HEC-RAS 2D Sediment Workshop: Optimizing Runtimes

Alex Sánchez, PhD
Stanford Gibson, PhD

Hydrologic Engineering Center,
Institute for Water Resources,
U.S. Army Corps of Engineers, U.S.A.

US Army Corps of Engineers

December 14, 2022

1

Introduction

- Sediment modeling project workflow **NOT** the same as a hydraulic modeling project
- Sediment models require higher data quality and modeling skill
- Sediment modeling is **ALWAYS** iterative
- New or inexperienced modelers often setup with a single high-resolution mesh, with too many grain classes, bad terrain, bad boundary and initial conditions and struggle to get stable results

2

2

Sediment Modeling Workflow

- Start Simple
  - Smallest domain possible
  - Coarsest mesh
  - Fewest number of grain classes
  - Fewest number of processes (e.g. Capacity Only)
  - Large time steps
  - Shortest simulation windows
  - “Easy” boundary conditions
- Slowly refine and add complexity
  - Modeling is an iterative process
  - Focus calibration on important parameters
  - Always conduct sensitivity analysis
- Most difficult part is…
  - Knowing how to get the “Best model for the least amount of effort”

3

3

Outline

- Hydraulics
  - Time Step
  - Mesh Size
  - Convergence
  - Matrix Solver
- Sediment
  - Computation Multiplier
  - Grain Classes
  - Convergence
  - Morphologic Acceleration Factor

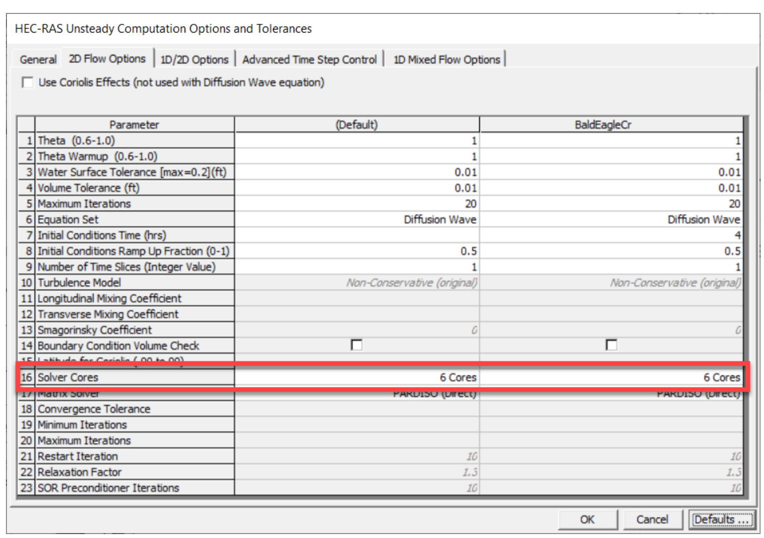
4

4

Optimizing Hydraulics

Number of Cores

- Set for each 2D area
- Default is “All Available”
- Optimal value typically 4-8
- It is more efficient to run multiple plans at the same time



Time Step

- Most common cause of numerical instabilities
- Maximum time step usually limited by just a few cells/faces in the mesh
- Theoretical time step limits are not the same as in practice
- In practice the time step is limited by many factors including
  - Forcing such as wind
  - External and internal flows
  - Local Courant numbers
  - Shape of hydraulic property tables
  - Governing equations
  - 1D/2D and 2D/2D coupling
  - Etc.

$$\Delta t < \Delta t_{\max}$$

7

7

Time Step

- RAS uses a global time for each 2D area (spatially constant) Δt
- Max time step varies in time for unsteady simulations $\Delta t_{\max}(t)$
- Max time step is a function of the 2D flow solver
- Rules of thumb
  - SWE-ELM: C < 3.0
  - SWE-EM: C < 1.0
  - DWE: C < 5.0

$$C = \frac{V \Delta t}{\Delta x}$$

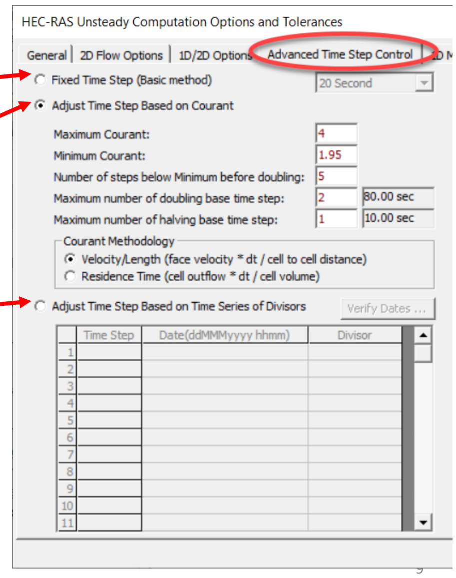
- Courant limitation for ELM and DWE is not strict but rather based on practice
- EM Courant limitation is strict

8

8

Time Step Options

1. Fixed Time Step (default)
   - Can use time step slicing for 2D areas
2. Adjust Time Step Based on Courant Number
3. Adjust Time Step Based on Date/Time and a Time step divisor



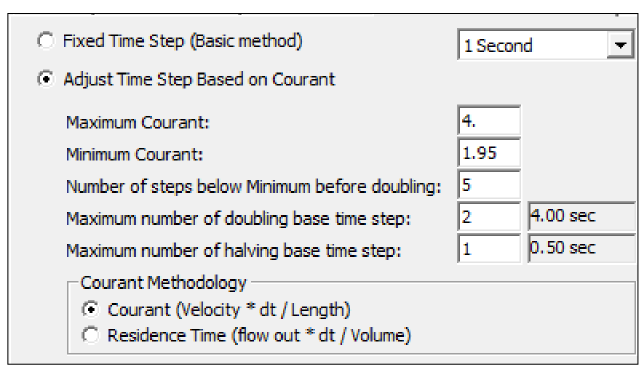
9

Adaptive Time Step

- Time Step Computation $\Delta t = \dfrac{\Delta t_{\max}}{2^N}$

$$C = \frac{V\Delta t}{\Delta x}$$

- Maximum Courant
  - Time step is halved if $C \geq C_{\max}$
- Minimum Courant
  - Time step is doubled if $C \leq C_{\min}$
- Important $C_{\min} < \frac{1}{2} C_{\min}$
- Number of steps before doubling
- Maximum number of doubling
  - Determines max times step
- Maximum number of halving
  - Determines min time step



10

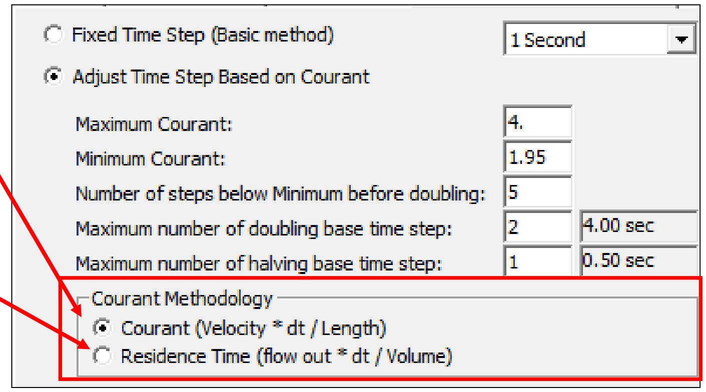
Courant Number Methods

- Standard Velocity Based

$$C_V = \frac{V \Delta t}{\Delta x}$$

- Residence time Based

$$C_\Omega = \frac{Q_{out} \Delta t}{\Omega}$$

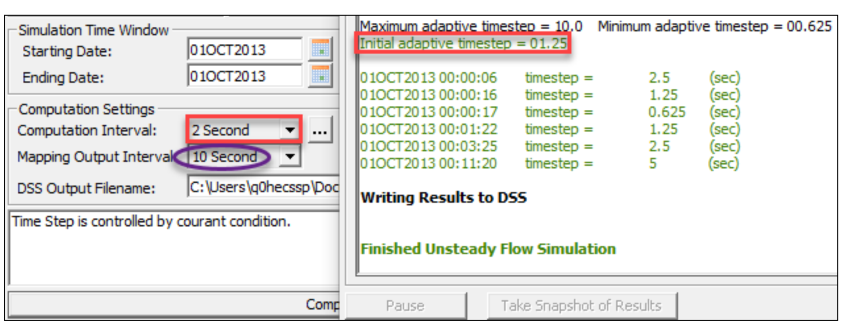


11

11

Mapping Output Interval

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



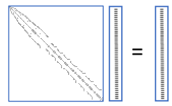
12

12

Matrix Solvers: Introduction

- Linear System of Equations

$$Ax = b$$



- Direct Solvers
  - Examples: Gaussian elimination, LU, Cholesky, and QR decompositions
  - Can be “black boxes”
  - Usually have few input parameters
  - High-accuracy
  - Can fail or be very slow for large matrices
  - Can be slow for unsteady or non-linear systems
- Iterative Solvers
  - Examples: GS, SOR, CG, GMRES
  - Require more options and input parameters
  - Usually require preconditioners, matrix balancing, ordering, etc.
  - Less accurate
  - Good for large problems
  - Good for unsteady or non-linear systems
  - Improper use can lead to instability problems or solution divergence

13

HEC-RAS Direct Solver

- PARDISO

  High-performance, robust, memory efficient and easy-to-use solver for symmetric and non-symmetric linear systems.
  - Version in Intel Math Kernel Library
  - Parallel on PC’s
  - Can be used as a “black box”
    - Very little parameters
    - No need matrix balancing, ordering, etc.

14

HEC-RAS Iterative Solvers

- SOR: Successive Over-Relaxation
  - Relaxation factor ($0 < \omega < 2$)
  - Asynchronous (ASOR) parallel implementation
  - Extremely simple
  - May take many iterations to converge but each iteration is very inexpensive
- FGMRES-SOR: Flexible Generalized Minimal RESidual
  - “Flexible” variant of GMRES which allows preconditioner to vary from iteration to iteration
  - SOR as preconditioner

15

15

Iterative Solver Input Parameters

- Convergence Tolerance
  - Determines the overall accuracy
- Minimum Iterations
  - Increases accuracy, avoids solution drift, and allows the solver to stabilize
- Maximum Iterations
  - Avoids stalling and too many iterations caused by a small convergence tolerance
- Restart Iteration (Only FGMRES-SOR)
  - Reduces run time and memory requirements
- Relaxation Factor
  - Used for both SOR solver and preconditioner
- SOR Preconditioner Iterations (Only FGMRES-SOR)
  - In-lieu of checking convergence which would be slower

Parameter	Range
Convergence Tolerance	0.001 – 0.000001
Minimum Iterations	3 – 6
Maximum Iterations	5 – 30
Restart Iteration (FGMRES Only)	8 – 12
Relaxation Factor	1.1 – 1.5
SOR Preconditioner Iterations (FGMRES Only)	5 – 20

16

16

Iterative Solvers: Stopping Criteria

- Error Estimate

$$E^m = \frac{\left\| D^{-1}\left(Ax^m - b\right)\right\|_2}{\sqrt{N}}$$

D : Diagonal of A

A : Coefficient matrix

x : Solution

b : Right-hand-size

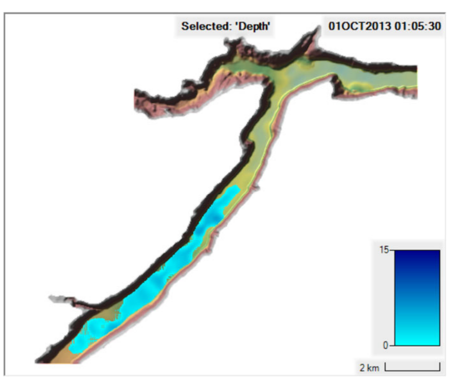
N : Number of rows

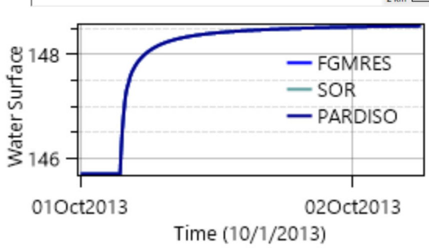
Iterative Solver Status	Criteria	Description
Iterating	$N_{min} < m < N_{max}$ and $E^m > T_C$ and $\frac{E^{m-1} - E^m}{E^1 - E^m} > T_S$ and $E^m < E^1$	Iterative solver is converging and will continue to iterate. N_{min}: Minimum number of iterations N_{max}: Maximum number of iterations T_C : Convergence tolerance $T_S = 0.1 T_C$: Stalling tolerance
Converged	$E^m \le T_C$	Convergence criteria met. Solution accepted.
Stalled	$\frac{E^{m-1} - E^m}{E^1 - E^m} \le T_S$	Convergence rate has decreased to an insignificant level without satisfying the converged criteria. The current solution is accepted and the iteration loop is exited.
Max Iterations	$m = N_{max}$	Maximum number of iterations reached without reaching the converged criteria. The current solution is accepted and the iteration loop is exited.
Divergent	$E^m > E^1$	Iterative solution is divergent. Either the normalized residuals are getting larger, or a Not a Number (NaN) has been detected.

17

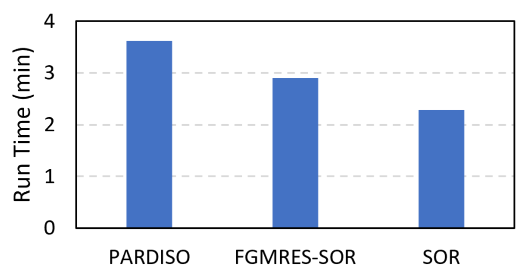
17

Example: EA Test 5 (Dam Failure)





Setting	FGMRES-SOR	SOR
Convergence Tolerance	0.0001	0.0001
Minimum Iterations	3	5
Maximum Iterations	20	30
Restart Iteration	10	
Relaxation Factor	1.3	1.3
SOR Preconditioner Iterations	10	

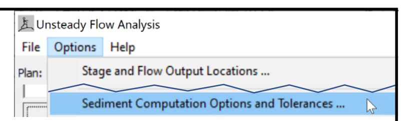


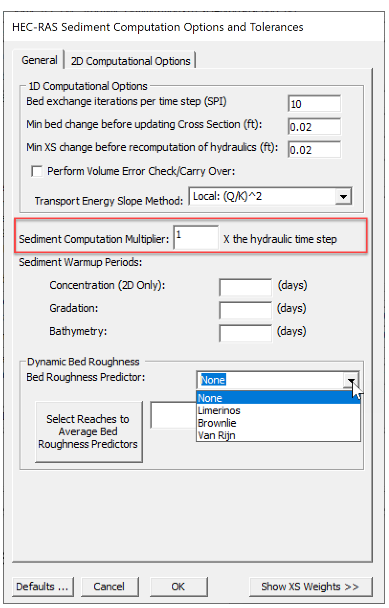
18

18

Optimizing Sediment

Sediment Computation Multiplier

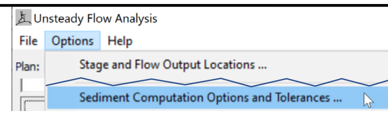




- Number of hydraulic time steps within a sediment time step

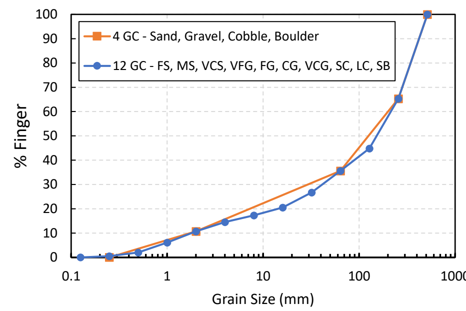
$$\Delta t_{Sed} = m\Delta t_{Hyd}$$

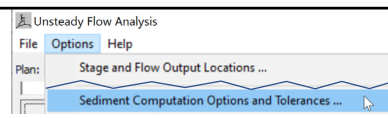
- Multiplier used in adaptive time stepping
- Uses time-average (conservative) fluxes and instantaneous hydraulics
- Output Mapping Interval enforced
- Reduces computational time
- Maximum value
  - Depends on application
  - Typically ranges from 2 to 20
  - Needs validation
- Adaptive scheme coming



Number of Grain Classes

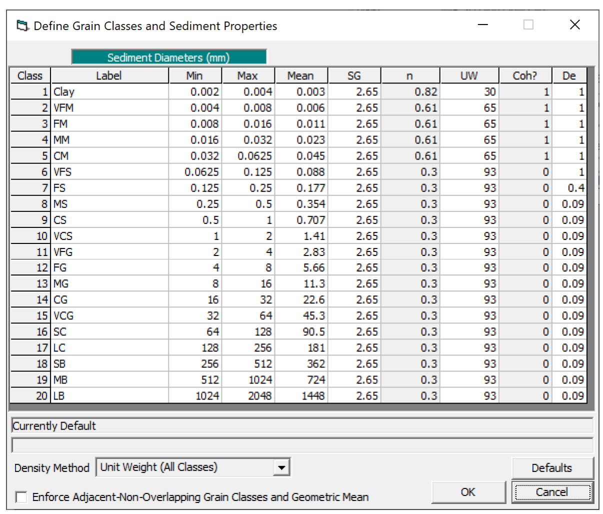
- Sediment Computational approximately proportional to number of grain classes
- However, grain classes are coupled together and solved iteratively
- Number of grain classes can affect convergence and number of iterations
- Reducing number of classes can have a huge impact on computational time
- Start with one grain class and slowly increase number of grain classes





Number of Grain Classes

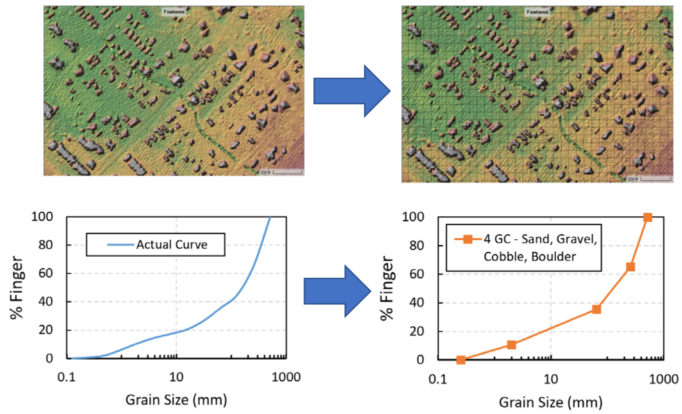
- Default Grain Classes
  - Wentworth (1992) scale
  - Logarithmic with base 2
  - More resolution in the finer particles
  - However, not based on actual particle physics or site-specific bed gradations
- Modeling considerations
  - Cohesive particles erode and often deposit at equal rates due to aggregates and flocs so no need for 5 cohesive grain classes
  - Grain class limits can be better defined using actual bed gradation curves



Class	Label	Min	Max	Mean	SG	n	UW	Coh?	De
1	Clay	0.002	0.004	0.003	2.65	0.82	30	1	1
2	VFM	0.004	0.008	0.006	2.65	0.61	65	1	1
3	FM	0.008	0.016	0.011	2.65	0.61	65	1	1
4	MM	0.016	0.032	0.023	2.65	0.61	65	1	1
5	CM	0.032	0.0625	0.045	2.65	0.61	65	1	1
6	VFS	0.0625	0.125	0.088	2.65	0.3	93	0	1
7	FS	0.125	0.25	0.177	2.65	0.3	93	0	0.4
8	MS	0.25	0.5	0.354	2.65	0.3	93	0	0.09
9	CS	0.5	1	0.707	2.65	0.3	93	0	0.09
10	VCS	1	2	1.41	2.65	0.3	93	0	0.09
11	VFG	2	4	2.83	2.65	0.3	93	0	0.09
12	FG	4	8	5.66	2.65	0.3	93	0	0.09
13	MG	8	16	11.3	2.65	0.3	93	0	0.09
14	CG	16	32	22.6	2.65	0.3	93	0	0.09
15	VCG	32	64	45.3	2.65	0.3	93	0	0.09
16	SC	64	128	90.5	2.65	0.3	93	0	0.09
17	LC	128	256	181	2.65	0.3	93	0	0.09
18	SB	256	512	362	2.65	0.3	93	0	0.09
19	MB	512	1024	724	2.65	0.3	93	0	0.09
20	LB	1024	2048	1448	2.65	0.3	93	0	0.09

Grain Size Distribution

- The **Terrain** is to the **Grid Resolution** as the **Grain Size Distribution** is to the **Grain Classes**
- In other words, the grain classes are analogous to the grid resolution for the grain size distribution



23

23

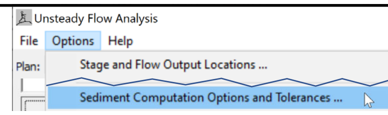
Matrix Solver Best Practices

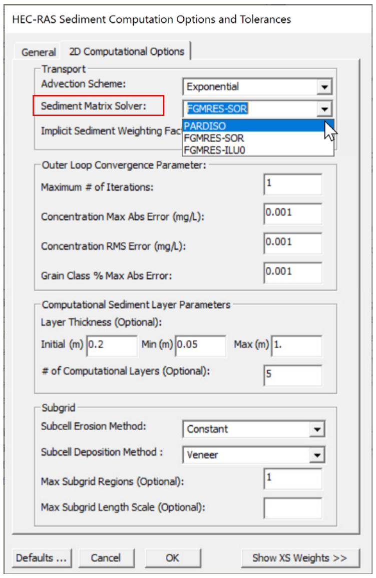
- Start with PARDISO
- Ensure model is stable and not going to max iterations every time step or reporting large water surface or volume errors
- Try FGMRES and SOR solvers
- Start with conservative parameters
- Compare with PARDISO
- Adjust parameters to optimize run time
- Iteration parameters left empty or set to zero are assigned a default value based on mesh size

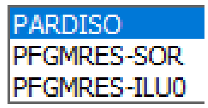
24

24

Matrix Solver Options







More Accurate and Stable

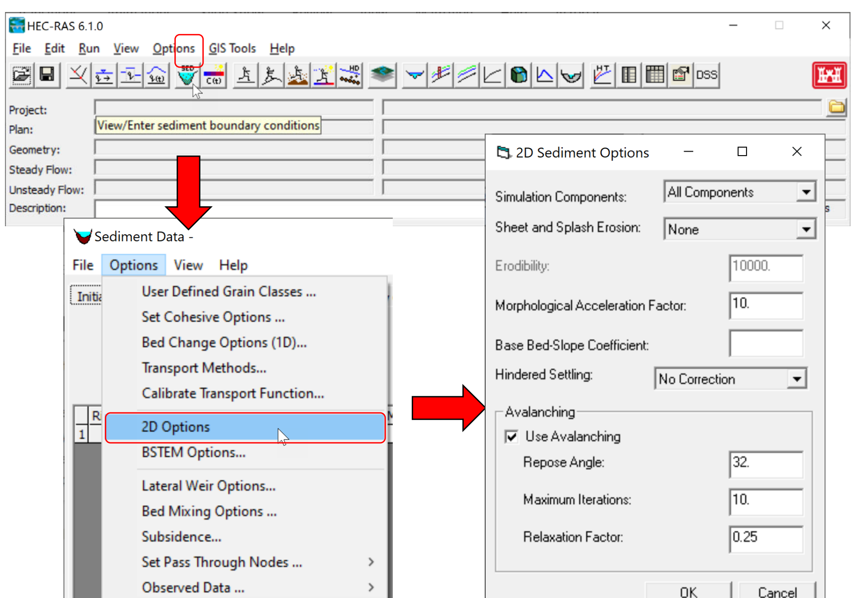
Faster

- PFGMRES - Preconditioned Flexible Generalized Minimal RESidual method
- FMRES - Flexible Generalized Minimal RESidual matrix solver
- SOR – Success Over-Relaxation used as the preconditioner (can also be a solver)
- ILU0 – Incomplete Lower Upper with Zero Fill-in
- Start with PARDISO and quickly switch to one of the other solvers for speed

25

25

2D Options

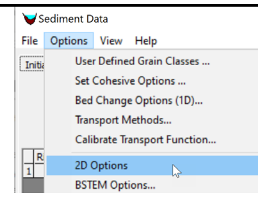


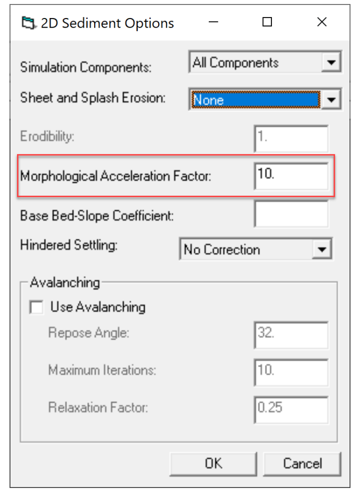
26

26

Morphologic Acceleration Factor

- Approaches/Uses
  - Turn off bed change
  - Scale bed change
    - Simulation time window and boundary conditions unchanged
    - Modeled bed change represents a factor longer simulated window
    - Best used for periodic boundary conditions
  - Scale time
    - Simulation time window and boundary condition time series scaled
    - Modeled bed change represents unscaled time window results
    - Best used for simulating events



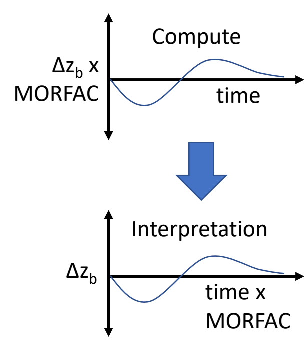


27

27

MORFAC: Scaling Bed Change

- A period is simulated with real time, but the bed change is representative or scaled time period
- Often used in coastal applications
- For example, a 5-year simulation can be run with MORFAC of 20 to simulate 100 years of change
- Changes the order of events which can have a negative impact on the accuracy of the approach

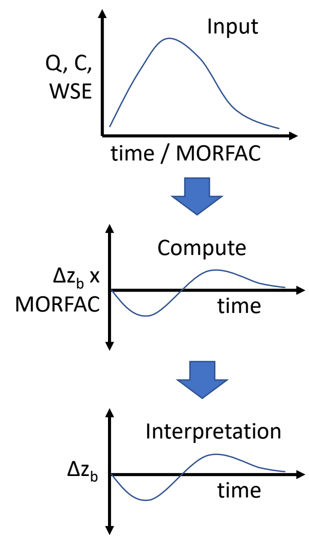


28

28

MORFAC: Scaling Time

- Simulation time is scaled to speed-up the simulation
- Time in all BC's needs be divided by MORFAC
- For example:
  A 12-month period is scaled to 1 month by dividing time by 12. The bed change is multiplied by 12 in the simulation and represents a year of bed change

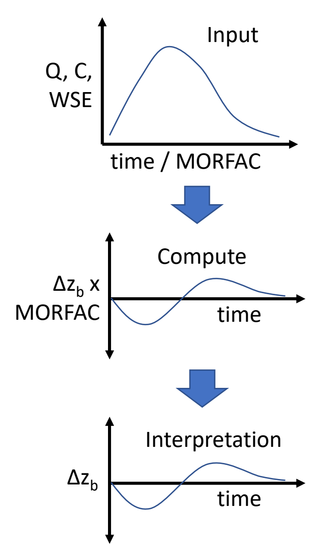


29

29

MORFAC: Scaling Time

- Simulation time is scaled to speed-up the simulation
- Time in all BC's needs be divided by MORFAC
- For example:
  A 12-month period is scaled to 1 month by dividing time by 12. The bed change is multiplied by 12 in the simulation and represents a year of bed change



30

30

31

Thank You!

HEC-RAS Website:
https://www.hec.usace.army.mil/software/hec-ras/

Online Documentation:
https://www.hec.usace.army.mil/confluence/rasdocs

US Army Corps of Engineers ®

32