections. However, the addition of more cross sections will most likely require a smaller simulation time step to satisfy the Courant condition.

Ineffective Flow Areas

Ineffective flow areas can be a source of model instability and excessive model iterations. If you have cross sections with wide flat floodplains, and you have ineffective flow areas turned on so those areas do not convey water until a higher water surface elevation is reached, the triggering of these ineffective flows from on to off can cause model stability/iteration issues. This can often occur due to inconsistent trigger elevations from one cross section to the next (i.e. the ineffective area turns off on one cross section but the very next cross section downstream still has the ineffective area turned on). When this happens try to adjust the trigger elevations to be consistent by adjusting them based on the water surface profile in the region (i.e. use the sloping water surface as a guide to how the trigger elevations should vary upstream to downstream).

Another common problem is that when the ineffective flow areas get turned off, the increase conveyance area is so great that the software cannot solve the equations, and the water

surface bounces up and down. Thus the water surface triggers the ineffective flow areas off and on. See an example of this in Figure 6. One way to alleviate water surface instability and bouncing from ineffective areas is to put higher Manning's n values in the portions of the cross sections that have ineffective flow areas. This will make the change in conveyance smaller when the ineffective areas turn off and water begins to convey.

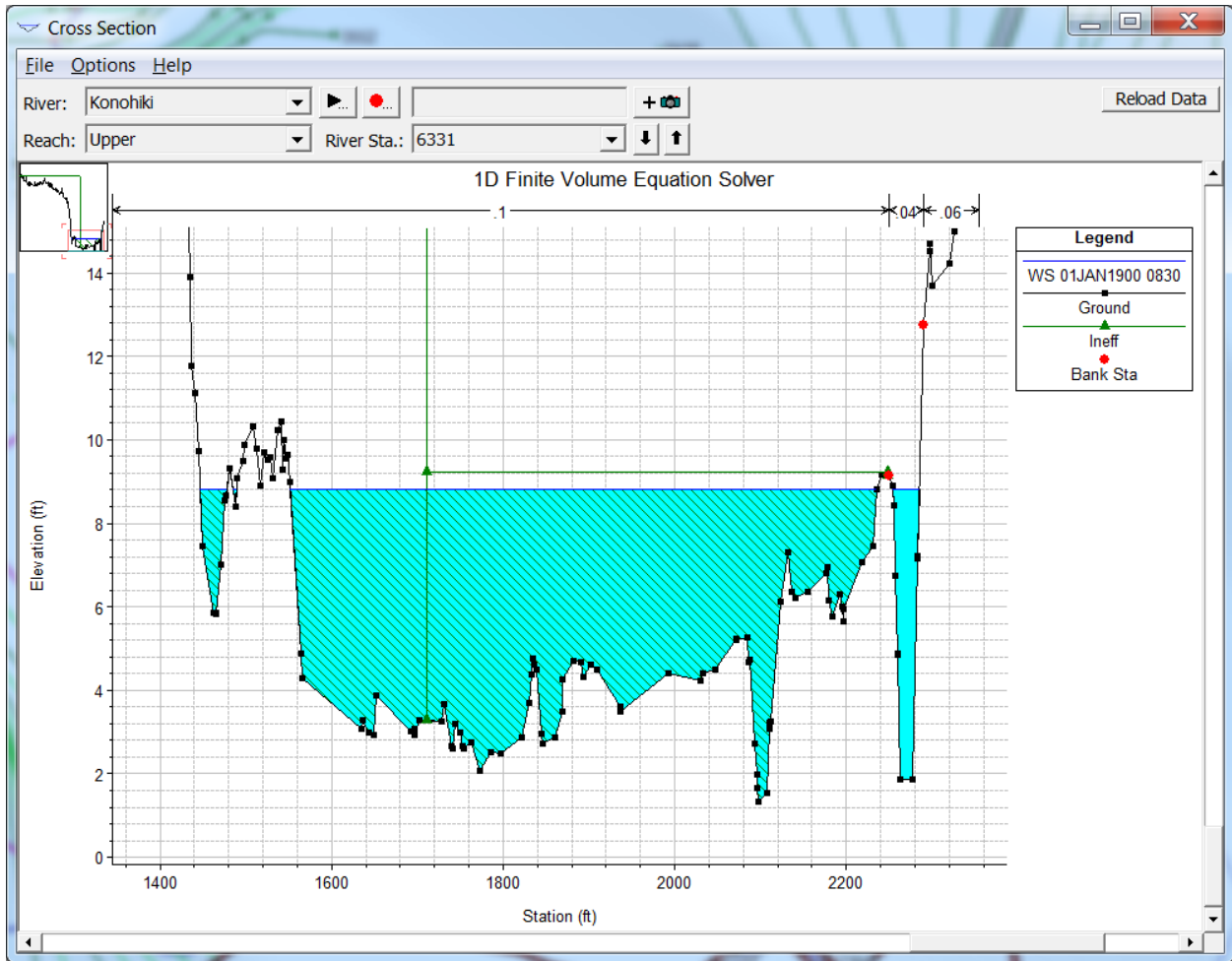


Figure 6. Example Ineffective flow area that controls a large portion of the cross section.

Ineffective flow areas can also be an issue around bridges, the same issues as described above exist. Inconsistency as to when ineffective flow areas turn off and on from upstream of the bridge to downstream can be a problem, as well as adding too much conveyance to the cross section when they turn off. Using higher Manning's n values is the best approach around

bridges for this issue. An example of using high Manning's n values in the overbank areas where the ineffective flow areas are present is shown in Figure 7.

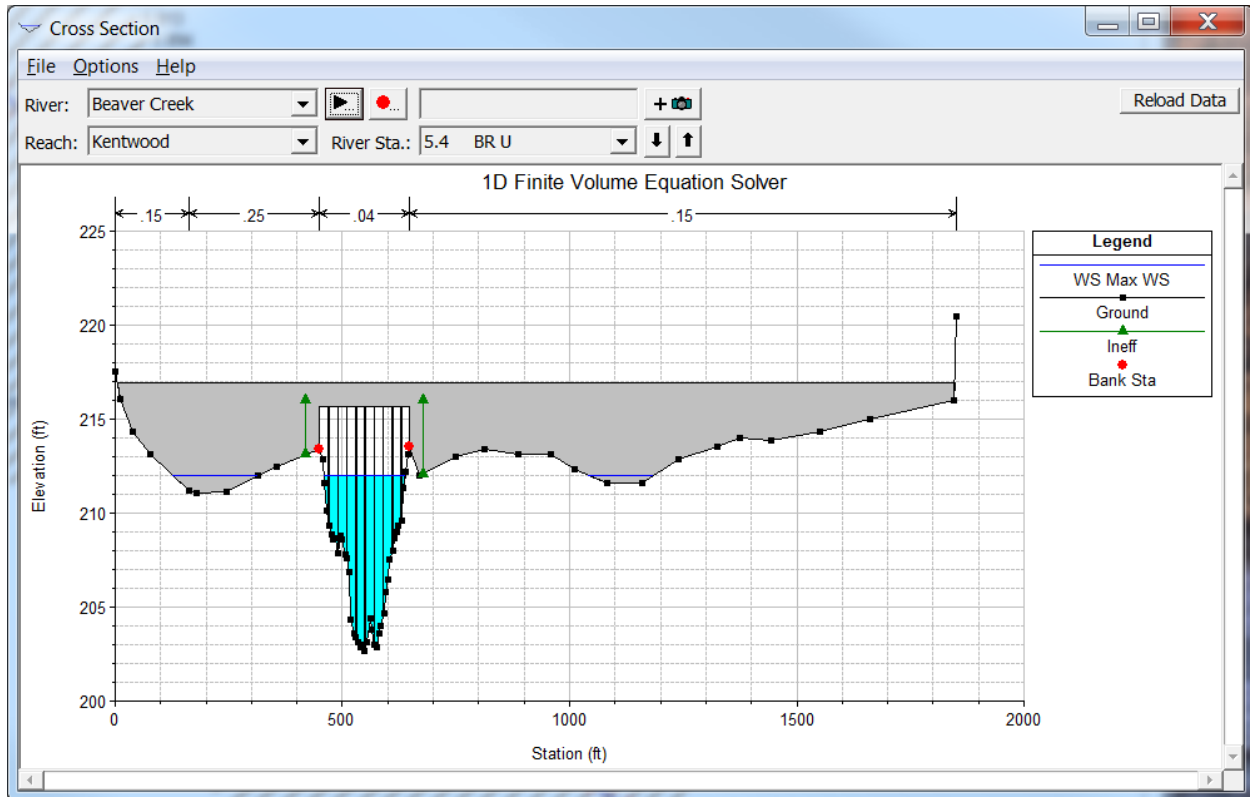


Figure 7. Bridge with Ineffective Flow Areas and High Manning's n Values.

Hydraulic Structures

Hydraulic structures can be a source of model stability/iteration problems in several ways. For structures like bridges and culverts, the rapid changes in water surface, velocity, and flow area around these structures can cause model stability issues. For other structures like inline, lateral, and SA/2D internal structures, as the weirs, culverts, and gates become highly submerged, small changes in water surface end up being big changes in flow. The high changes in flow causes the model to iterate, and the water surface to go up and down.

To reduce problems for bridges and culverts, try to make sure that the cross section placement is good, and that ineffective flow areas are defined properly. Use high Manning's n values in the overbanks where you have ineffective flow areas define. Also make sure the family of curves (hydraulic property tables) that get created for bridges and culverts are reasonable and transition smoothly throughout the range of expected water surface elevations.

For structures with high submergence, there are options in the unsteady flow computations, under “Computation Options and Settings” that you can set to smooth out structure iteration problems due to submergence. These are the “Weir and Gate Flow Submergence decay exponents”. Turning the decay exponents up to 3.0 will smooth out iteration issues, and make the model run with fewer iterations (see Figure 9).

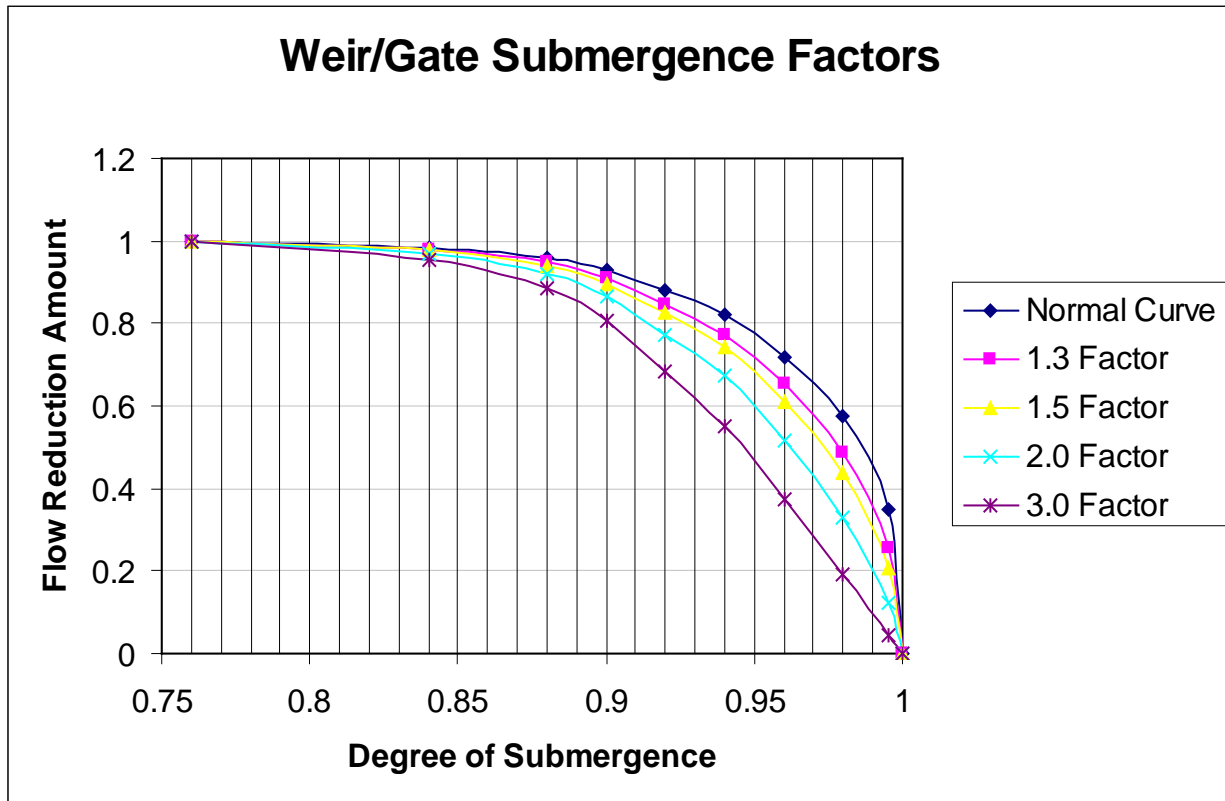


Figure 8. Weir and Gate Flow Submergence Curves.

Storage Areas

Another source of model instability/iteration issues can be storage areas. The general problem is that when a storage area starts dry, and is first starting to fill, the storage elevation-volume curve may be very abrupt and not well defined. This can lead to a lot of model iterations while the storage area is filling to some appreciable depth and a smoother portion of the elevation-volume curve. The solutions to this issue are to first make sure you have well defined/smooth transitioning elevation-volume curves at the low end of the curve (see Figure 9). If there is still a lot of iterations going on at storage areas when they start to fill, then think about modifying the low end of the curve, to make the curve less steep at the very beginning. This is not a big error, as this end of the curve is often only representing very low lying areas and holes in the terrain.

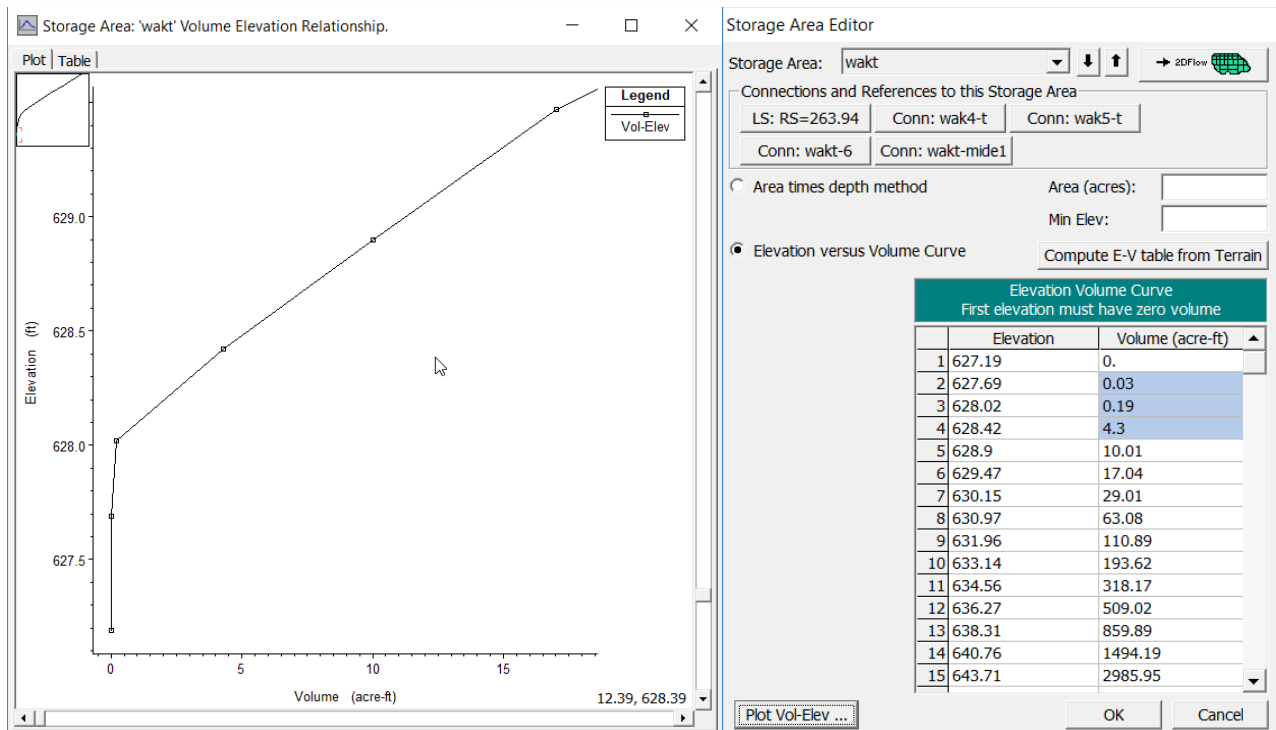


Figure 9. Example Storage Area Elevation-Volume Curve with too few points at lower end of curve.

Steep River Reaches

Steep river reaches can often be a source of model stability and iteration issues. This is most often due to the rapid changes in depth, velocity, area, etc... that occur over very short distances. Steep reaches will generally need more cross sections (spaced at closer intervals) in order for these transitions to be computed more gradually from one cross section to the next. Additionally it is very common for people to under estimate Manning's n values in steep reaches. This leads to even more rapid changes in depth, velocity, and area where the river goes from mild to steep and back to mild. So for the steep regions of a river reach, make sure your Manning's n values are high enough. See Chapter 3 of the Hydraulic Reference manual for more information on Manning's n values for steep streams.

Hydraulic Property Tables

HEC-RAS precomputes hydraulic property tables for all cross sections, bridges and culverts, and weirs. Several different model stability issues can arise when property tables are not well defined. For cross sections you need property tables that go high enough to encompass the entire range of all events, but you also need enough definition in the tables at the low end. If this is not the case, model instability problems and excessive iterations problems can occur at both the low flow end and the high flow end of an event.

For bridges and culverts, users are required to put in a maximum elevation for the curves, and can optionally put in a maximum flow for the curves. Too low of a maximum

elevation may produce stability and iteration problems if the water surface goes above the curves (you will get a warning about this if it happens). Further, if there are not enough points in the curves at the low flow end, and they are not well defined (transition abruptly), model instability issues will result. Therefore, make sure you have a lot of points to define the curves. Another stability issue with bridges and culverts can occur with the Maximum Flow option. The Maximum Flow option is used to stop the HTAB parameter calculations at the specified flow rate. If you route an event that has a higher flow, then the model will have iteration trouble at that flow rate and it will only let that max flow pass through the structure, thus cutting off the hydrograph to that maximum flow. Therefore, verify that none of the hydraulic structures are set with too low of a maximum flow rate (see Figure 10).

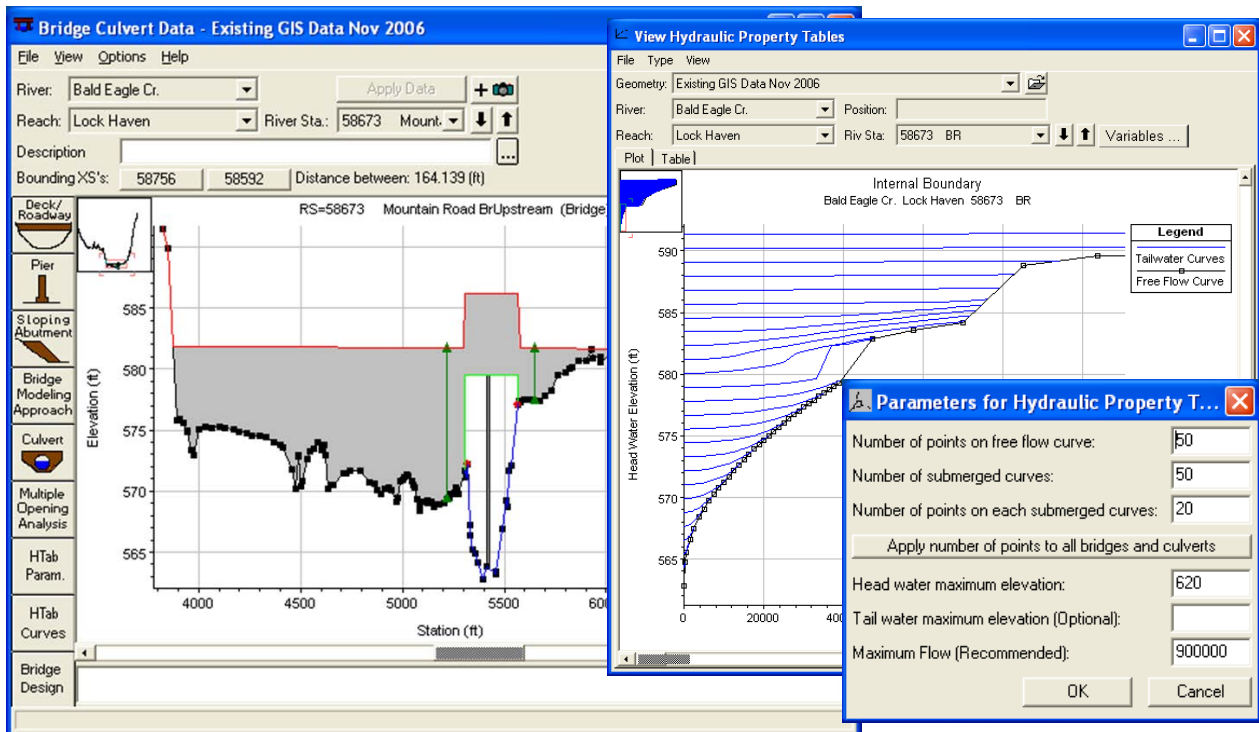


Figure 10. Example Bridge Curves and Property Table Controls.

Cross Sections Too Close Together

Cross sections that are too close together can cause model stability/iteration problems, when a hydrograph is rising rapidly. If the flood wave hits these cross sections that are spaced close together, and the time step for that situation is too large, the model over steepens the leading edge of the flood wave and creates a wall of water at that location. An example of this completely wrong wall of water is shown in Figure 11. The only fix for this is to use a smaller time step, or remove one of the cross sections to make the reach length between cross sections longer, thereby flattening the water surface slope between those two cross sections.

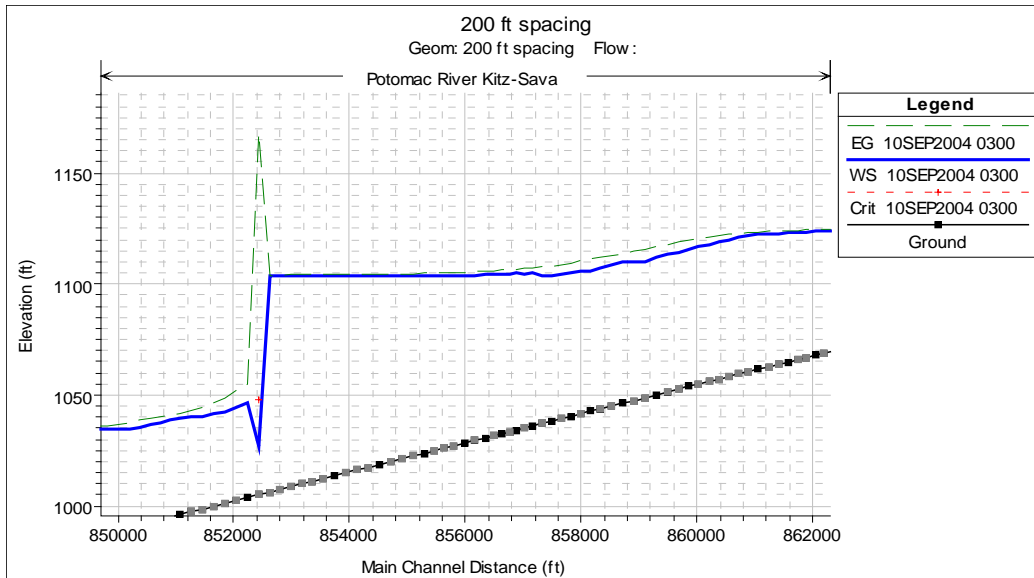


Figure 11. Example of model instability due to Cross Sections too close together

Another cross section spacing issue that causes model instability is based on a legacy limitation of HEC-RAS. In early versions of HEC-RAS, you could not start a lateral structure just below the first cross section of a river reach and you could not end a lateral structure just above the last cross section in a river reach. Lateral structures had to start at the second cross section from the upstream boundary and end at the next to last cross section in the reach. This is no longer the case in RAS, and no longer a limitation. However, users got in the bad habit of placing two cross sections extremely close together at the beginning and end of a river reach. Sometimes even placing those right on top of each other spatially, but with a 1.0 foot reach length between them. This is a very bad practice and it will cause model stability/model iteration issues. Remove those cross sections that are right on top of the first and last section or even those that are very close to the first and last cross section of a river reach.

2D Flow Area Stability

2D Flow Areas can also have several issues that cause excessive iterations and even sometimes model stability issues (the 2D finite volume solver is more stable than the 1D finite difference scheme). A list of items that can cause excessive iterations/model stability issues in 2D Flow Areas are discussed in this section.

Cell Size and Time Step

Just like with the 1D river reaches, if you have too large a time step for the cell size and the flood wave velocities, the model will iterate more and even possibly go unstable. The Diffusion Wave equations are less sensitive to this than the full equations, meaning you can get away with larger time steps for the same mesh resolution/event with Diffusion Wave. However, the Full St. Venant equations are more accurate and may be necessary for your model.

Use the Courant condition guidance discussed earlier, and refer to what is written in the 2D User's Manual, to select an appropriate time step. Your model may actually run faster with a smaller time step, if it is stable, than with a larger time step that is unstable. To assist in understanding the Courant number, users can now add a map layer called "Courant" within RAS Mapper, that can be used to plot the magnitude (by color) of the Courant number at each cell. This map can also be animated in time to see how the courant number is changing throughout the simulation. An example Courant number plot from HEC-RAS mapper is shown in Figure 12.

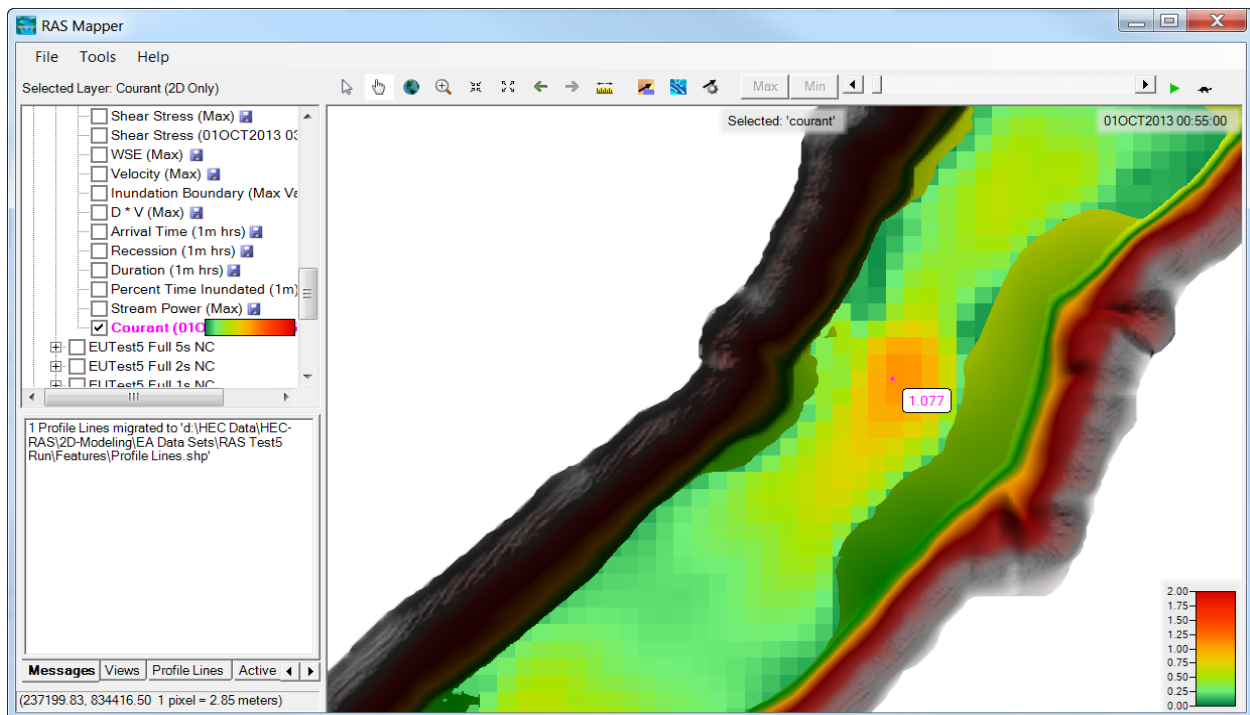


Figure 12. Example Courant Number Spatial Plot from HEC-RAS Mapper.

Using a variable time step, which is new to version 5.0.4 of HEC-RAS, can really help with satisfying the Courant condition. It will allow you to use larger time steps at lower/slower flows, and smaller time steps and faster moving flows. The use of variable time step in HEC-RAS will be discussed in a subsequent section.

2D Initial Conditions

Initial conditions for 2D models can chew up a lot of computation time. So it is absolutely a good idea to create a restart file. Users should run their 2D models (even combined 1D/2D models) until they have a good base flow in the entire system, and then write out a restart file that can be used for all other runs.

2D model initial conditions can be created by using the 2D “Initial Conditions Time” that is contained in the “Computational Options and Tolerances” window, under the 2D Flow Options tab (see Figure 13). Additionally, using the overall model warmup option is a good idea for models that have 1D reaches and/or hydraulic structure connections outside of the 2D Flow Areas.

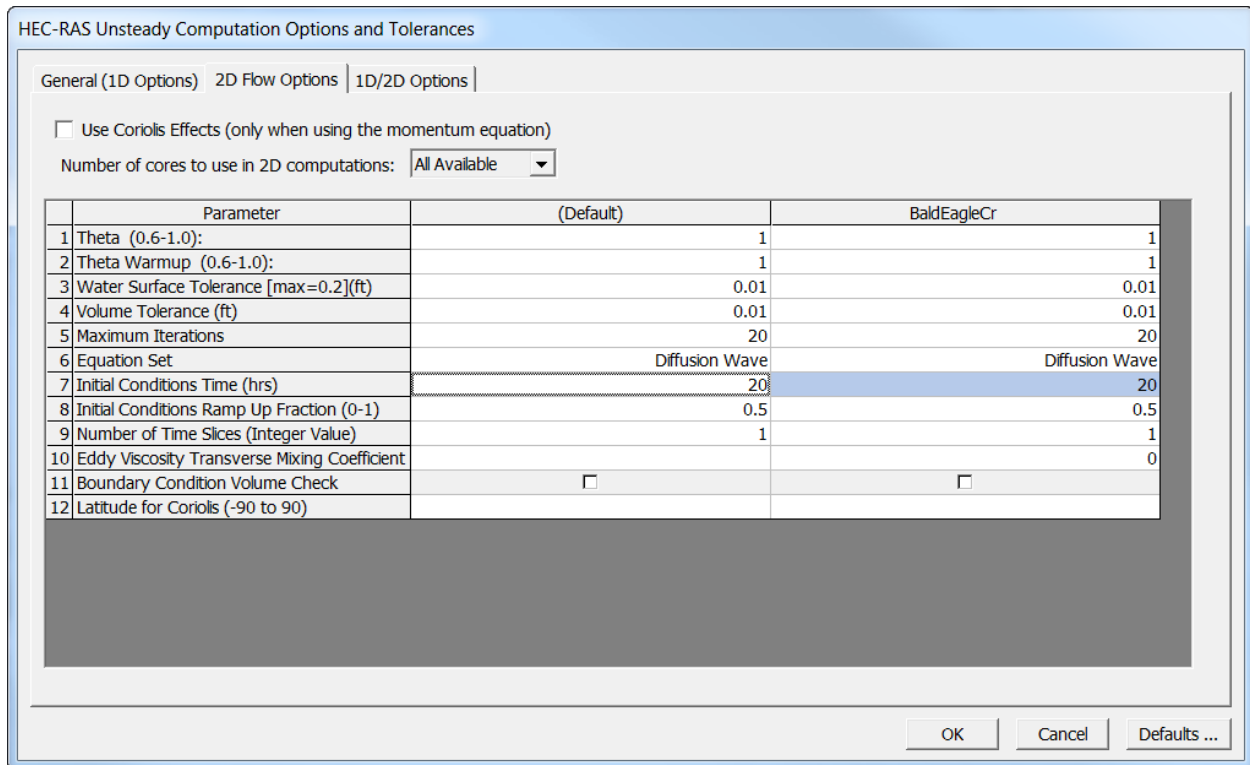


Figure 13. 2D Flow Options with Initial Conditions Time Highlighted.

Flood Wave Wetting Front

When a flood wave is traveling through a dry area, the initial wetting of cells (wetting front) can cause a lot of model iterations. Additionally you may see larger water surface errors when cells first start to get wet. Sometimes this is due to poor cell size selection in that specific area; poor use of breaklines such that the cell faces do not line up with high ground very well; or too large of an elevation change across individual cells. Adjusting cell sizes and using more breaklines to align the cell faces with the high ground barriers to flow can improve model stability with cell wetting.

Weird Shaped Cells/Small Cells

Faces and Cells that either transition too rapidly (small to large or large to small) or that have some faces that are much smaller than the surrounding faces, can cause excessive model iterations and stability issues. Therefore, if your model is going to the maximum iterations a lot for some cells, look at the cells closely. Do the sizes transition smoothly? Are there any very small faces. Minor adjustments in either moving the cell centers, deleting cells, or adding additional cells to smooth out the transitions and remove small faces can get rid of these issues.

Breaklines and 2D Polygon Refinement Tool

In general, users tend to not have enough breaklines to align cell faces where there is high ground barriers to flow in their 2D model. A lack of breaklines will lead to inaccurate models, but can also lead to model stability/iteration issues. To assist users with face alignment and refining the 2D mesh, HEC-RAS version 5.0.4 introduces a new mesh development/modification tool called the **Mesh Refinement Regions** tool. This tool will allow you to draw polygons and specify a new cell size inside of the polygon. Additionally, the tool allows you to specify a separate cell size along the polygon boundary and use the boundary as a breakline to align the cell faces along that boundary. Using more breaklines and the new polygon refinement regions tool will not only improve the accuracy of the mesh/model, but also to make it more stable and iterate less. The new polygon refinement region tool is discussed in more detail below in the section on reducing the number of cells in a 2D model.

Channel Alignment/Cell Size

Another issue with model stability (and accuracy) is to take the time to define the channel portion of the 2D mesh appropriately. The 2D cell faces need to be aligned with the high ground at the sides of the main channel. Therefore, the modeler must locate the edge of the main channel and define it using cell faces. The edge of the main channel is that which separates what is really channel flow versus overbank flow. Define the channel using either breaklines or the new Polygon Refinement tool (this is the author's preference) to define the line of separation from the channel flow to the overbank flow. This will align the cell faces

along this line, which will make the channel portion of your model more accurate, but it will also make the model more stable and iterate less. The results of using the new Polygon Refinement Region tool for defining the main channel cells is shown in Figure 14.

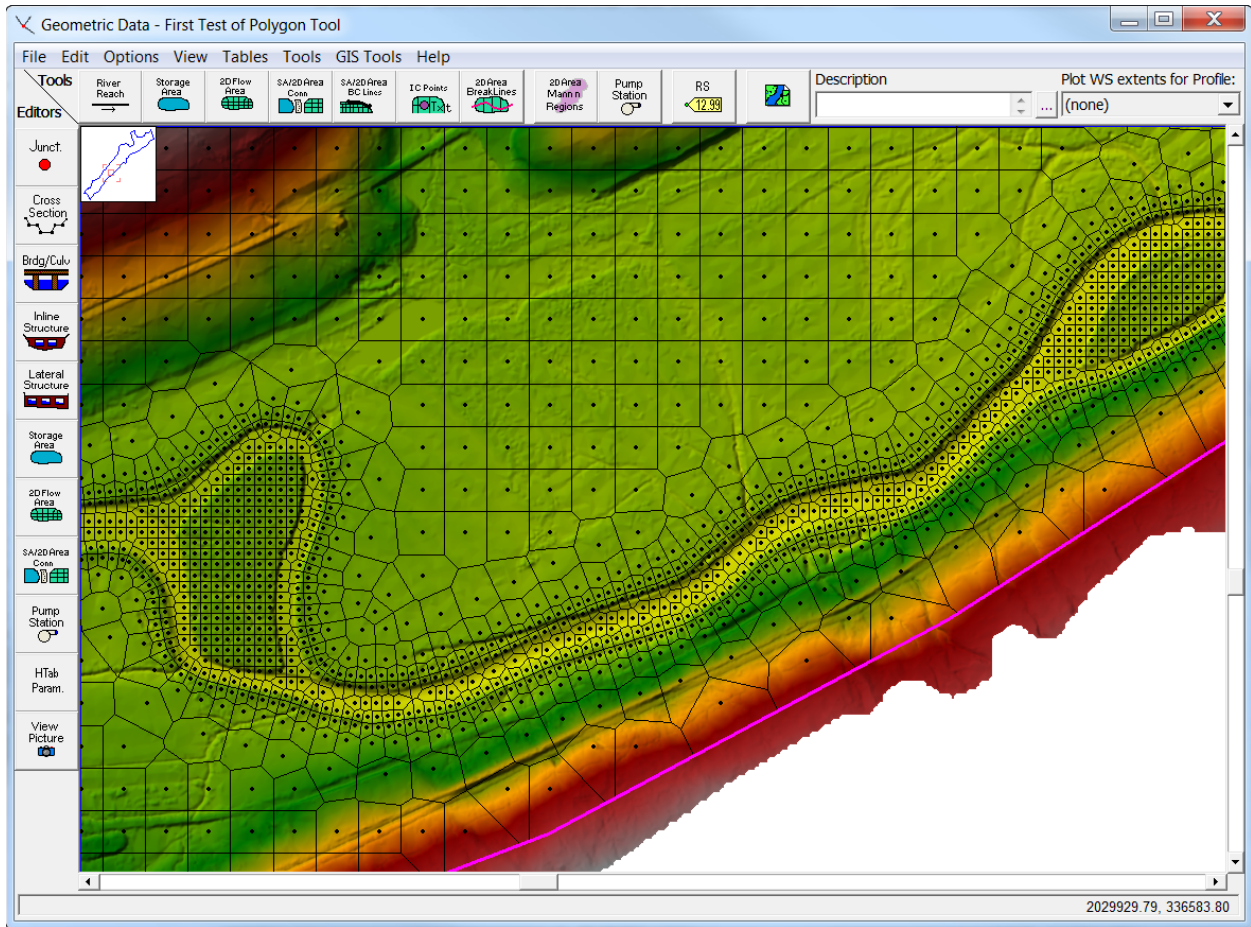


Figure 14. Example mesh with detailed channel cells by using mesh Polygon Refinement Tool.

Partial Cell Wetting

An issue that can sometimes come up and cause excessive model iterations, is when just a corner of a cell has flow going through it at a high velocity. This is made even worse when the flow comes into a small portion of a cell face, but is allowed to leave out of a much larger portion of the cell. Both of these issues makes it very hard for the HEC-RAS to solve the equations and find a water surface elevation that balances volume conservation (inflow – outflow = storage). Therefore, if you have some cells that go to max iterations for a lot of time steps in a row, and when you look at them only a corner (or small portion of the cell) is being used, this is most likely the cause of longer run times. Adjust cell sizes or alignment in that area to get rid of the issue. See Figure 15 for an example of a partially wet cell that will cause iteration problems.

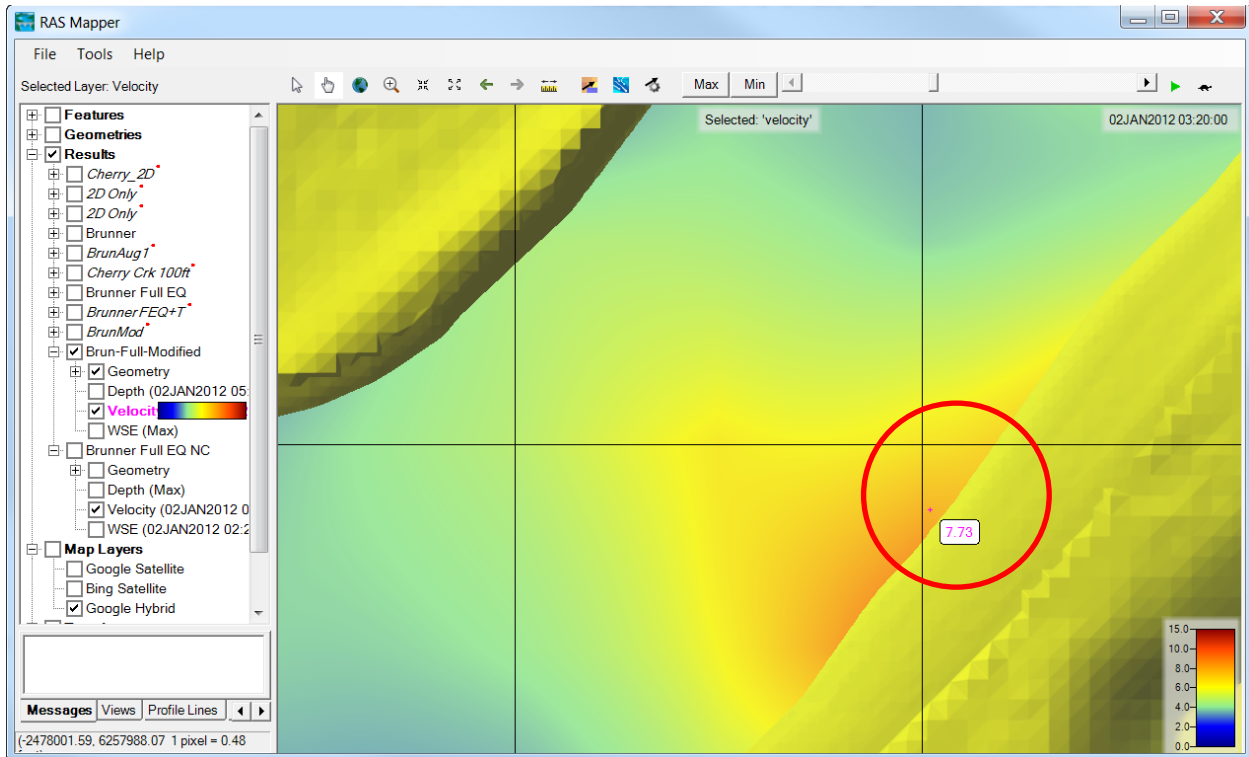


Figure 15. Example of just a corner of a cell wetting.

Internal Hydraulic Structures

2D Flow Areas can have all of the same issues with hydraulic structures as described above for 1D river reaches. So, weirs, culverts, gates, etc... can all cause model stability/iteration issues inside of 2D areas. Some common problems with structures inside of 2D flow areas are:

1. Too small of a cell size at the opening of a culvert or gate. If you have small cells at the upstream or downstream end of a culvert or gate, you will need small time steps for this to work and stay stable. Keep in mind that flow is computed at each time step for the culvert or gate. Based on the time step, that is a specific volume of water ($V = Q \times T$). Smaller cells have less volume. So if the culvert or gate is large, and/or your time step is large, the volume computed over the time step may be more than the available volume in the cells connected to the ends of the hydraulic outlet. This will cause stability/iteration issues. To fix instability issues around structures in 2D areas, you either have to use smaller time steps, or use larger cells at the ends of the culverts and gates.
2. A second issue that can arise is when the hydraulic outlets get fully submerged (i.e. the headwater elevation is nearly the same as the tailwater elevation). The same options used for 1D to improve stability during high submergence can be

used for 2D. For structures with high submergence, there are options in the unsteady flow computations window, under “Computation Options and Settings” that you can set to smooth out structure iteration problems due to submergence. These are the “Weir and Gate Flow Submergence decay exponents”. Turning the decay exponents up to 3.0 will smooth out issues due to submergence by more gradually reducing flows during submergence, and make the model run with fewer iterations.

3. A third issue for internal SA/2D Hydraulic structures is the equation type used for computing the flow going over top of the structure. Two options are available: weir equation or full 2D equations. If the structure is high, and when flow goes over top of it, if it passes through critical depth and falls over the top (flow not controlled by a hydrostatic flow distribution and friction), then the weir equation is more appropriate. Using the 2D equation for this type of flow situation can cause model instability and excessive iterations. If the structure is not that high, and will become easily submerged, then the 2D equations will be more accurate and more stable.
4. Another issue that can come up is when you have a lateral structure connected to a 2D area. This is similar to the issue discussed above. If the lateral structure is high, and flow will pass through critical depth when it goes over the top, then the weir equation should be used for modeling the overflow. If the lateral weir is very low, and flow leaving the channel does not act like weir flow as it goes into the 2D Flow Area, then the “Normal 2D Equation domain” option should be used. However, occasionally the 2D equation approach may cause some instability or iterations, in which case, switching back to a weir, with a very low weir coefficient may reduce the stability/iterations problem.

Dam and Levee Breaching

In general, dam and levee breaching will require smaller time steps, and results in rapid changes in flow, depth, velocity, etc. If the model has extremely small breach times, this will require smaller time steps to stay stable. Look at the breach times closely, and make sure they are realistic. Increasing a breach time even a small amount can result in a much more stable model that can use larger time steps, or at least will not iterate as much.

Other options that can help with dam and levee breaching are to use the sine wave breach development curve for a smoother breach development and try establishing more base flow in the downstream system before the breach occurs. Additionally, using a variable time step may significantly improve run times for these types of models. A variable time step will allow you to use a larger time step leading up to the breach, a smaller time step during the breach and for some time after, then a larger time step again after the breaches has dampened.

Boundary Conditions

Both upstream and downstream boundary conditions can be a source of model instability or excessive iteration issues. For upstream flow hydrograph boundary conditions connected to 2D Flow Areas, the user is required to enter an energy slope. This slope is used to compute a water surface elevation for any given flow in the boundary condition. From the water surface, the program figures out how much flow is going into each cell along the boundary condition, as well as the associated velocity of that flow. If too steep of a slope is used, lower water surface elevations will be computed and more flow at higher velocities will be placed into fewer cells. This higher flow and velocity may cause instabilities and excessive iterations. Therefore, if the cells connected to that boundary condition are iterating a lot and/or the water surface is lower than the cells downstream of them, or the velocities are higher than the cells downstream of them, then this is a sign you have entered too steep of a slope. Reduce the slope to improve the stability in the area near the boundary condition.

For downstream boundary conditions, users generally use either a rating curve or normal depth type of boundary conditions. Normal depth also requires an energy slope to be entered, and this type of boundary condition can have the same issues as an upstream boundary condition, as described above. Too steep of a user-entered slope can cause high velocities and stability/iteration issues. Check to ensure those energy slopes at model boundaries are reasonable.

For a rating curve, issues can arise when the low flow portion of the rating curve is not well defined or is very steep in slope (i.e. rapid changes in water surface elevation for small changes in flow). Make sure the rating curves are accurate, well defined at the lower end, and they transition smoothly through the range of water surface elevations and flows.

Variable Time Step

A new feature in HEC-RAS 5.0.4 is a variable time step option based on Courant Numbers, or the user can enter a table of dates and times versus time step. The variable time step capability can dramatically reduce the run time required for a model because the model can be set up to use larger time steps when the flow and velocities are lower, the time step will automatically decrease as the flow and velocities increase. Once the main flood wave passes, and velocities begin to decrease again, the time step can also increase again.

This new Variable Time Step option is available from the Unsteady Flow Analysis window. There is a new button right next to the Computation Interval directly on the Unsteady Flow Analysis window, as shown in Figure 16.

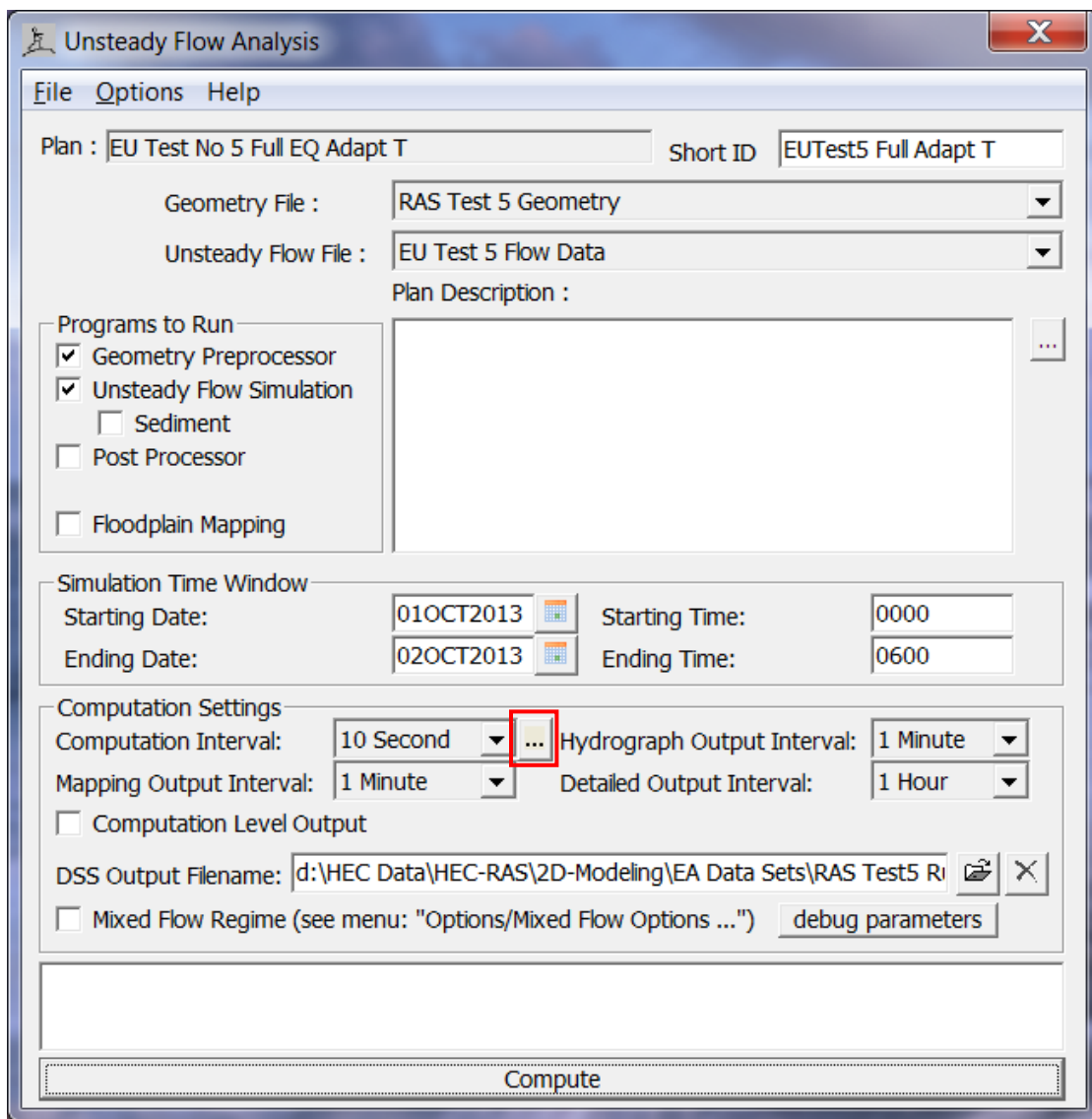


Figure 16. Button for Variable Time Step Option Shown on Unsteady Flow Analysis Window.

When the Variable Time Step option button is pressed, a new window will appear as shown in Figure 17.

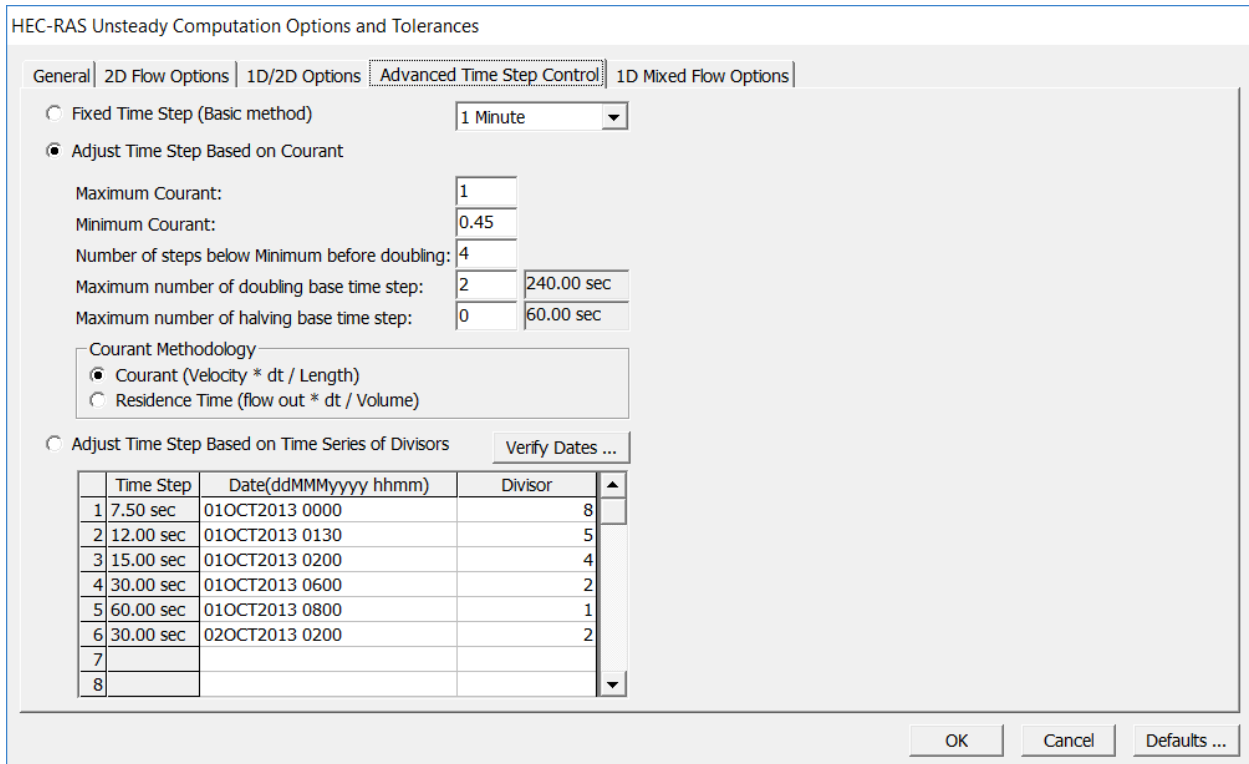


Figure 17. Variable Time Step editor.

As shown in Figure 17, HEC-RAS now has three different methods for selecting and controlling the computational time step: a Fixed Time Step (default); a variable time step based on the Courant number; and a variable time step based on a user entered table of dates, times, and time step divisors.

For the details on how to use the new Variable Time Step capabilities, see the “New Features” document that comes with version 5.0.4 of the software.

Too Many Points in Structure Station/Elevation Definitions

A common problem that decreases computational efficiency is having too many points in the Station-Elevation tables defining the embankment of a structure (inline, lateral, and SA/2D structures). It is very easy to get station/elevation data from a terrain model in GIS, and end up with 500 points in every structure to define the top profile of the structure. In general, it is not necessary to have that many points to define a structure accurately for hydraulic overflow calculations. Additionally, when flow goes over the structure, a separate weir flow or hydraulic calculation is required between every two points in the structure that are wet. This can lead to an extremely large number of hydraulic calculations required to compute flow over

the hydraulic structures. Additionally, these calculations are carried out every iteration of every time step that water is going over the structure.

Computational time can be dramatically reduced by filtering the number of points in the station/elevation data for all of the hydraulic structures down to a reasonable number (generally, around 100 or fewer points is fine for most structures). The fewer the number of points in a structure, the faster it will compute.

HEC-RAS has a “Filter” button on all of the station/elevation data editors for each of these types of structures. You have to do it one at a time, but you need to view the comparison of the filtered line with the original line to make sure the reduced number of station/elevation points is appropriate. While it may seem tedious to edit each individual structure, if you have a lot of structures, it is well worth doing to improve computational speed. Figure 18 contains a plot of a Lateral Structure that had over 2000 points in the Station-Elevation table, filtered down to 100 points. As you can see 100 points is more than adequate for modeling the terrain in detail.

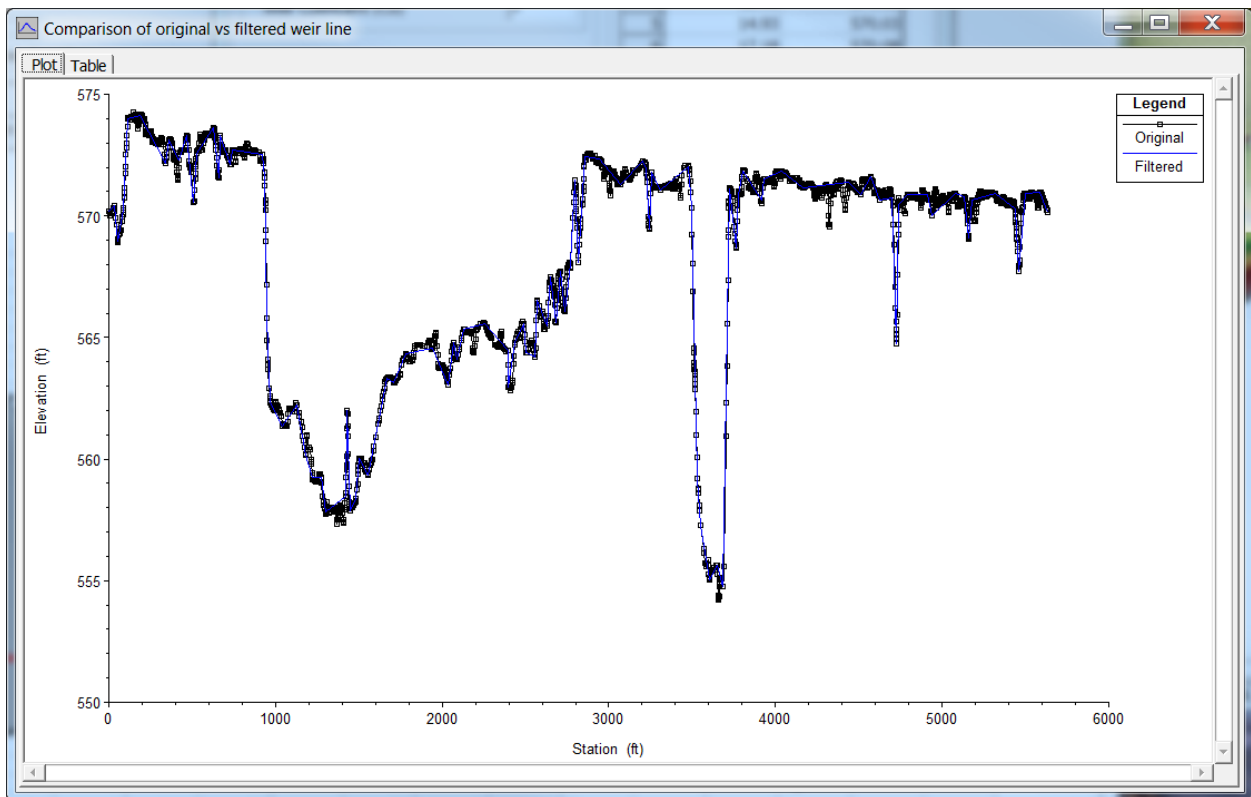


Figure 18. Example Lateral Structure with over 2000 Points, Filtered to 100 Points.

Too Many 2D Cells

If you have 2D Flow Areas in your model, the computational time is directly related to the number of cells in the 2D Flow Areas. More cells requires more computational time. So in general, while you want to have enough cells, at a small enough size to get accurate answers, too many cells will dramatically increase model run times. There are several things to look at when trying to reduce the number of cells in a 2D Flow Area.

The first thing to think about is, having a boundary polygon much larger than needed. If you have a lot of cells that never get wet, even for your largest events, those cells are still contributing to computational time. They will still be placed into the solution matrix, and some operations are done on them every time step to see if they have gotten wet or not. This will slow down the computations. Keep in mind that the computational time increases non-linearly with the number of cells, so this increased computational time can be significant. If you have larger areas that never get wet, modify the outer boundary of the 2D area to cover a smaller area to remove unneeded cells.

Another issues is that sometimes there are small cells all over the model. While there will definitely be areas that require smaller cells for improved accuracy, there are also areas that you can get away with much larger cells. For example: backwater zones where water backs into an area slowly with a flat water surface; areas inside of a levee that will fill with a flat water surface; large, flat overbank areas, where the water will move out there slowly and generally have a mild sloping water surface as it moves out and fills the area. In these types of areas, because of the slower changes in water surface elevation and velocities, larger cells can be used. The new Mesh Refinement Region tool in version 5.0.4 of HEC-RAS i allows the user to draw a polygon and define a new cell size inside of the polygon. Additionally, a different cell size can be defined along the boundary of the polygon. The cell size along the polygon is gradually transitioned to the nominal cell size (or a user-defined cell size) outside of the polygon. This tool can be used to thin out (reduce) the number of cells in an area, or increase the number of cells in an area. An example mesh, where the new Mesh Refinement Region tool was used to make cells of one size in the channel, but much larger cell sizes in the overbank areas, is shown in Figure 19.

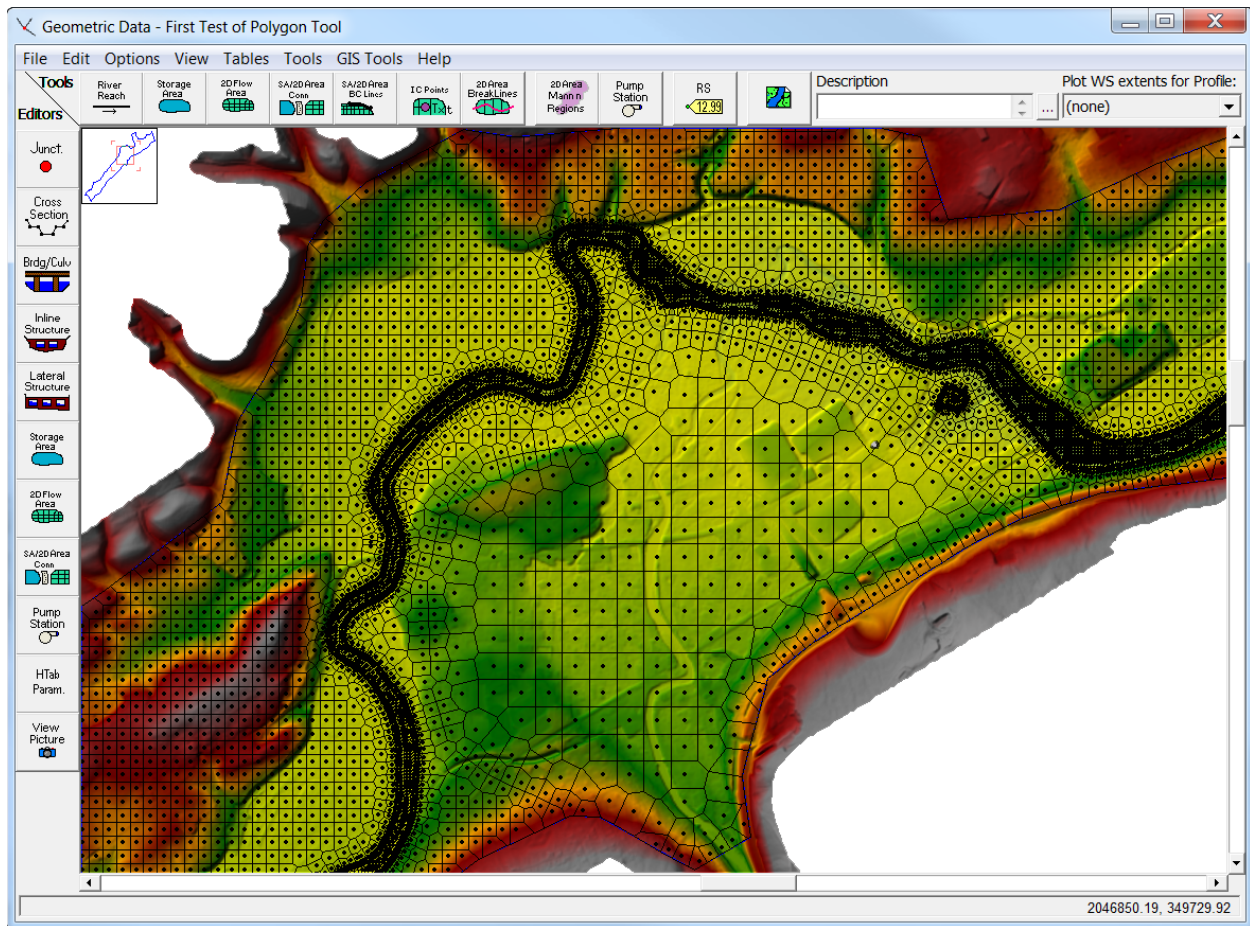


Figure 19. Example 2D Mesh with Variable cell sizes for main channel and overbank areas.

Another issue that comes up is when users have tried to use very small cells to represent a road or a levee, or even some natural overflow ground. Instead of using small cells, use breaklines to define the top high ground profile of the structure, then set a reasonable cell size/spacing along the breakline. Because the faces of the cells are lined up with the high ground, there is not the need to have very small cells to ensure that flow does not go over the structure until it reaches the correct elevation.

Pardiso Solver for Large Complex 1D Models

If you have a large 1D model with many rivers, and the river reaches have many interconnections (i.e. loops), using the Pardiso matrix solver may decrease the computational time of the model. The Pardiso solver is an option you can turn on from the Unsteady Flow Computational Options and Settings.

By default, the 1D equation solver uses a Skyline/Gaussian reduction matrix solution technique to solve the 1D unsteady-flow equations. There is an option to use the “Pardiso” matrix equation solver. Experience has shown that even though the Pardiso solver can make use of multiple processor cores, the Skyline matrix solver is generally faster for dendritic river systems. However, for large, complex, looped systems, the Pardiso solver may produce faster computational results. The user should try the Pardiso solver if they have a larger complex system of interconnected reaches. If you turn on the Pardiso solver, there is an option to control the number of processor cores used for solving the matrix. By default, “All Available” will be used. However, the speed of the solution is sensitive to the number of cores used, and it is not always faster with using the maximum number of cores available (typically, larger models will benefit from using more cores). Users should test the number of cores to get the maximum computational speed benefit for large models.

Re-Ordering the Reaches for Large Complex 1D Models

For very large 1D models, with many separate reaches, which are interconnected (i.e. loops in the system), the order of the reaches can cause the matrix to be larger than it needs to be, and thus cause the program to take longer to compute. By default, HEC-RAS has a routine that tries to figure out the reach order, and place it in the matrix correctly. In general, the algorithm will always create an order that will work, but may not be the most efficient.

Creating a user defined reach order that minimizes the size of the matrix and speeds up the computations can be quite difficult. To do this requires a lot of understanding and experience on how the skyline matrix solver works, and how the matrix gets put together. Please review the section in Chapter 2 of the Hydraulic Reference Manual on the Skyline matrix solution. Users can set the reach order for computations by going to the Geometric Data Editor, selecting “Tools”, then “Reach Order for Computations”. An example of this is shown in Figure 20, for the Sacramento River system. The general approach is explained:

1. Find the upstream reach that is the furthest away from the downstream outlet, based on computation nodes and not distance. Computation nodes are cross sections, bridges, culvert, lateral structures, etc... Make this the first reach in the system
2. Going down stream to the next junction, pick the second largest reach (based on number of nodes), and do that one next, and so on.
3. When you come to a flow split, pick the side that has the fewest nodes in it first. Continue down that side until you get to the junction that brings the other side back in, then go to the top of that system, back up to the flow split.
4. Continue down the system as described in Steps 1 through 3, until you get to the last reach.

When you go to run the model, if the reach order is wrong (i.e. in an order that it will not be able to compute), the simulation will fail during the initial conditions phase of the simulation. If this happens, check the order of the reaches to make sure you did not make a mistake.

If creating a user defined reach order seems to complex, but you would still like to try it, please contact the HEC-RAS team for assistance with your specific model.

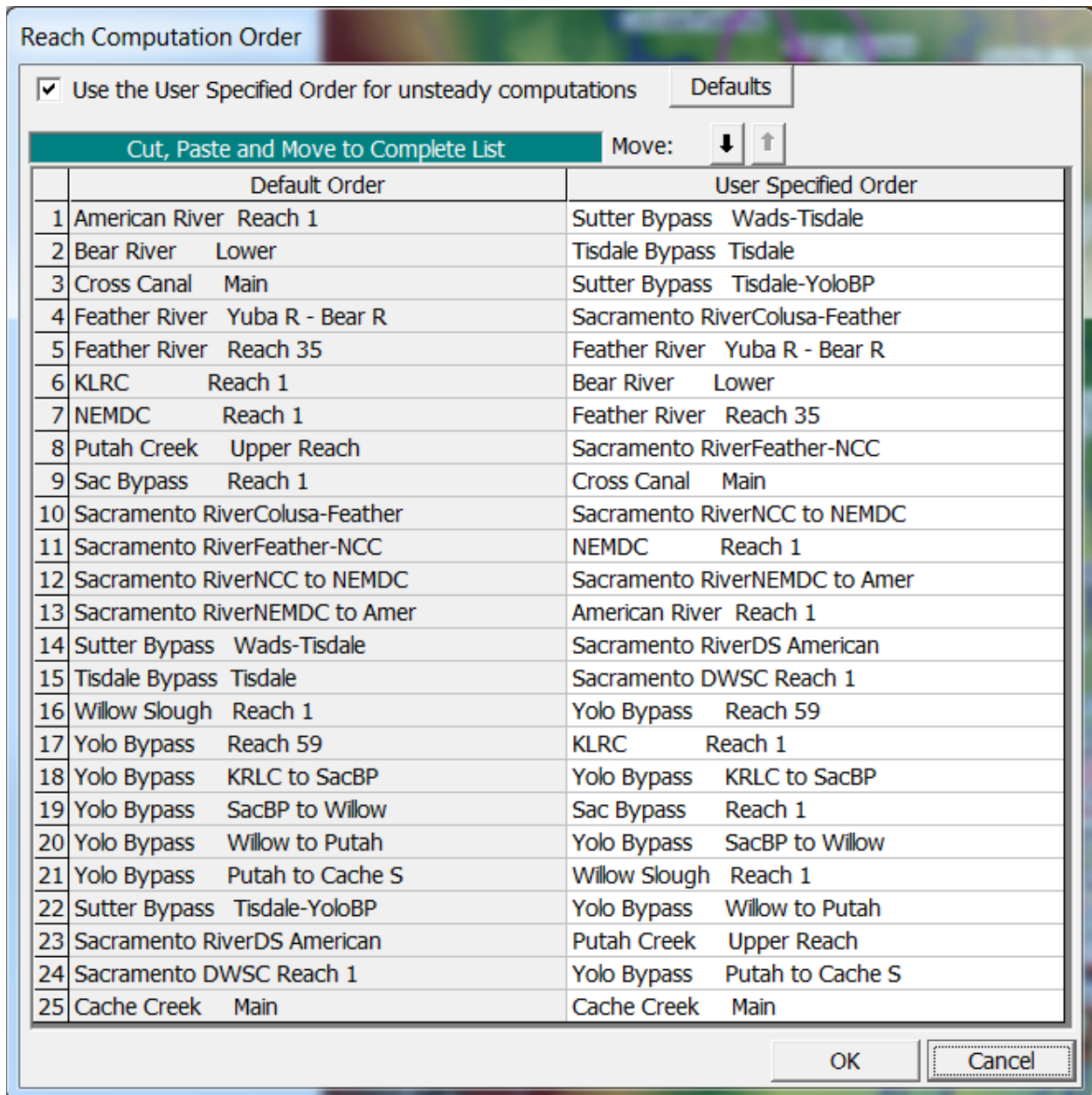


Figure 20. Example of User Defined River Reach Order for Computations.