Computational Parameters

Cameron Ackerman, P.E., D.WRE

USACE, Institute for Water Resources, Hydrologic Engineering Center

This presentation will discuss considerations for running an HEC-RAS model that contains 2D Flow Areas. The presentation will cover performing computations using the interface including the 2D preprocessor; computation options, cell size considerations, and time step selection.

Overview

- Performing Computations
  - 2D Preprocessor
- Computation Options
- Cell Size Considerations
  - Associating a Terrain Model
- Time Step Selection

Performing Computations

- Floodplain Mapping
  - Creates a stored map
  - Must set up map in RAS Mapper prior to run
- Mapping Output Interval
  - Time Series Data written to HDF file at Cross Sections, Storage Areas, and 2D Cells
  - XS
    - WSE, Flow, Velocity
  - Cells
    - WSE, Depth cell center
    - Velocity at cell faces
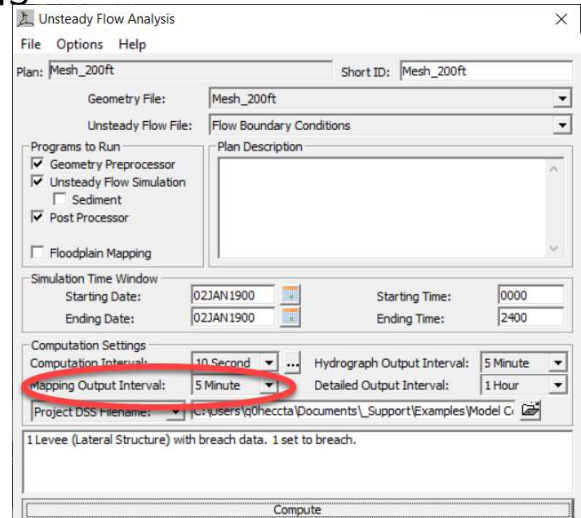
3

Computations are performed from the Unsteady Flow Simulation dialog.

A new option available is the Floodplain Mapping option. This option will create stored map output (saved to disk) based on the output map established through the RAS Mapper. Setting up RAS Mapper prior to the simulation run is required. You should Associate the Terrain model with the Geometry file prior to running the 2d preprocessor, at a minimum.

The other pertinent output option is the Mapping Output Interval. This is the time interval that results will be written to the output HDF5 file. You can then access the output file through the Mapper to display depth and velocities.

The output written to the output file for cross sections is different than for the 2D Cells. For XSs, you can plot WSE, flow, and velocity. For Cells, you get depth at the cell center, velocity at the cell points (corners), average velocity at the cell faces, and average shear stress on the cell face.

Performing Computations

1. Writing 1D Geometry
   - Bank Lines, Edge Lines (1D)
   - Interpolation Surface (1D)
2. 2D Geometry Preprocessor
   - Process 2D Cells and Faces
3. 1D Geometry Preprocessor
   - Hydraulic Tables for Cross Sections
4. Unsteady Flow Simulation
5. Post Processor (1D only)
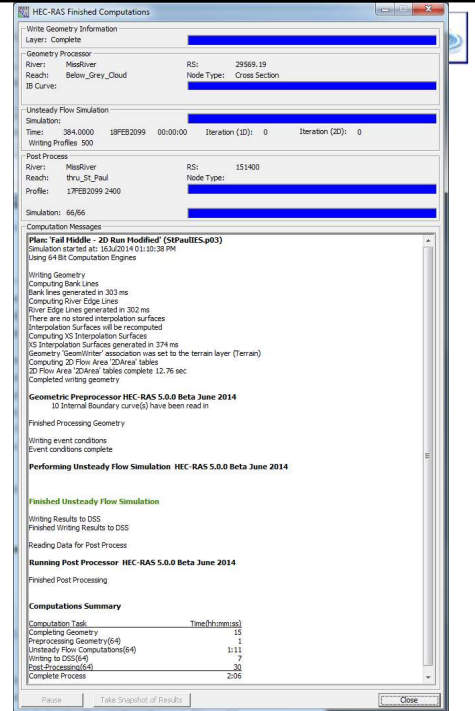
Performing the computations is a lot like a 1D unsteady flow simulation. However, the 2D Cell preprocess is first to develop hydraulic tables for the cells. Elevation/Volume for the cells and Elevation/Area information for cell faces is computed. The 2D preprocessor also computes geospatial information such as bank lines and edge lines that transition from cross section to cross section, as well as an interpolation surface based on those transition lines. Hydraulic tables for each cross section will then be computed. After the preprocessors have run, the Unsteady Flow Simulation occurs. Lastly, the postprocessor will create output for the cross section output. All 2D Cell output is will be displayed in RAS Mapper and inundation maps/grids will be computed as needed.
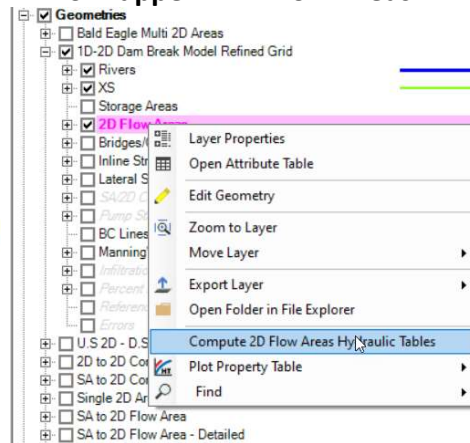
2D Pre-processor to compute cell information

Unsteady-flow Simulation

RAS Mapper – 2D Flow Areas

The 2D preprocess is accessed in RAS 2 different ways. The first way is that when a user runs a simulation, the 2D preprocessor is run to create hydraulic table information for each 2D Cell, if necessary. The second way is that users can access the preprocessor through RAS Mapper by right-clicking on the 2D Flow Area and selecting "Compute 2D Flow Area Hydraulic Tables".

If you have already run the preprocessor in RAS Mapper, then during the unsteady simulation, it will not have to reprocess the tables.

Computation Options

- Defaults
- Per 2D Flow Area

HEC-RAS Unsteady Computation Options and Tolerances		
General 2D Flow Options 1D/2D Options Advanced Time Step Control 1D Mixed Flow Options		
☐ Use Coriolis Effects (not used with Diffusion Wave equation)		

|	Parameter	(Default)	BaldEagleCr
1	Theta (0.6-1.0)	1	1
2	Theta Warmup (0.6-1.0)	1	1
3	Water Surface Tolerance [max=0.2](ft)	0.01	0.01
4	Volume Tolerance (ft)	0.01	0.01
5	Maximum Iterations	20	20
6	Equation Set	Diffusion Wave	Diffusion Wave
7	Initial Conditions Time (hrs)		4
8	Initial Conditions Ramp Up Fraction (0-1)	0.5	0.5
9	Number of Time Slices (Integer Value)	1	1
10	Turbulence Model	*Non-Conservative (original)*	*Non-Conservative (original)*
11	Longitudinal Mixing Coefficient		
12	Transverse Mixing Coefficient		
13	Smagorinsky Coefficient	*0*	*0*
14	Boundary Condition Volume Check	☐	☐
15	Latitude for Coriolis (-90 to 90)		
16	Solver Cores	6 Cores	6 Cores
17	Matrix Solver	PARDISO (Direct)	PARDISO (Direct)
18	Convergence Tolerance		
19	Minimum Iterations		
20	Maximum Iterations		
21	Restart Iteration	*10*	*10*
22	Relaxation Factor	*1.3*	*1.3*
23	SOR Preconditioner Iterations	*10*	*10*

OK   Cancel   Defaults ...

6

The DEFAULT column in the table has the base RAS defaults parameter values.
A new column will be added for each 2D Flow Area and it's name will be at the top

Computation Options

- Theta
- Theta Warmup
- Water Surface Tol
- Volume Tolerance
- Max Iterations
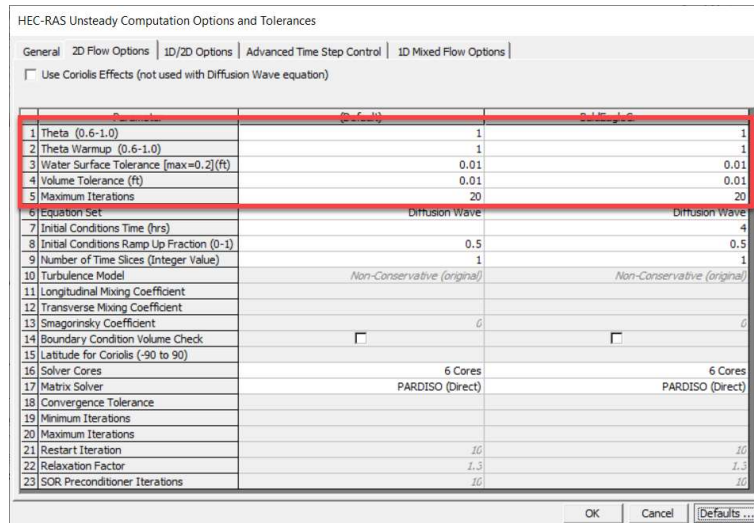
Theta (0.6 – 1.0): 1.0 (default) This is the implicit weighting factor that is used to weight spatial derivatives between the current solution time line and the previously solved solution time line. Theta of 1.0 (Default), uses only the currently solved time line for the spatial derivatives. This provides for the most stable solution, but possibly at the loss of some accuracy. Theta of 0.6, provides for the most accurate solution of the equations, but tends to be less stable. In general it has been found that in application of most real world flood runoff types of events, Theta of 1.0, will give about the same answers as Theta of 0.6. However, this should be tested for each model due to site specific geometry and flood propagation, in which it may make a difference in the results.

Theta Warm-up (0.6 – 1.0): 1.0 (default)   This is the value of Theta (see description above) that is used during the model warmup and ramp up periods. This value of Theta is only used if the user has turned on the unsteady flow warm-up option, or the Boundary Condition Ramp up Option for 2D areas.

2D water surface calculation tolerance (ft): 0.01 (default)   This is the 2D water surface solution tolerance for the iteration scheme. If the solution of the equations gives a numerical answer that has less numerical error than the set tolerance, then the solver is done with that time step. If the error is greater than the set tolerance, then the program will iterate to get a better answer. The program will only iterate up

to the maximum number of iterations set by the user. The default is set to 0.01 ft based on experience in using the model for a range of applications.

RAS does volume error check for 2D and has to relate that to a water surface error using the Elev-Vol curve.

Volume Tolerance (ft): 0.01 (default) This is the 2D water volume solution tolerance for the iteration scheme. The volume error is converted to feet of error, by taking the currently solved for water surface elevation into the elevation-volume curve for the cell, then calculating the change in water surface elevation based on the current volume error at that point on the curve. If the solution of the equations gives a numerical answer that has less volume error than the set tolerance (in terms of ft), then the solver is done with that time step. If the maximum error is greater than the set tolerance, then the program will iterate to get a better answer. The program will only iterate up to the maximum number of iterations set by the user. The default is set to 0.01 ft based on experience in using the model for a range of applications.

Maximum Number of iterations (0 – 40): 20 (Default)   This is the maximum number of iterations that the solver will use while attempting to solve the equations (in order to get an answer that has a numerical error less than the user specified tolerance at all locations in the 2D computational mesh domain). The default is set to 20. However, the user can change it from 0 to 40. It is not recommended to change this unless you are sure that changing the value will either improve the chances that the model will converge (I.e. increasing the value), or speed up the computations without causing any significant errors.

Computation Options

- Equation Set

- Diffusion Wave
- Shallow Water Eqn (faster)
- Shallow Water Eqn (conservative)
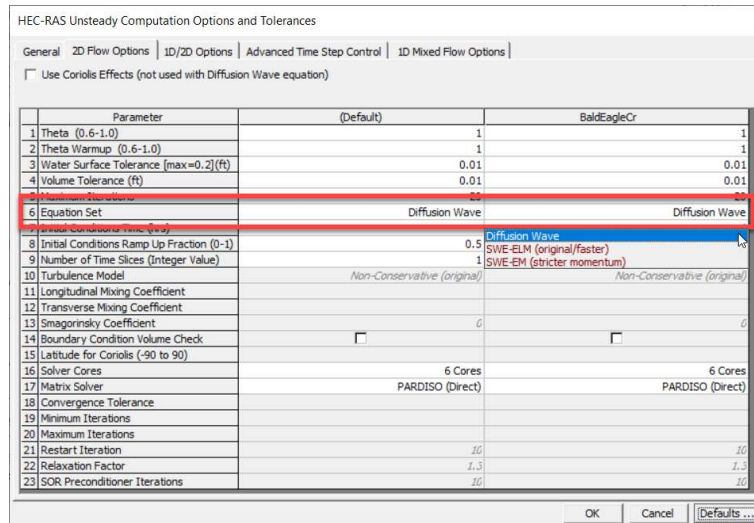
Equation Set: Diffusion Wave (Default) or Full Momentum Equation
The HEC-RAS two-dimensional computational module has the option of either running the **2D Diffusion Wave equations, or the Full 2D St. Venant equations** (sometimes referred to as the full 2D shallow water equations). The default is the 2D Diffusion Wave equation set. In general, most flood applications will work fine with the 2D Diffusion Wave equations. The Diffusion Wave equation set will run faster and is inherently more stable. However, there are definitely applications where the 2D Full St. Venant equations should be used for greater accuracy. The good news is that it easy to try it both ways and compare the answers. It is simply a matter of selecting the equation set you want, and then running it. Create a second Plan file, use the other equation set, run it, and compare it to the first Plan for your application.

There are 2 versions of the Shallow Water Equation solver. A the original version – which is faster – and a newer version, which is more momentum conservative. The original/faster version will be more that appropriate for river hydraulic modeling. The newer approach is more for EXTREMELY rapid flood wave propagation in detailed settings like a lab experiment.

Computation Options

Initial Conditions

- Time
- Ramp Up Fraction


Initial Conditions Ramp up Time (hrs): Default is Blank (not used)
This option can be used to "Ramp up" the water surface from a dry condition to a wet condition within a 2D area (or from a flat water surface if an initial water surface elevation was entered). When external boundary conditions, such as flow and stage hydrographs (or 1D reaches), are connected to a 2D area, the first value of the connected flow or stage may be a high (i.e. a very large flow or a stage much higher than the cell elevation it is attached to). If the model were to start this way, such a high discontinuity may cause a model instability. So this option allows the user to specify a time (in hours) to run the computations for the 2D Flow Area, while slowly transitioning the flow boundaries from zero to their initial value, and the stage boundaries from a dry elevation up to their initial wet elevation. The user specifies the total "Initial Conditions Ramp up Time" in this field (10 hours for example). The user must also specify a fraction of this time for Ramping up the boundary conditions. A value of 0.5 means that 50% of the Initial Conditions time will be used to Ramp Up the boundary conditions to their initial values, the remaining time will be used to hold the boundary conditions constant, but allow the flow to propagate through the 2D Flow Area, thus giving it enough time to stabilize to a good initial condition throughout the entire 2D Flow Area. The Ramp up time for the boundary conditions is entered in the next row, which is labeled "**Boundary Condition Ramp up Fraction".**

1D/2D Model and Computational Time Step
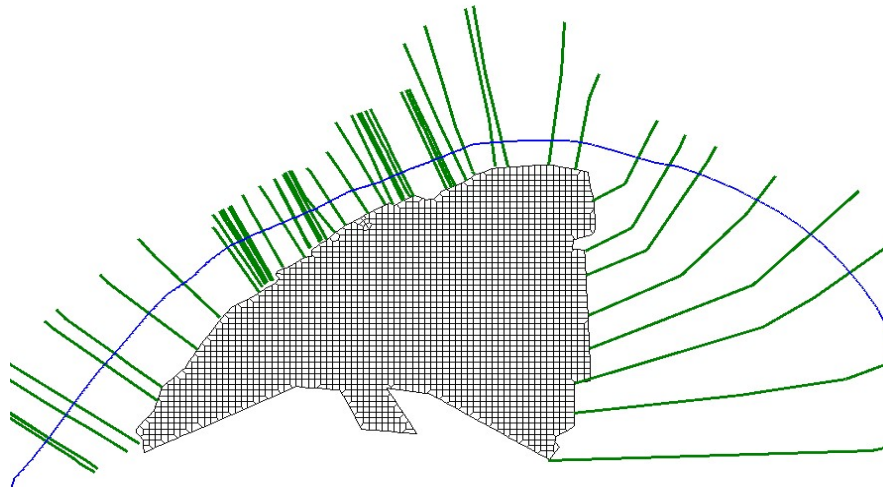
Here is an example where cross sections are spaced at 1000ft intervals (on average).
It was determined that the 2D Grid Cell size should be 100ft (on average).
To satisfy the Courant condition we consider spacing (cell size) and average velocities.
If velocities are the same in the 2D Area as in the river, then we should choose a time
slice that is 1000/100 or 10 (if we find velocities are typically much slower in the 2D
Area, then we could use a smaller number of slices).

Computation Options

- Time Slices

Number of Time Slices (Integer Value): 1 (Default)
This option allows the user to set a computational time step for a 2D area that is a fraction of the overall Unsteady flow computation interval. For example, if the user has set the Unsteady Flow overall computation interval to 10 minutes, then setting a value of 5 in this field (for a specific 2D area) means that the computation interval for that 2D area will be 1/5 of the overall computation interval. Which for this example would be 2 minutes (e.g. 10/5 = 2). Different values can be set for each 2D Flow Area. The default is 1, which means that 2D Flow Area is using the same computational time step as the overall unsteady flow solution (computation Interval is entered by the user on the unsteady flow analysis window).

Main RAS Unsteady Flow only (currently) allows you to go to 1second timestep. If you are running a very detailed model, you may need to go down below 1sec so you can set time slicing to further reduce to 0.5 s or 0.1s.

Computation Options

- Boundary Condition
  Volume Check

Consider the situation where there is a lateral structure connected to a 2D Flow Area. RAS knows the water surface elevation in the 2D Cell(s) adjacent to the lateral structure and compute weir flow based on the WSE. IF you have the Volume Check on, RAS will make sure that there is enough water volume available in the 2D area to pull from (over the weir) based on the weir calculation. If there is not, RAS will iterate and reduce flow over the lateral structure.

Computation Option

• Number of Cores
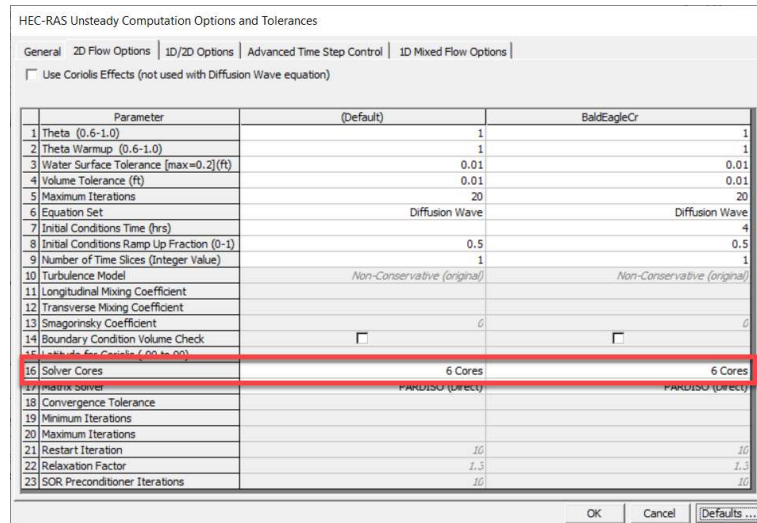
The 2D Solver uses Pardiso and will use the number of cores available on your computer.

The HEC-RAS two-dimensional computational module was developed from the ground up with parallel processing in mind. The HEC-RAS 2D computations will use as many CPU cores as there are available on your machine (which is the default mode for running). However, HEC-RAS provides the option to set the number of cores to use for the 2D computations. In general, it is recommended to use the default of "All Available". However, you may want to experiment with this for a specific data set to see if it will either speed up or slow down computations based on a specific number of cores. The ideal number of cores for a given problem is size and shape dependent (shape of the 2D Flow Area). As you use more cores, the problem is split into smaller pieces, but there is overhead in the communications between the pieces. So, it is not necessarily true that a given problem will always run faster with more cores. Smaller data sets (2D areas with fewer cells) may actually run faster with fewer cores. Large data sets (2D Areas with lots of cells) will almost always run faster with more cores, so use all that is available.

Processing Time vs Number of Cores

Speedup
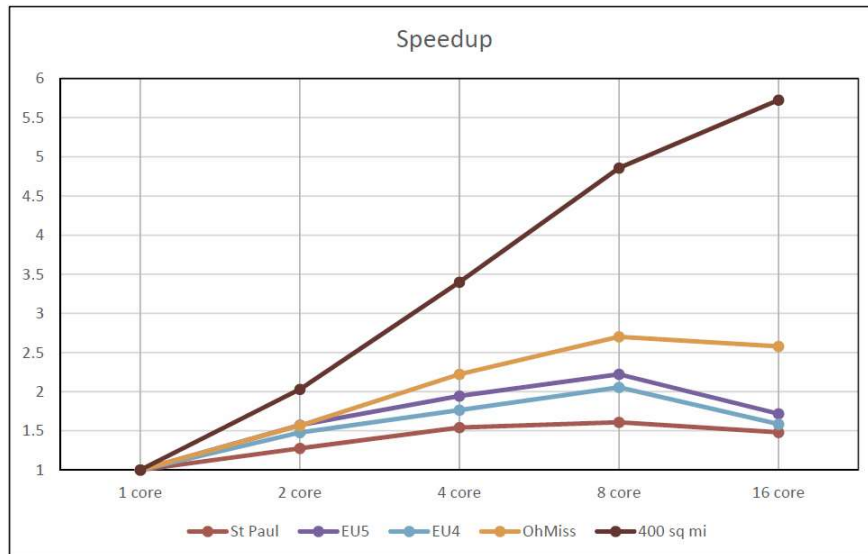
Shown are the results of testing a few data sets by running them with different numbers of Cores. Each model was run several times with the number of cores set to: 1, 2, 4, 8, and 16. As you can see four of the data sets had speed improvements up to 8 cores, but actually ran slower with 16 cores. These are smaller data sets ranging from 10,000 to 80,000 cells. However, one data sets had speed improvements all the way up to 16 cores. This was the largest data set, with 250,000 cells.

Mesh Cell Size

- Cell size determination is based on
  - Land surface features of the Terrain
    - Faces control water movement
  - Water Surface Slope
    - Capturing localized effects in model requires smaller cell sizes (analogous to cross section spacing for rapidly varied flow).
- Start with a "large" cell size and refine the model through iteration.
- You always test different cell sizes to see what affect they have on water surfaces and velocities.

Cell faces should capture the detail in the terrain features that govern/direct the movement of flow. Because, RAS uses the cell faces as cross sections, cell size can vary though out the model, yet still produce accurate results despite having large cells.

Because a single water surface is computed at the center of each cell, the larger the cells are the farther apart the cell centers; therefore, the water surface slope is averaged over a longer distance. This may be appropriate where the water surface slope does not vary much, but will be not be appropriate where the water surface is changing rapidly. If the water surface changes rapidly, smaller cell sizes should be used to capture the water surface.

Key factors for developing a good computational mesh with HEC-RAS are the following:
1. Make sure the cell sizes, shapes, and orientations adequately describe the terrain. Specifically, since the cell faces control the movement of water, there must be enough of them, oriented correctly to describe the key features of the terrain that will control water movement. This includes barriers to flow, such as roads, levees, and natural high ground areas, that will prevent flow from going from one area to

another, until the water surface elevation is higher than the barrier.
2. The cell size must be adequate to describe the water surface slope and changes in the water surface slope. If the water surface slope does not change rapidly, larger cell sizes can be used to accurately compute the water surface elevation and slope. If the water surface slope changes rapidly, then smaller cell sizes need to be used to have enough computation points to describe the changing water surface, as well as compute the force/energy losses that are occurring in that area. While cell sizes (and shapes) can vary, transitioning from larger to smaller cell sizes should be done gradually to improve computational accuracy.

Time Step

- Time Step selection will be based on the size of the grid cells and velocity of water in the cells.
- Dependent on Equation Set selected
  - Diffusion Wave approximation
  - Full Shallow Water Equations

$$S_f = S_o - \frac{\partial y}{\partial x} - \frac{V}{g}\frac{\partial V}{\partial x} - \frac{1}{g}\frac{\partial V}{\partial t}$$

The 2d Computations in HEC-RAS support the Diffusion Wave approximation or Full Saint Venant Equations. The Diffusion Wave approximation is faster and more stable; however, is not a accurate a solution. In the DW approximation the friction slope is approximated by ignoring the local (due to time) and convective acceleration (due to change in location) (inertial) terms of the full SVE.

Time Step

- Full Shallow Water Equations
  - Experience shows, max C = 3.0

$$C = \frac{V * \Delta T}{\Delta X} \leq 1.0$$

- Diffusion Wave Approximation
  - Experience shows, max C = 5.0

$$C = \frac{V * \Delta T}{\Delta X} \leq 2.0$$

C	=	Courant Number
V	=	Velocity of the Flood Wave (ft/s)
ΔT	=	Computational Time Step (seconds)
ΔX	=	The average Cell size (ft)

17

Practical Time Step Selection: The way to use these equations is to find the area(s) with high velocities and rapid changes in water surface and velocity (with respect to space and time). Take the average cell size in that area for ΔX. Put in the maximum velocity in that area for V (average velocity is a good first estimate for this calculation). Then select a ΔT, such that the Courant Number (C) is equal to the suggested value (i.e. 1.0 for Shallow Water Equations and 2.0 for Diffusion Wave). However, you may be able to get away with a Courant number as high as 3.0 for the shallow water equations and 5.0 for the Diffusion Wave equations, and still get stable and accurate results. If the event being modeled changes gradually with time and space, larger time steps can be used (i.e. Courant numbers approaching the maximum listed values). If the flood wave being modeled changes rapidly with respect to time and space, then you will need to use a time step closer to a Courant number of 1.0 (i.e. C = 1.0) for the high velocity zones.

Note: The user should always test different cell sizes (ΔX) for the computational mesh, and also different computational time steps (ΔT) for each computational mesh. This will allow the user to see and understand how the cell size and computational time will affect the results of your model. The

selection of ΔX and ΔT is a balance between achieving good numerical accuracy while minimizing computational time.

Variable Time Step

- Reduce time step to improve accuracy
- Increase time step to reduce computation times

For models that have long computation times, it may be useful to use an adaptive time step.

The time step can be shortened to improve accuracy during high velocity and rapidly changing conditions and then the time step can be increased during other parts of the run in order to reduce computation time.

For models that have short computation times, it is generally easier to just run the entire model using the minimum, fixed time step that is needed.

However, for models with long computation times, an adaptive time step can make a significant difference.

Time Step Options

- Fixed Time Step (default)
- Adjust Time Step Based on
    - Courant Number
    - Date/Time and a Time step divisor

There are three options for the time step. The time step can be fixed for the entire run. The time step can be adjusted based on the courant number. The time step can be adjusted based on the date and time.

Variable Time Step

- Maximum Courant before halving time step
- Minimum Courant before doubling time step
  - **Min < ½ Max !**
- Number of steps before doubling
- Maximum number of doubling
- Maximum number of halving
- Courant (default) or Residence Time

○ Fixed Time Step (Basic method)	1 Second ▾
⦿ Adjust Time Step Based on Courant	
Maximum Courant:	4.
Minimum Courant:	1.95
Number of steps below Minimum before doubling:	5
Maximum number of doubling base time step:	2 4.00 sec
Maximum number of halving base time step:	1 0.50 sec

Courant Methodology
⦿ Courant (Velocity * dt / Length)
○ Residence Time (flow out * dt / Volume)

When using the courant option, for every time step RAS Unsteady will determine the maximum courant number by computing the courant number at every 1D cross section and at every 2D face.

If this maximum courant number is greater than the user entered maximum courant, then Unsteady will halve the time step for the next time step. The time step can only be halved up to the user specified amount. In the example above, the initial time step is 1.0 s and the user has entered "1" for the number of times it may be halved, so the minimum time step is 0.5 s.

If the maximum computed courant is less than the user entered minimum courant, then Unsteady can double the length of the time step for the next time step. In the example above, the user is allowing it to be doubled "2" times for a maximum time step of 4.0 second.

The user entered minimum courant value should be less than half the user entered maximum courant value. For example, if the maximum, computed courant number was exactly 4.0 and the time step was halved, the courant number (for the same flow conditions) for the next time step would be 2.0. If the user entered minimum courant was greater than this (for instance if it was 3), then it would be immediately exceeded and the time step would then be doubled causing the time step to rapidly flip back and forth.

Mapping Output Interval

- Adaptive time step is always integer interval of Mapping Output Interval, so Unsteady may need to adjust base Computation Interval

Continued from previous slide…
In order to improve stability, the user can enter a value for the number of steps below minimum before doubling. In the example given, the computed courant number has to be below 1.95 for 5 consecutive time steps before the time step will be doubled.

The time step must always land exactly on the Mapping Output Interval (this is a RASMapper output performance issue). This may cause RAS to adjust the user entered Computation Interval. In the adaptive time step example above, the user has entered a base time step of 2 seconds and a mapping interval of 10 seconds. If the 2 second time step were doubled to a 4 second time step, the unsteady computations would land on 8 seconds and 12 seconds, skipping over the 10 second output interval. To prevent this, Unsteady has reduced the initial time step to 1.25 seconds. This time step can be doubled up to three times (2.5 seconds, 5 seconds, and 10 seconds) while always landing on the 10 second output interval.

The adaptive time step is not required to land exactly on the Hydrograph Output Interval nor the Detailed Output Interval. If it does not, it will interpolate results at the user entered value.

Time Series of Divisors

- Control time step length by date
- Enter integer halving divisor to reduce time step

Although it generally easier to use the courant method, the user can specify the time step at different points in the model run based on the date.

The Computation Interval should be entered as the maximum time step that is desired during any part of the run. This time step can then be reduced during user specified time windows.

In the example above, the computation interval has been specified as 2 minutes and a divisor of 2 has been entered for 01Jan1990 1200. So the model will *start* the run with a 60 second time step. This time step is used up until the point where the user has entered a 1 divisor, which will cause it to use the maximum time step of 120 seconds. At later dates, the user has specified even smaller time steps.

Because the courant method is based on the maximum courant number anywhere in the model, two closely spaced 1D cross sections or a single, small cell (for instance at a break line) might repeatedly have a large courant number even though there is not an accuracy issue at that location. In this case, using the date based option might work better. Alternately, there may be rapidly changing conditions where it is desired to have a smaller time step, but the maximum courant number is not significantly different than the maximum courant number during other portions of the model run.

Questions?

US Army Corps
of Engineers ®

IWR

HEC