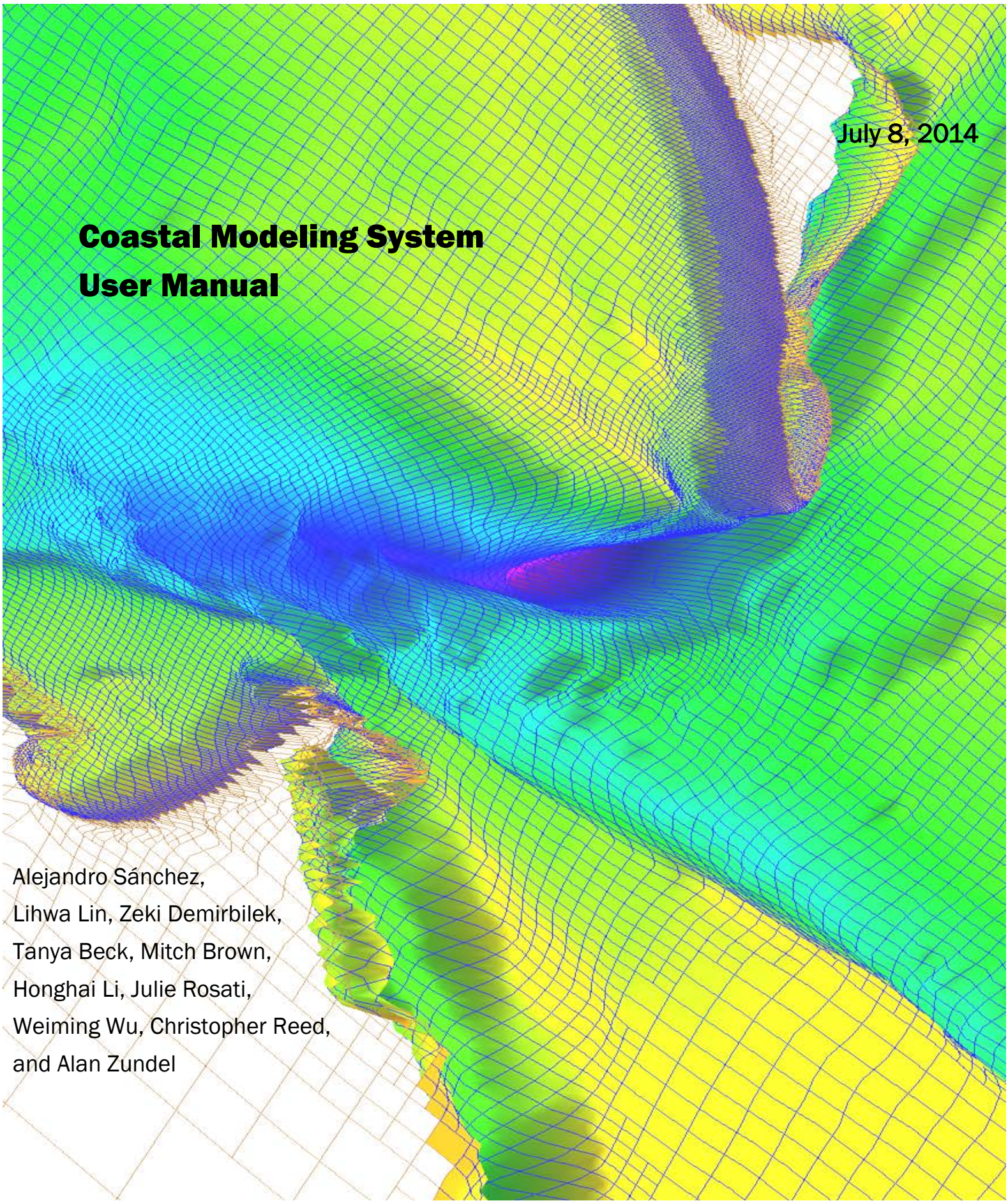


July 8, 2014

Coastal Modeling System
User Manual

Alejandro Sánchez,
Lihwa Lin, Zeki Demirbilek,
Tanya Beck, Mitch Brown,
Honghai Li, Julie Rosati,
Weiming Wu, Christopher Reed,
and Alan Zundel

Alejandro Sánchez, Lihwa Lin, Zeki Demirbilek, Tanya Beck, Mitchell Brown, Honghai Li, and Julie Dean Rosati

Coastal and Hydraulics Laboratory, U.S. Army Engineer Research and Development Center
3909 Halls Ferry Road, Vicksburg, MS 39180-6199

Weiming Wu

Department of Civil and Environmental Engineering
Wallace H. Coulter School of Engineering
Clarkson University
Potsdam, New Work 13699

Christopher Reed

Reed and Reed Consulting, LLC.

Alan Zundel

Aquaveo, LLC.

CMS User Manual

Contents

Unit Conversion Factors xx

1 Introduction **21**

Limitations and Recommendations 21
Introduction to the Modeling Process 23
System Requirements 26

2 CMS-Flow Model Setup **27**

Overview of Input and Output 27
Input Files *27*
Output Files *36*
Overview of SMS Interface 37
Geospatial Information 39
Horizontal Coordinate Systems and Conventions *39*
Vertical Coordinate System and Conventions *43*
Cartesian Grids *47*
Editing Spatial Datasets *53*
Mapping a Spatial Dataset as the Elevation Dataset *56*
Interpolating a Scatter Set to the Whole Grid *58*
Interpolating a Scatter Set to a Portion of Grid *59*
Exporting a Dataset *62*
CMS Input Format Version Number *63*
Timing 64
Hydrodynamics 67
Water Properties *68*
Coriolis *71*
Bottom Friction *73*
Wall Friction *82*
Eddy Viscosity *85*
Wave Mass Flux *89*
Wind and Atmospheric Pressure Forcing *91*
Boundary Conditions *107*
Preprocessing CMS-Flow Input Time Series *193*
Salinity Transport 195
Overview *195*
Setup *196*
Initial Condition *197*
Boundary Conditions *198*
Advanced Settings *199*
Step-By-Step Salinity Transport Application for Matagorda Bay, TX *200*
Sediment Transport 208
Timing *211*

Sediment Properties *213*
Transport Model *223*
Sediment Transport Formula *225*
Scaling Factors *227*
Adaptation Coefficient *230*
Bed Slope Term *241*
Total-Load Correction Factor *242*
Bed Composition *243*
Hiding and Exposure *256*
Avalanching *258*
Hard Bottom *260*
Boundary and Initial Conditions *263*
Step-by-Step Sediment Transport Application for Shark River Inlet, NJ *263*
Numerical Methods 264
Temporal Solution Scheme *264*
Implicit Solver Options *265*
Skewness Correction *266*
Advection Schemes *267*
Wetting and Drying *268*
Parallelization *269*
Hot Start 270
Hot Start Output File *272*
Initial Conditions Input File *275*
Output Options 280
Global Output *280*
Observation Cells (Save Points) *286*
Simulation Statistics *288*
Diagnostic File 290

3 CMS-Wave Model Setup 293
Advanced CMS-Wave Features 296
Full-plane *296*
Wave Run-up, Infra-gravity Wave, Nonlinear Wave-Wave Interaction, Muddy Bed, Spatial Wind Input *296*
Permeable Structures *297*
Grid Nesting *299*
Spatially Varied Spectral Wave Input *302*

4 Surface Roller Model Setup 305

5 Coupling of CMS-Flow and CMS-Wave 306

6 Running CMS 309
Recommended Folder Structure 309
SMS CMS-Flow Menu 310
Standalone Program 312
Windows Priority Levels 316

7 Viewing and Post-Processing Results 318
Water Levels & Currents 318
Sediment Transport 336
Calculating Sediment Transport Roses at Observation (Save) Points 336
Calculating Sediment Fluxes across Observational Arcs 342
Morphology 350

8 Support Policy 363

9 Summary 364

10 List of Symbols 365

11 References 367

12 Appendix A: Description of Input Files 376

13 Appendix B: Description of Output Files 382

14 Appendix C: Matlab Scripting 386

15 Appendix D: Determining the sediment size Classes 390

16 Appendix E: Goodness of Fit Statistics 392

17 Appendix F: Piecewise Lagrangian Polynomial Interpolation 394

18 Appendix G: Inverse Distance Interpolation 395

19 Appendix H: Providing Sea Buoy Data to CMS-Wave 396

20 Appendix I: UTM and State Plane Zones 402

21 Appendix J: Glossary 406

Figures

Figure 1-1. General steps or stages in a modeling study. 23

Figure 2-1. CMS-FlowCard File viewed in TextPad. 34

Figure 2-2. CMS-Flow Card File viewed in Notepad++....35

Figure 2-3. CMS-Flow Model Control: Model Parameters tab for specifying time series output of CMS-Flow calculated datasets.....37

Figure 2-4. General Tab of th *CMS-Flow Model Control* window in SMS 11.1.....38

Figure 2-5. CMS-Flow local coordinate system and grid angle conventions used in the Card and Telescoping Grid Files. The I and J axis indicate the CMS grid axis, while the X and Y axis indicate the global coordinate system. 39

Figure 2-6. Schematic of showing the difference between MSL (tidal) and geodetic datums. 43

Figure 2-7. Types of Cartesian grids supported by the SMS interface and CMS-Flow....47

Figure 2-8. Selecting a single cell 53

Figure 2-9. Selecting cell with a rectangle....53

Figure 2-10. Selecting cells with a polygon in SMS 11.1. 54

Figure 2-11. Selecting cells with a polygon in SMS. 54

Figure 2-12. SMS Edit Window. 55

Figure 2-13. Manning's *n* contours after modifying the areas under all three bridges....55

Figure 2-14. Selecting the Map Elevation menu option in SMS 11.1 56

Figure 2-15. Selecting the Map Elevation menu option in SMS 11.1 57

Figure 2-16. Interpolating a scatter set of Manning's *n* values to the CMS-Flow grid. 58

Figure 2-17. Interpolating a scatter set of Manning's *n* values to the CMS-Flow grid. 58

Figure 2-18. Mapping of a dataset (bottom friction in this case) as the Elevation dataset. 59

Figure 2-19. SMS Interpolation window....60

Figure 2-20. SMS *Interpolation* window....61

Figure 2-21. Mapping of *Depth* dataset as the *Elevation* dataset. 61

Figure 2-22. Manning's *n* contours after modifying the areas under all three bridges. 62

Figure 2-23. *Flow* tab of th *CMS-Flow Model Control* window in SMS 11.1. 67

Figure 2-24. Water density as a function of temperature and salinity (at atmospheric pressure). 68

Figure 2-25. Water dynamic viscosity as a function of temperature and salinity (at atmospheric pressure). 68

Figure 2-26. Example of the Coriolis parameter as a function of latitude using the f-plane and beta-plane approximations. 71

Figure 2-27. *Bottom Friction Dataset* specification via the *Input* tab of the *CMS-Flow Model Control* window in SMS 11.0 77

Figure 2-28. Bottom friction specification via the *Flow* tab of th *CMS-Flow Model Control* window in SMS 11.1. 78

Figure 2-29. Three SMS Project Explorer showing the Manning's *n*, Bottom Friction Coefficient and Roughness Height datasets. 79

Figure 2-30. Wall friction specification in the *CMS-Flow Model Control* window of SMS 11.0 83

Figure 2-31. Wall friction specification in the *CMS-Flow Model Control* window of SMS 11.1. 84

Figure 2-32. Toggle for turning *On* and *Off* the mixing terms in the SMS 11.0 interface. 86

Figure 2-33. Eddy viscosity parameter specifiecation within the *Flow* tab of th *CMS-Flow Model Control* window in SMS 11.1 87

Figure 2-34. CMS-Flow Wind tab in the CMS-Flow Model Control window in SM 11.1. 91

Figure 2-35. CMS-Flow Wind/Wave tab in the CMS-Flow Model Control window in SM 11.0 92

Figure 2-36. CMS-Flow Wind tab in the CMS-Flow Model Control window in SM 11.1. 93

Figure 2-37. CMS-Flow Wind/Wave tab in the CMS-Flow Model Control window. 100

Figure 2-38. Schematic of wind and currents in same (left) and opposing directions (right). 103

Figure 2-39. Examples of good and bad boundary cellstrings. 109

Figure 2-40. Automated Cellstring generation in SMS 11.0 110

Figure 2-41. Selecting a celstring by either clicking on the cellstring handle (left) or dragging a selection box of the cellstring (right). 111

Figure 2-42. Deleting a cellstring by right-clicking on a cellstring and select *Delete BC* from the pop-up menu. 112

Figure 2-43. Deleting a cellstring in SMS by pressing the *Delete* key and selecting *Yes* in the pop-up window. 112

Figure 2-44. Opening the CMS-Flow Boundary Conditions window by left clicking. 114

Figure 2-45. Opening the CMS-Flow Boundary Conditions Window through the CMS-Flow menu. 115

Figure 2-46. CMS-Flow Boundary Conditions Window in SMS 11.0 115

Figure 2-47. CMS-Flow Boundary Conditions Window in SMS 11.1. 116

Figure 2-48. Single water level time-series specification in SMS 11.1. 128

Figure 2-49. Single Water Level Time Series data structure within the CMS-Model Parameters File. 130

Figure 2-50. Selecting the cellstring along the ocean boundary, and clicking on Assign BC. 131

Figure 2-51. Defining the single water surface elevation curve. 132

Figure 2-52. Top: is the blank *XY Series Editor*; Bottom: after the data is pasted in. 133

Figure 2-53. CMS-Flow Boundary Conditions window in SMS 11.1 with an assigned WSE-Forcing from an extracted dataset. 135

Figure 2-54. CMS-Flow grid for Shinnecock Inlet, NY. 136

Figure 2-55. Local CMS-Flow grid and larger-domain ADCIRC mesh for Shinnecock Inlet, NY. 137

Figure 2-56. CMS-Flow Boundary Conditions window in SMS 11.1 with an WSE boundary condition. 138

Figure 2-57. *Extract Boundary Conditions* window in SMS 11.1. 139

Figure 2-58. CMS-Flow Boundary Conditions window in SMS 11.1 with an WSE boundary condition. 140

Figure 2-59. *Extract Boundary Conditions* window in SMS 11.1. 141

Figure 2-60. Setting the tidal database executable location in the SMS Preferences window. 145

Figure 2-61. Example of a flux boundary specified at an oblique angle to the boundary. Contours indicate the current magnitude with warmer colors being larger current velocities. 152

Figure 2-62. Nested CMS-Flow grid within a larger ADCIRC mesh. Note: The ADCIRC mesh was run in geographic coordinates, but was converted to the same horizontal projection as the CMS grid for display. 166

Figure 2-63. Example snapshots of water levels and current velocities for the ADCIRC (top) and CMS-Flow (bottom) simulations for the same time. 168

Figure 2-64. Computationa grids showing bathymetry for parent and child CMS-Flow grids. 170

Figure 2-65. Example of a tidal boundary condition specified with an incident tidal wave direction parallel to the shore contours (not realistic). 173

Figure 2-66. Example computational grid for southern California. 179

Figure 2-67. Sample water levels and current velocites for southern California forced with the water surface elevation constituents extracted from the ENPAC2003 tidal datbase. 180

Figure 2-68. Location of Oil Platform Harvest, CA. 180

Figure 2-69. Comparison of tidal constituent amplitudes (a) and phases (b) at Oil Platform Harvest, CA, for three different tidal databases. 181

Figure 2-70. Salinity tab in the *CMS-Flow Model Control* window in SMS 11.1. 196

Figure 2-71. Specification of boundary salinity time-series in the *CMS-Flow Boundary Conditions* window. 198

Figure 2-72. CMS domain, quadtree grid, and bathymetry of Matagorda Bay, TX. Red dots are the survey stations and red arrows indicate freshwater inflow locations. 200

Figure 2-73. Files for the CMS-Flow salinity simulation. 201

Figure 2-74. Setting up the salinity calculation and specifying spatially varied initial salinity. 203

Figure 2-75. CMS-Flow data tree. 203

Figure 2-76. Initial salinity distribution. 204

Figure 2-77. Salinity boundary types in the CMS. 205

Figure 2-78. Salinity specifications along the *WSE-forcing* boundary. 205

Figure 2-79. Salinity boundary input from a xys file. 206

Figure 2-80. SMS 11.0 interfaces for sediment transport. 208

Figure 2-81. SMS 11.1 interfaces for sediment transport (sediment transport deactivated). 209

Figure 2-82. SMS 11.1 interfaces for sediment transport (sediment transport activated). 210

Figure 2-83. Sediment *Size Class* section within the *Sediment* tab of the *CMS-Flow Model Control* window in SMS 11.1. 215

Figure 2-84. Sediment size class *Fall Velocity Method* options in the *Size Class* section within the *Sediment* tab of the *CMS-Flow Model Control* window in SMS 11.1. 218

Figure 2-85. Sediment size class *Critical Shear Method* options in the *Size Class* section within the *Sediment* tab of the *CMS-Flow Model Control* window in SMS 11.1. 220

Figure 2-86. *Adaptation* section within the *Sediment* tab of the *CMS-Flow Model Control* window in SMS 11.1. 230

Figure 2-87. Example of the total load a*daptation* length specification within the *Adaptation* section within the *Sediment* tab of the *CMS-Flow Model Control* window in SMS 11.1. 238

Figure 2-88. Bed Composition section within the *Sediment* tab of the CMS-Flow Model Control window. 245

Figure 2-89. Bed layer block section within the *Sediment* tab of the *CMS-Flow Model Control* window in SMS 11.1. 248

Figure 2-90. *Avalanching* section within the *Sediment* tab of the *CMS-Flow Boundary Conditions* window. 258

Figure 2-91. CMS-Flow Model Control window showing the location where the hard bottom dataset is specified. 260

Figure 2-92. Specification of the Hardbottom Dataset in the *Sediment* tab of the *CMS-Flow Model Control* window in SMS 11.1 261

Figure 2-93. SMS Project Explorer showing Hard bottom dataset 262

Figure 2-94. Location of Hot Start input and output options within the *Flow* tab of the *CMS-Flow Model Control* window in SMS 11.0. 270

Figure 2-95. Location of Hot Start input and output options within the *Flow* tab of the *CMS-Flow Model Control* window in SMS 11.0. 271

Figure 2-96. HDFView showing the structure of the CMS Hot Start File. 272

Figure 2-97. Dataset Toolbox showing a time step sample of the water elevation and current velocity datasets for use in a hot start (initial condition) file. 278

Figure 2-98. Dataset Toolbox showing a time step sample of the water elevation and current velocity datasets for use in a hot start (initial condition) file. 279

Figure 2-99. *Output* tab in SMS 11.0 280

Figure 2-100. *Cells* tab in SMS 11.0 286

Figure 101. Top Generating wave spectra from wave parameter input; Bottom: Generated wave parameters with a snapshot of spectral output. 294

Figure 102. CMS-Wave model control options 295

Figure 3-103. CMS-Wave *Assign Cell Attributes* in SMS11.1. 299

Figure 3-104. Nesting output 6 locations (blue triangle) and monitoring output 3 stations (red square). 300

Figure 3-105. The child grid domain and spectral input stations (blue triangle). 300

Figure 3-6. CMS-Wave *Nest Grid* and *Nesting Options* menu in SMS11.1. 301

Figure 3-7. Child grid domain and two wave input locations Pt1 and Pt2. 303

Figure 3-8. Example of running merge-eng-to-nst.exe in DOS 304

Figure 6-1. Recommended folder structure for CMS projects. 309

Figure 6-2. Changing the CMS-Flow model executable. 310

Figure 6-3. Example of launching the CMS from the CMS-Flow menu. 311

Figure 6-4. Example of launching the inline CMS-Flow steering run. 313

Figure 6-5. Example of launching the inline CMS-Flow steering run. 314

Figure 6-6. Example of launching the inline CMS by double-clicking on the executable. 316

Figure 7-1. Northern portion of the Shark River estuary. 319

Figure 7-2. Location of NOAA (red) and USGS (blue) water level gauges. 320

Figure 7-3. Contour tab settings for color display and value range. Note the blue colors were selected for the visible range. 321

Figure 7-4. Water surface elevation color contours modified. 321

Figure 7-5. Vector Display options set for a gridded display across the domain. 322

Figure 7-6. Vectors and color contours set for displaying current magnitude (color) and direction (vectors). Note that the vectors are slightly scaled on this setting to reflect magnitude. 323

Figure 7-7. Observation point placed at the location of the Belmar tide station. 324

Figure 7-8. Input of time series water surface elevation data into an observation point. 325

Figure 7-9. Plot Wizard for time series data (top); Time Series plot of Water Elevation (bottom) 326

Figure 7-11. Observed time series of water level at Sandy Hook and Belmar ("Bay") and calculated water level at Belmar. 328

Figure 7-12. Sampling time steps of the Water_Elevation scalar dataset in the Dataset Toolbox. 329

Figure 7-13. Sampling time steps of the Water_Elevation scalar dataset in the Dataset Toolbox. 330

Figure 7-14. Example of the vector display along the measured transects. Note the vectors are scaled to the magnitude, and these scales can be manipulated in the Scatterset Vector tab. 331

Figure 7-15. Data calculator used to convert the vector magnitude from centimeters to meters for comparison to the calculated velocity magnitude. 332

Figure 7-16. Color options turned on for scatter points. 333

Figure 7-17. Example of a feature arc with arrows in the ebb direction. Redistribute Vertices, Reverse Arc, and other options are available in the drop down menu. 334

Figure 7-18. Plot Wizard data options for displaying measured and calculated data from an observation arc. 335

Figure 7-19. Plotted measured and calculated data from the larger observation arc to the East. Note that the positive direction reflects the observation arcs direction. 336

Figure 7-20. Opening the CMS-Flow Cell Attributes window. 337

Figure 7-21. CMS-Flow Cell Attributes window. 338

Figure 7-22. Opening the SMS Save As window. 339

Figure 7-23. SMS Save As window. 340

Figure 7-24. Total-load Sediment transport rates for Shark River Inlet. 342

Figure 7-25. Data tree showing the sediment and hydrodynamic simulation statistics. 343

Figure 7-26. Example of the Net Total-load Sediment Transport Rate vectors and magnitudes for Shark River inlet after a short 24-hr simulation. 344

Figure 7-27. Creating a new Map Coverage of type Obesrvation. 345

Figure 7-28. Setting the current Map Coverage to an Observational type. 345

Figure 7-29. Example of a Feature Arc for calculating sediment transport. 346

Figure 7-30. Highlighted Feature Arc with the positive direction shown by the arrows. 347

Figure 7-31. Reversing the arc direction. 347

Figure 7-32. Selecting the correct Start and End Times, Feature Arcs, and Scalar and Vector Datasets in Step 2 of 2 of the Plot Wizard. 348

Figure 7-33. Example plot of an integrated net sediment transport rate across a feature arc. 349

Figure 7-34. Selecting View Values... after left-clicking on the time series plot. 349

Figure 7-35. Net sediment transport rate integrated across..... 350

Figure 7-36. Data Calculator Options in the Dataset Toolbox. 351

Figure 7-37. Display of calculated bathymetry now converted to elevations. Note the legend has been modified to display the new contours..... 352

Figure 7-38. Observation Coverage attributes..... 353

Figure 7-39. *Plot Wizard* Step 2. 354

Figure 7-40. Plotted cross-section. 354

Figure 7-41. Data Export Options. 355

Figure 7-42. Sampling time steps from a solution dataset..... 356

Figure 7-43. Converting a solution dataset to a scatter dataset. 357

Figure 7-44. Mapping a scatterset contour to the map module. 358

Figure 7-45. Mapped coverage of the 6-m contour at Months 1 and 6 (left and right, respectively). 359

Figure 7-46. Left: Two arcs drawn to connect the 6m contour back to a common point to close off the polygon. Right: Highlighted polygon after created. 360

Figure 7-47. Options for selecting and deleting data from Mesh, Cartesian Grids, and Scattersets using polygons..... 361

Figure 7-48. Selected cells from a polygon. Volume information given in lower, right corner of screen. 362

Figure D- 1. Example of scripting showing the files used. 386

Figure E- 1. Computed grain size distribution from the Matlab script above..... 391

Figure G- 1. NODC buoy data access website. 397

Figure G- 2. NODC buoy data access world map. 397

Figure G- 3. NODC buoy data access regional map. 398

Figure G- 4. NDBC buoy spectral data download web page..... 398

Figure G- 5. Run ndbc-spectra.exe in DOS 399

Figure G- 6. CDIP buoy data access web page. 400

Figure G- 7. CDIP buoy data access web page..... 400

Figure G- 8. CDIP buoy spectral data download web page. 401

Figure G- 9. Run cdip-spectra.exe in DOS. 401

Figure 19-1. UTM zones for the Lower 48 states (from USGS)..... 405

Tables

Table 2-1. CMS-Flow Input Files 28

Table 2-2. CMS-Flow Card notation....29

Table 2-3. CMS-Flow Card format operators. 30

Table 2-4. CMS-Flow Card Format key word description. 30

Table 2-5. Example input parameters and card formats. 31

Table 2-6. Definition of Temperature Unit Character String Variables....32

Table 2-7. Definition of Angle Unit Character String Variables....32

Table 2-8. Definition of Time Unit Character String Variables. 32

Table 2-9. Definition of Length Unit Character String Variables. 32

Table 2-10. Definition of Velocity Unit Character String Variables. 33

Table 2-11. Definition of Volume Flux Unit Character String Variables. 33

Table 2-12. CMS-Flow Output Files 36

Table 2-13. CMS-Flow cards related to the single tidal water level boundary condition. 40

Table 2-14. Horizontal datums currently supported by CMS. 41

Table 2-15. Horizontal coordinate systems currently supported by CMS. 41

Table 2-16. Horizontal coordinate units (HorizUnits) currently supported by CMS. 42

Table 2-17. CMS-Flow cards related to the single tidal water level boundary condition. 44

Table 2-18. Horizontal datums currently supported by CMS. 45

Table 2-19. Horizontal coordinate units currently supported by CMS. 45

Table 2-20. CMS-Flow cards related to the grid files and datasets....49

Table 2-21. Example input parameters and card formats 63

Table 2-22. CMS-Flow Cards associated with timing. 65

Table 2-23. CMS-Flow cards for specifying the water properties. 69

Table 2-24. CMS-Flow Card used to specify the Coriolis parameter approximation....72

Table 2-25. Mean values of Nikradse roughness heights for different bottom types (modified from Soulsby (1983). 74

Table 2-26. Base Manning's n coefficients as a function of the median grain size for the upper flow regime (Benson and Dalrymple 1984). 74

Table 2-27. SMS Cards related to the bottom friction datasets 76

Table 2-28. Advanced Cards to set bottom friction dataset to a constant value 79

Table 2-29. Advanced Card used for specifying the wave-current mean stress formulation 81

Table 2-30. Advanced Card used for activating the bed slope friction factor 81

Table 2-31. SMS Cards related to the wall friction 82

Table 2-32. CMS card used to set turbulence model and eddy viscosity terms 87

Table 2-33. CMS card used to turn on or off the wave volume flux velocities 90

Table 2-34. CMS cards used for spatially constant wind forcing. 93

Table 2-35. CMS-Flow cards related to winds and atmospheric pressure 96

Table 2-36. CMS-Flow cards used to specify the Oceanweather wind, atmospheric pressure and grid coordinates files. 97

Table 2-37. CMS-Flow card for specifying the reference frame. 101

Table 2-38. CMS-Flow card for specifying the reference frame. 103

Table 2-39. CMS-Flow cards related to winds and atmospheric pressure 104

Table 2-40. CMS-Flow cards related to winds and atmospheric pressure 105

Table 2-41. CMS-Flow cards related to winds and atmospheric pressure 106

Table 2-42. CMS-Flow cards related to the flux boundary condition. 119

Table 2-43. Tidal Constituents names (cname) and speeds (in degrees per solar hour) implemented in CMS. 124

Table 2-44. CMS-Flow Cards related to the Tidal Boundary specification. 124

Table 2-45. CMS-Flow Cards related to the Harmonic Boundary specification. 126

Table 2-46. CMS-Flow cards related to the single water level time series boundary condition. 128

Table 2-47. CMS-Flow cards related to the extracted water level time series boundary condition. 135

Table 2-48. CMS-Flow cards related to the flux boundary condition. 146

Table 2-49. CMS-Flow card for specifying cellstrings. 147

Table 2-50. CMS-Flow cards related to the flux boundary condition. 149

Table 2-51. CMS-Flow cards related to the flux boundary condition. 150

Table 2-52. CMS-Flow cards related to the flux boundary condition. 151

Table 2-53. CMS-Flow cards related to the wse boundary condition block. 154

Table 2-54. CMS-Flow cards related to the single water level time series boundary condition. 155

Table 2-55. CMS-Flow cards related to the single water level time series boundary condition. 157

Table 2-56. CMS-Flow cards related to the multiple water level time series boundary condition. 160

Table 2-57. CMS-Flow cards related to the single tidal current velocity boundary condition. 163

Table 2-58. CMS-Flow cards related to the wse boundary condition block. 164

Table 2-59. CMS-Flow Cards related to the Tidal Boundary specification. 171

Table 2-60. Tidal Databases supported by CMS-Flow. 177

Table 2-61. CMS-Flow cards related to the tidal database boundary block 178

Table 2-62. CMS-Flow cards related to the WSE and Currently Velocity boundary blocks. 182

Table 2-63. CMS-Flow cards related to the single tidal current velocity boundary condition. 186

Table 2-64. Water salinity classification. 195

Table 2-65. CMS card used to turn *On* or *Off* the salinity transport 196

Table 2-66. CMS card used to specify the salinity initial condition 197

Table 2-67. CMS card used to turn *On* or *Off* the sediment transport 199

Table 2-68. CMS card used to turn *On* or *Off* the sediment transport 210

Table 2-69. CMS card used to set the sediment transport and morphologic time steps. 211

Table 2-70. Porosity of natural sand beds (modified from Soulsby 1997). 213

Table 2-71. CMS-Flow card used for specifying the global sediment properties. 214

Table 2-72. CMS-Flow card used for specifying the sediment size class block 216

Table 2-73. CMS-Flow card used for specifying the sediment size class diameter. 216

Table 2-74. CMS-Flow cards used for specifying the sediment size class fall velocity. 218

Table 2-75. CMS-Flow cards used for specifying the critical shear stress. 221

Table 2-76. CMS card used to specify the sediment transport model. 223

Table 2-77. CMS-Flow cards used to specify the sediment transport formula and related parameters and options. 225

Table 2-78. CMS-Flow card used for specifying the global sediment properties. 227

Table 2-79. CMS card used to specify the morphology acceleration factor. 228

Table 2-80. CMS card used to specify the Schmidt number 229

Table 2-81. CMS card used to specify the total-load adaptation coefficient. 232

Table 2-82. CMS card used to specify the bed-load adaptation coefficient. 234

Table 2-83. CMS card used to specify the suspended-load adaptation coefficient. 236

Table 2-84. CMS card used to *specify the sediment bed slope diffusion coefficient* 241

Table 2-85. CMS-Flow cards related to the total load correction factor. 242

Table 2-86. CMS-Flow cards used to specify the composition for single-sized sediment transport. 243

Table 2-87. CMS card used to specify the total-load adaptation coefficient 246

Table 2-88. CMS-Flow used for setting a constant mixing layer thickness. 246

Table 2-89. CMS-Flow cards used to specify a *Bed Layer* block. 248

Table 2-90. CMS card used to specify the bed layer thickness. 249

Table 2-91. Sediment sorting classification based on the geometric standard deviation (Folk and Ward, 1957). 250

Table 2-92. CMS card used to specify the bed layer percentile diameters 252

Table 2-93. CMS-Flow card used to specify the option for specifying the bed material composition 253

Table 2-94. CMS-Flow card used for setting the hiding and exposure coefficient 257

Table 2-95. CMS-Flow cards related to avalanching. 259

Table 2-96. CMS Flow cards related to the boundary conditions. 263

Table 2-97. CMS-Flow cards related to the temporal solution scheme. 264

Table 2-98. CMS-Flow cards related to the implicit solver options. 265

Table 2-99. CMS-Flow cards related to the skewness correction. 266

Table 2-100. CMS-Flow cards related to the implicit solver options. 267

Table 2-101. CMS-Flow cards related to wetting and drying. 268

Table 2-102. CMS-Flow card used to specify the number of threads. 269

Table 2-103. CMS-Flow cards used to specify the hot start output file. 273

Table 2-104. Path and name for hot start file variables. 274

Table 2-105. CMS-Flow cards used to specify the hot start input file. 275

Table 2-106. Path and name for initial condition file variables. 276

Table 2-107. CMS-Flow card for specifying the initial condition file. 279

Table 2-108. CMS-Flow cards related to the output datasets. 281

Table 2-109. CMS-Flow cards used for specifying the time series and lists. 282

Table 2-110. Variable group output file name specification. 283

Table 2-111. Advanced output datasets. 284

Table 2-112. CMS-Flow card for compressing the XMDF output file 284

Table 2-113. CMS-Flow cards used to output Tecplot and SMS Super ASCII files. 285

Table 2-114. CMS-Flow card for compressing the XMDF output file 287

Table 2-115. CMS-Flow cards related to output statistics 288

Table 2-116. Hydrodynamic Statistics 289

Table 2-117. Sediment Statistics 290

Table 2-118. Sediment Statistics 290

Table 4-1. CMS-Flow cards related to the surface roller 305

Table 5-1. CMS-Flow cards related to steering 306

Table A1. Description of column data in the CMS-Flow Telescoping Grid File. 376

Unit Conversion Factors

Multiply	By	To Obtain
cubic yards	0.7645549	cubic meters
degrees (angle)	0.01745329	radians
feet	0.3048	meters
knots	0.5144444	meters per second
miles (nautical)	1,852	meters
miles (U.S. statute)	1,609.347	meters
miles per hour	0.44704	meters per second
pounds (force)	4.448222	Newton
pounds (force) per foot	14.59390	Newtons per meter
pounds (force) per square foot	47.88026	Pascals
square feet	0.09290304	square meters
square miles	2.589998 E+06	square meters
tons (force)	8,896.443	Newton
tons (force) per square foot	95.76052	kilopascals
yards	0.9144	meters

1 Introduction

"... and though nature begins with the cause and ends with the experience, we must follow the opposite course, namely begin with experience and by means of it investigate the cause."
- Leonardo Da Vinci

Buttolph et al. (2006) described the graphical interface for CMS in the SMS Version 9.0 (Zundel 2006). A summary of key features of the interface is provided in this chapter to familiarize users with the CMS-Flow interface. The interface is designed to facilitate the model setup and create input files and as well as view output. The interface described here is for SMS Version 11.0 and 11.1. Because the interface changes continuously the structure of this document does not necessary follow that the interface. Instead the structure is based on the types of input and output information.

Limitations and Recommendations

CMS is designed to model hydrodynamics and sediment transport in coastal waters. The model is depth-averaged is does not calculate the vertical profile of current velocities and suspended sediments (although assumed profiles are used for some corrections). Caution is needed when using very small cell sizes compared to the flow depth, since the assumptions of the Shallow Water Equations may be violated. A 3D model may be needed in these situations. The influence of the sub-grid scale turbulence can be particularly relevant in these situations. The implicit hydrodynamic model is capable of handling subcritical, transcritical and supercritical flow regimes but may require a very small time for transcritical and supercritical flows making the computation making the model inefficient. For flows with a high Froude number, it is recommended to use the explicit flow model. Flow through structures is handled using empirical equations.

Sediments are simulated as a passive scalar (no interaction with water) and assumed to be noncohesive, and have constant density and porosity. The hydrodynamic model has capability of including the mass transport due to waves. This produces a net offshore current velocity in the surf zone which is consistent with field measurements. The offshore sediment transport caused by the undertow is counter acted by an onshore sediment transport due wave asymmetry, and skewness. Currently, the formulations for onshore sediment transport are still under development. Therefore, when the Stokes velocities are activated in CMS-Flow, the sediment

transport is still calculated using the total flux velocities and no cross-shore sediment transport is included.

It is recommended to always turn on the surface roller model. This model is very fast and represents an insignificant increase in computational costs. The results however, have been shown to significantly improve when simulating nearshore currents and water levels (Sánchez et al. 2011b). The default parameters and setup options in CMS are designed for robustness, speed, and in some cases, consistency with previous versions of the CMS. For example, the default advection scheme is a first order exponential scheme although higher order schemes are available because the exponential scheme is very stable. Therefore it is recommended for the user to become familiar with the model theory, numerical methods, and input options in order to be able modify the default model parameters intelligently.

The CMS is designed to run on a desktop PC for grids with less than 500,000. The recommended maximum grid size depends on the simulation duration, time step, wave coupling, and sediment transport. But in general it is recommended to keep the total number of active cells less than 150,000 for short term simulations (weeks to months), less than 80,000 for mid-term (months-years), and less than 30,000 for long-term simulations (multiple-years). The grid resolution is therefore a compromise between accuracy and computational costs. For nearshore circulation, it is recommended to have at least 10 computational cells in the in the surf zone in the cross-shore direction. For channels a minimum of three cells is recommended perpendicular to the channel axis. For inlets it is recommended to have at least ten cells across the entrance. However, these are just rules of thumb and it is recommended to test the grid convergence by comparing results from different grids with different resolutions.

Introduction to the Modeling Process

The steps of a numerical modeling study may vary, but a general flow chart is presented in Figure 1-1. A description of each stage is provided in the sections below.

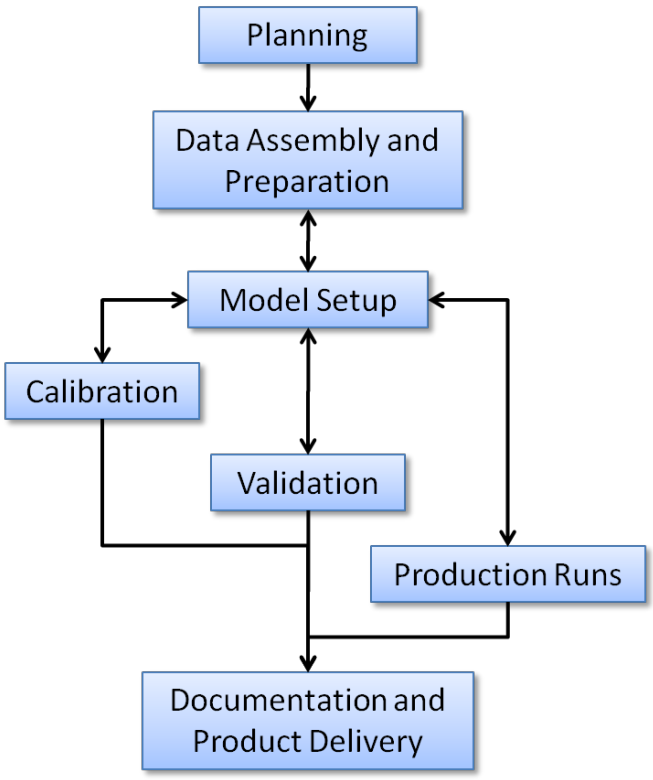


Figure 1-1. General steps or stages in a modeling study.

Planning Stage

If the modelers and key personnel involved are not familiar with the study area, then a site visit should be made. At that time, the study sponsors can point out concerns and various features of interest, such as conditions of jetties and shoreline, general hydrodynamics, and type of sediment. Those with local knowledge can be interviewed to learn about the presence of sand shoals and current conditions. At this point, a meeting of the modeling team and project delivery team is beneficial to review modeling objectives and essential features governing the waves, circulation, and sediment transport at the site. Several bathymetry grids may need to be developed to represent a past condition, existing condition, and the engineering alternatives under consideration. At this stage it is also important to conduct a

literature review and define the scope of modeling work, timelines, milestones, budgets, and group responsibilities.

Data Assembly and Preparation

The first step to start a CMS modeling project is to gather and assemble data. The important data can include geometry, bathymetry, shoreline position, location and configuration of coastal structures, oceanographic and atmospheric data, and sediment characteristics in the study domain. If existing, the datasets from previous modeling studies at the project site should be reviewed and mined for useful information. In addition, historic and recent aerial photographs, preferably vertical photographs that can be rectified, should be assembled, from which locations of structures and shorelines can be obtained. Reliable shoreline and bathymetry data with geo-referenced images can assist to design and build a model grid system. It is essential to correctly process bathymetric data and set a domain volume for a model grid. Volume errors related to shoreline and bathymetry data could affect the tidal prism estimate and flow calculations in the model, especially near inlet entrance, navigation channel, and coastal embayment.

All the assembled data would provide sufficient information to describe the physical processes occurring in the study area and lay a solid foundation for configuring and calibrating/validating the CMS later. Based on the data coverage, a model simulation period will be selected.

Model Setup

The model setup here refers to the process of grid generation, selections of model parameters, and specifications of model forcing. It may need to be done multiple times during a project for the purposes of (1) tests, (2) calibration, (3) validation, and (4) production or project alternatives. Because all those simulations may be set up for different time periods, the model will require corresponding forcing, boundary conditions, and possibly bathymetry and bottom characteristics.

Initial test runs may be conducted to guide the user in determining proper model setup for different cases such as grid resolution, domain extent, boundary condition types, and time steps. Test runs can also be conducted in the calibration, validation, and production stages of the study, which are

a good modeling practice for improving the quality of the model results and detecting errors early. For a new modeling project it may be helpful to start simple with basic options and fewer processes and slowly increase the complexity until the desired level is achieved. This way, the problems with the setup can be identified more easily.

Calibration

Model should be calibrated by comparing calculated results against measurements. The model output time intervals shall be consistent with data sampling frequency to avoid bias in model-data comparisons. For time series comparisons, the grid depth at the model output location should be similar to the data sampling station. Flow conditions can be misrepresented if the model and the data station have different depth and location. Through the calibration process, a set of model parameters, and model geometry and bathymetry may require adjustments.

Validation

A calibrated model is run for model validation through model-data comparisons for a different period (preferably). A validation process proves that the model can be applied to the study area and provides reasonable results.

Production

The production runs can be conducted after the model calibration and validation. Those runs should include sensitivity tests - various experiments with different model parameters and different model forcing, and alternative conditions – modifications and additions of coastal structures and adjustments of model configuration.

Documentation and Product Delivery

As the final step of a CMS modeling project, extensive analysis of model output will be conducted and the study findings will be properly documented. This stage also includes the archive of the model I/O information, technology transfer, and product delivery.

System Requirements

The CMS runs on Personal Computers (PC's). The following system requirements are needed for the CMS and SMS interface:

- Operating System: Windows XP, 7 or 8
- RAM (Rapid Access Memory): 2 GB (3 GB recommended)
- Graphics card: OpenGL 1.5 or higher enabled
- Screen resolution: 1024x768 (1280x1024 or higher recommended)

Note:

- The RAM requirement is needed for both running the model and the SMS interface.
- The graphics card and screen resolution are required for the SMS interface.

2 CMS-Flow Model Setup

Overview of Input and Output

The CMS-Flow (and also for most morphodynamic models) input and output can be classified as:

1. Input
   a. Control file (model parameters, options, pointers to other files).
   b. Geometry (grid).
   c. Initial condition (bathymetry, bed composition, etc.).
   d. Boundary conditions
2. Output
   a. Transient solution
      i. Global (for the whole domain)
      ii. Point (at a single point or cell)
   b. Statistical parameters
   c. Hot start (may include internal variables)
   d. Diagnostic Files

A detailed description of the CMS input and output files is provided below.

Input Files

There are at least three required files to run CMS-Flow: (1) the *Card File* (*.cmcards), (2) the *XMDF Grid File* (*_grid.h5), (3) and the *XMDF Model Parameters File*. If a telescoping grid is used then an additional ASCII *Telescoping Grid File* is required. For a description of types of Cartesian grids available in CMS-Flow see section Cartesian Grids. If any input datasets such as the Manning's coefficient or bed composition are specified in separate XMDF files then the file name and path must be specified in the *Card File*. Using separate input files for user-defined datasets makes it easier to make project alternatives or conduct sensitivity studies. The user-defined datasets can be easily created and exported from SMS. The procedure is described in detail in subsequent sections. A brief description of the CMS input files is provided in the table below.

Table 2-1. CMS-Flow Input Files

File	Name	Description
Card File	*.cmcards	File name including path for the CMS-Wave sim file.
XMDF Grid File	*_grid.h5	Specifies the coordinates, and depths of the CMS-Flow grid(s). If a telescoping grid is used, multiple grids are saved for each refinement level.
XMDF Model Parameters File	*_mp.h5	Specifies boundary forcing information, wind forcing, and geographical coordinates.
Telescoping Grid File	*.tel	Specifies the cell coordinates, dimensions, and depths for the telescoping grid file. Only saved in the case a telescoping grid is used.
User-defined Dataset Files	*.h5	Determines the extrapolation distance used for flow variables on the wave grid.

CMS-Flow Card/Control File

In CMS-Flow, the *Control File* is an ASCII text file referred to also as the *Card File* because it contains a list of cards used to specify model input such as parameters, coefficients, options, etc. A card is simply a unique character string which the model uses to identify a specific model input. The *Card File* is the main input file which sets input parameters and points to other files. In general most of the “light” input data is specified in the *Card File*. Light data refers to input parameters which is relatively small in size and can easily be specified by the user in a few lines such as global parameters. Any input that is relatively large, such as time series, and spatially varying datasets are referred to as “heavy” data. Heavy data is stored in separate files such as the *Grid File*, *Model Parameters File*, or other user-specified files. The heavy data is then simply pointed to in the *Card File*.

Example 2-1. CMS-Flow Cards.

```
CMS_VERSION             4.0        !Version of input card file
BATHYMETRY_DATASET      "Flow_Shark_grid.h5" "Flow_Shark/Datasets/Depth"
GRID_ANGLE              13.0 'deg'    !Clock-wise from X-axes
TIME_LIST_1             2  0.0 3.0 1.0  3.0 6.0 0.5
```

Formatting Rules and Conventions

The CMS-Flow Card File is a simple ASCII file with the following simple formatting rules and conventions:

1. Commented lines should be preceded by the characters ! or # or ignored. Comments can start at any column within the card file.
2. Cards are always written in capital letters.
3. Any character string in all capital letters after a card is an argument from a list options for that specific card (e.g. ON or OFF).
4. Card arguments are read in free-format. This means that input arguments should always be separated by at least one blank space or a comma.
5. Cards should always begin on a new line (i.e. only one card is allowed per line).
6. Character strings do not need to be enclosed in single or double quotation unless they contain white spaces. No characters besides letters (no accents) and numbers are allowed in character strings except: '(', ')', and '-'.
7. The end of the card file is specified by the END_PARAMETERS card. Any statements after this card are ignored.

Style Recommendations

The following recommendations are optional and provided for readability and to avoid commonly made formatting errors.

1. Single line cards should start at the first column unless the card is specified within a block in which case the card should be indented by 2 to 3 spaces.
2. Input arguments should start at column 37 for readability.
3. Double quotes should be used for file names and paths
4. Single quotes should be used for units.

Table 2-2. CMS-Flow Card notation.

Notation	Description
!	Indicates the beginning of a comment. Comments may begin at any column. Recommended for inserting headers and comments.
#	Indicates to ignore the rest of a line. Recommended for commenting cards.
ON	Activates a model feature
OFF	Deactivates a model feature
" "	Indicates a character string such as a file name or path
' '	Indicates a character string for input argument units such as 'm^3/s'

Many input cards have relationships to one another. For example a card may only be applicable if another card is specified. This is referred to here as a dependency. Dependencies are hierarchal in nature and may be listed either downwards or upwards in the dependency tree. One example of a dependency is the angle and repose and avalanching activation cards. The angle of repose is dependent on the avalanching being activated; otherwise it is not applicable. When two cards cannot exist together because they conflict with one another, this is referred to as an exclusion. An example of an exclusion is the bottom roughness specification using both a Manning's n and bottom friction coefficient. The bottom roughness can only be specified once and therefore the input cards used to specify the Manning's and bottom friction coefficient would conflict with one another. In the tables describing the input cards it is convenient to use to establish a notation for the card input formats.

Table 2-3. CMS-Flow Card format operators.

Notation	Description
`[]`	Indicates a required input value (card or argument)
<	Specifies the range of an argument to be less than a value
>	Specifies the range of an argument to be greater than a value
<=	Specifies the range of an argument to be less or equal to a value
>	Specifies the range of an argument to be greater or equal to a value
=	Specifies the value(s) of an input argument

Table 2-4. CMS-Flow Card Format key word description.

Notation	Description
`begin`	Block structure begin statement
`end`	Block structure end statement
`card`	Character string for a card
`arg`	Variable name for a card argument
`units`	Unit type for input argument (generally optional)
`options`	List of card argument options
`type`	Argument type as floating (*`float`*), integer (*`int`*), or character (*`char`*)
`float`	Argument type as a real or floating

int	Argument type as an integer
char	Argument type as a character string
limits	Limits of the input argument if they exist
default	Argument default value if it exists. If an automatic default value is available then the automatic mode can be used.
dependencies	List of cards for which there exist a dependency
exclusions	List of cards for which there exist an exclusion
symbol	Mathematical symbol used for an input argument
required	Indicates whether in input argument is required or not
typical	Specifies a typical value or range for an input card.

Below is an example of how the CMS-Flow cards are specified including comments at the end of each line.

Table 2-5. Example input parameters and card formats.

Input	Format	Notes
CMS card file version	[card=CMS_VERSION] [name=**version**, type=float]	Specifies the version number of the CMS Card file.
Implicit weighting factor, θ	[card=IMPLICIT_WEIGHTING_FACTOR] [arg=**ImpWghtFac**, type=*float*, limits=(0.0<= **ImpWghtFac**<=1.0), default=0.0, required=false]	**ImpWghtFac** =0 is first order and **ImpWghtFac** =1 is second order.
Sediment fall velocity	[card=FALL_VELOCITY, comment='!units are optional'] [arg=**SedFallVel**, type=*float*, limits=(**SedFallVel**>0.0), required=false, default=*automatic*, dependencies=*none*, exclusions=FALL_VELOCITY_FORMULA] [units=**SedFallVelUnits**, type=*char*, options=**VelocityUnits**, default='m/s']	
Sediment fall velocity formula	[card= FALL_VELOCITY_FORMULA] [arg=SedFallVelForm, type=char, options=(SOULSBY,WU), exclusions=FALL_VELOCITY]	
Output time list	[card=(TIME_LIST_1, TIME_LIST_2, TIME_LIST_3, TIME_LIST_4)] [type=int, name=nint, limits=(nint>=0)] for(i=1:nint, [name=tstart(i), type=float, limits=(tstart(i)>0.0)] [name=tend(i), type=float, limits=(tend(i)>=tstart(i))] [name=tinc(i), type=float, limits=(tinc(i)>=0.0)])	

Table 2-6. Definition of Temperature Unit Character String Variables.

TempUnits=(Celcius, Farhenheit, Kelvin, Rankin)
Celcius=(‘C’, ‘ºC’, ‘Celcius’)
Farhenheit=(‘F’, ‘ºF’, ‘Fahrenheit)
Kelvin=(‘K’, ‘ºK’, ‘Kelvin)
Rankin=(‘R’, ‘ºR’, ‘Rankin)

Table 2-7. Definition of Angle Unit Character String Variables.

AngleUnits=(**Radians, Degrees**)
Radians=(‘rad’, ‘radian’, ‘radians’
Degrees=(‘deg’, ‘degree’, ‘degrees’, ‘º’)

Table 2-8. Definition of Time Unit Character String Variables.

TimeUnits=(**Seconds, Minutes, Hours, Days, Weeks, Years**)
Seconds=(‘s’, ‘sec’, ‘second’, ‘seconds’)
Minutes=(‘min’, ‘minute, ‘minutes’)
Hours=(‘h’, ‘hr’, ‘hrs’ ‘hour’, ‘hours’)
Days=(‘d’, ‘dy’, ‘day’, ‘days’)
Weeks=(‘w’, ‘wk’, ‘week’, ‘weeks’)
Years=(‘y’, ‘yr, ‘yrs’, ‘year’, ‘years’)

Table 2-9. Definition of Length Unit Character String Variables.

LengthUnits=(**Micron, Millimeter, Centimeter, Kilometer, Inch, Feet**)
Micron=(‘um’, ‘micron’, ‘microns’)
Millimeter=(‘mm’, ‘millimeter’, ‘millimeters’)
Centimeter=(‘cm’, ‘centimeter’, ‘centimeters’)
Meter =(‘m’, ‘meter’, ‘meters’)
Kilometer =(‘km’, ‘kilometer’, ‘kilometers’)
Inch =(‘in’, ‘inch’, ‘inches’)
Feet =(‘ft’, ‘foot’, ‘feet’)

Table 2-10. Definition of Velocity Unit Character String Variables.

VelocityUnits=(**mmps, cmps, mps, kph, fps, knot, mph**)
mmps=(‘mmps’, ‘mm/s’, ‘mm/sec’, ‘millimeter/second’, ‘millimeters/second’)
cmps=(‘cmps’, ‘cm/s’, ‘cm/sec’, ‘centimeter/second’, ‘centimeters/second’)
mps=(‘mps’, ‘m/s’, ‘m/sec’, ‘meter/second’, ‘meter/second’)
kph=(‘kph’, ‘km/h’, ‘km/hr’, ‘kilometer/hour’, ‘kilometers/hour’)
fps=(‘fps’, ‘ftps’, ‘ft/s’, ‘ft/sec’, ‘feet/second’)
knots =(‘knot’, ‘knots’)
mph=(‘mph’, ‘mi/hr’, ‘miles/hour’)

Table 2-11. Definition of Volume Flux Unit Character String Variables.

VolumeFluxUnits=(**cms, cmm, cmh, cfs, cfm, cfh, cmsc, cmmc, cmhc, cfsc, cfmc, cfhc**)
cms=(‘cms’, ‘m^3/s’, ‘meter^3/sec’, ‘cubic meter/second’)
cmm=(‘cmm’, ‘m^3/min’, ‘meter^3/minute’, ‘cubic meter/minute’)
cmh=(‘cmh’, ‘m^3/hr’, ‘meter^3/hour’, ‘cubic meter/hour’)
cfs=(‘cfs’, ‘ft^3/s’, ‘ft^3/sec’, ‘cu ft/s’, ‘cubic foot/second’)
cfm=(‘cfm’, ‘ft^3/min’, ‘feet^3/minute’, ‘cu ft/s’, ‘cubic foot/minute’)
cfh=(‘cfh’, ‘ft^3/hr’, ‘ft^3/hour’, ‘cu ft/s’, ‘cubic foot/hour’)
cmsc=(‘cmsc’, ‘m^3/s/cell’, ‘meter^3/sec/cell’, ‘cubic meter/second/cell’)
cmmc=(‘cmmc’, ‘m^3/min/cell’, ‘meter^3/minute/cell’, ‘cubic meter/minute/cell’)
cmhc=(‘cmhc’, ‘m^3/hr/cell’, ‘meter^3/hour/cell’, ‘cubic meter/hour/cell’)
cfsc=(‘cfsc’, ‘ft^3/s/cell’, ‘ft^3/sec/cell’, ‘cu ft/s/cell’, ‘cubic foot/second/cell’)
cfmc=(‘cfmc’, ‘ft^3/min/cell’, ‘feet^3/minute/cell’, ‘cu ft/s/cell’, ‘cubic foot/minute/cell’)
cfhc=(‘cfhc’, ‘ft^3/hr/cell’, ‘ft^3/hour/cell’, ‘cu ft/s/cell’, ‘cubic foot/hour/cell’)

Note:

- The unit character strings are NOT case sensitive. Therefore, `'km/hr'` is equivalent to `'Km/hr'`.

It is recommended to become familiar with the *CMS-Flow Card File* and view it using a text editor which allows syntax highlighting such as Textpad (see Figure 2-1) or Notepadd++ (see Figure 2-2).

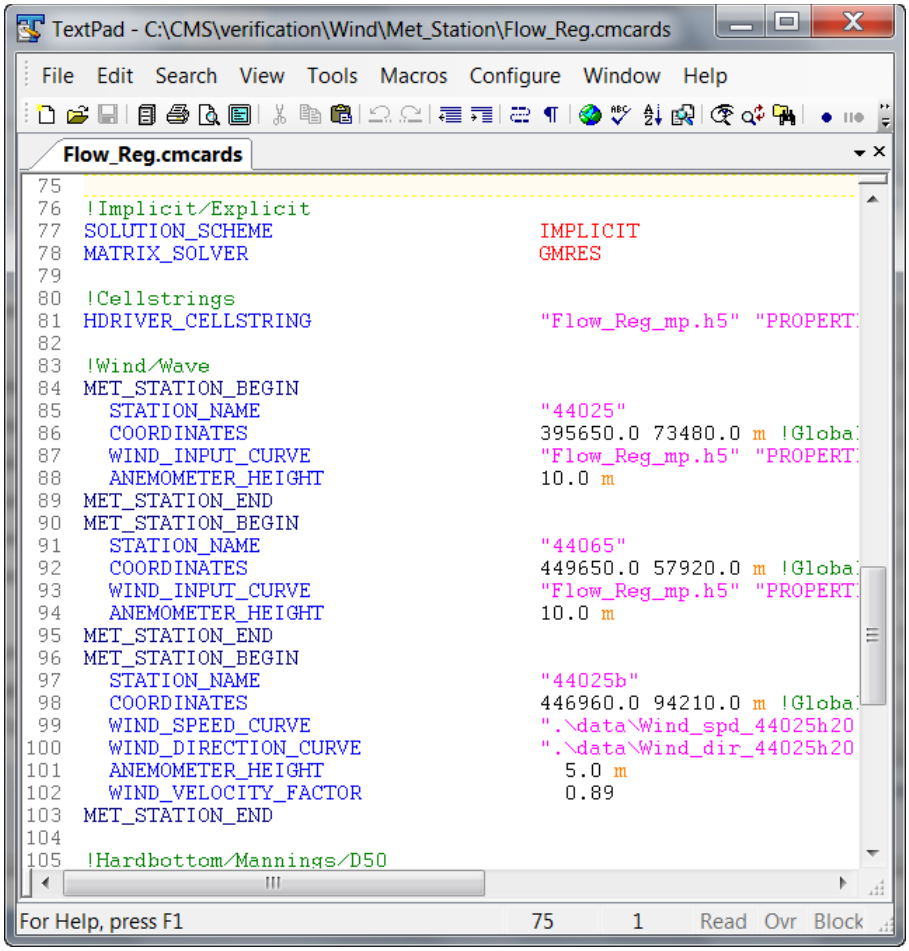


Figure 2-1. CMS-FlowCard File viewed in TextPad.

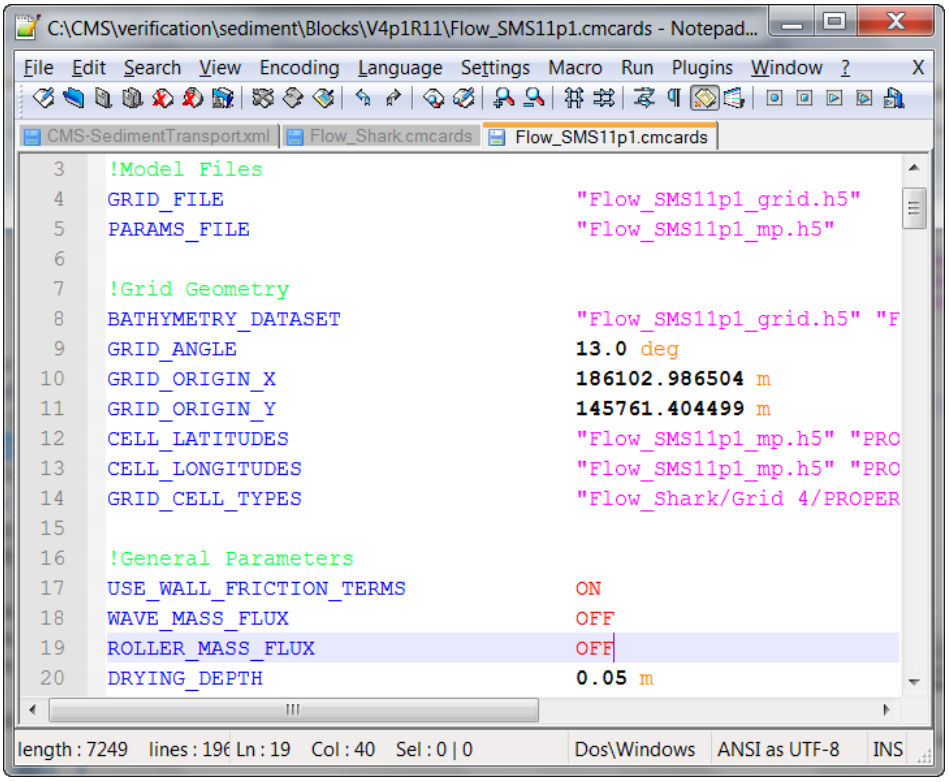


Figure 2-2. CMS-Flow Card File viewed in Notepad++.

Tips:

1. It is recommended to become familiar with the CMS-Flow card file and to learn how to manually edit and enter cards without having to use the SMS interface. When the user saves the CMS-Flow project in SMS, all of the input files are rewritten including the grid and model parameters file (which can be large). Therefore if only one parameter needs to be changed in the card file, it is much easier and faster to open the card file, manually edit the field, and re-save the card file instead of potentially having to reload the project in SMS, wait for the project to display, open the *Model Control Window*, edit the input parameter, and re-save the project.
2. It is recommended to view the CMS-Flow Card File in a text editor which supported user-defined syntax highlighting such as Notepad++, Textpad, and UltraEdit. Syntax definitions for Textpad and Notepadd++ are available from the CIRP wiki website at http://cirp.usace.army.mil/wiki/Utilities.
3. Since all of the statements after the END_PARAMETERS card are ignored. This section of the card file is useful for placing metadata or cards saving card sections when testing different model setups.

4. There are basically two types of cards in CMS: (1) single line cards, and (2) block declaration cards. Single line cards have input arguments specified on the same line as the card and in general to not depend on the order in which the cards are specified. Block declaration cards specify the beginning and end of a block. Blocks are used to group several input parameter statements. Blocks may contain multiple single line cards or other blocks.

Output Files

The minimum output files are the *XMDF Global Solution File* (*.h5), the Diagnostic File (CMS_DIAG.txt). By default, all of the solution variables are output to the same Global Solution File, but it is possible If *Observation Cells (Save Points)* are selected for output time series at individual cells, then additional ASCII files are written for each of the output variables. More information on the *Observation Cells* is provided in the section Output: Observation Cells (Save Points).

Table 2-12. CMS-Flow Output Files

File	Name	Description
XMDF Global Solution Files	*_sol.h5 *_wse.h5 *_vel.h5 *_visc.h5 *_wave.h5 *_morph.h5 *_trans.h5	XMDF file containing the time varying global solution including water levels, current velocities, etc. The different files are described in detail in the *Output* section of this document.
Diagnostic File	CMS_DIAG.txt	Name of the CMS-Flow Diagnostic File. Contains the model setup, time stepping history, subset of the screen output.
Observation Cells (Save Point) Files	*.txt	Specifies the coordinates, and depths of the CMS-Flow grid(s). If a telescoping grid is used, multiple grids are saved for each refinement level.
Super ASCII Global Solution Files	*.sup *.xy *.dat	SMS ASCII file containing the time varying global solution including water levels, current velocities, etc.
Tecplot ASCII Global Solution Files	*.dat *.his	Tecplot ASCII file containing the time varying global solution including water levels, current velocities, etc.

Overview of SMS Interface

All of the CMS-Flow model parameters, settings, and output options are controlled from the *CMS-Flow Model Control Window* (see Figure below). The window also has a section for Advanced Cards in which features and options can be entered which have not been incorporated into the SMS interface yet or more advanced model features more experienced users.

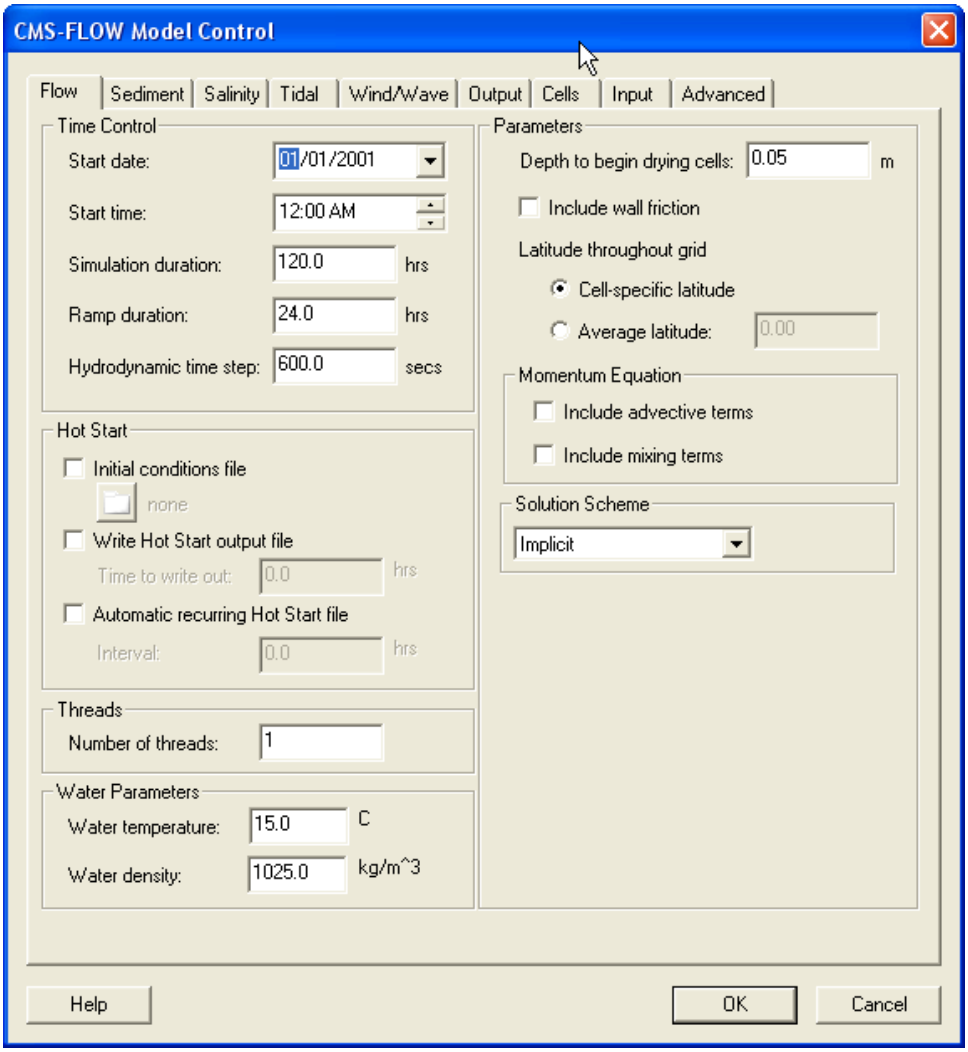


Figure 2-3. CMS-Flow Model Control: Model Parameters tab for specifying time series output of CMS-Flow calculated datasets.

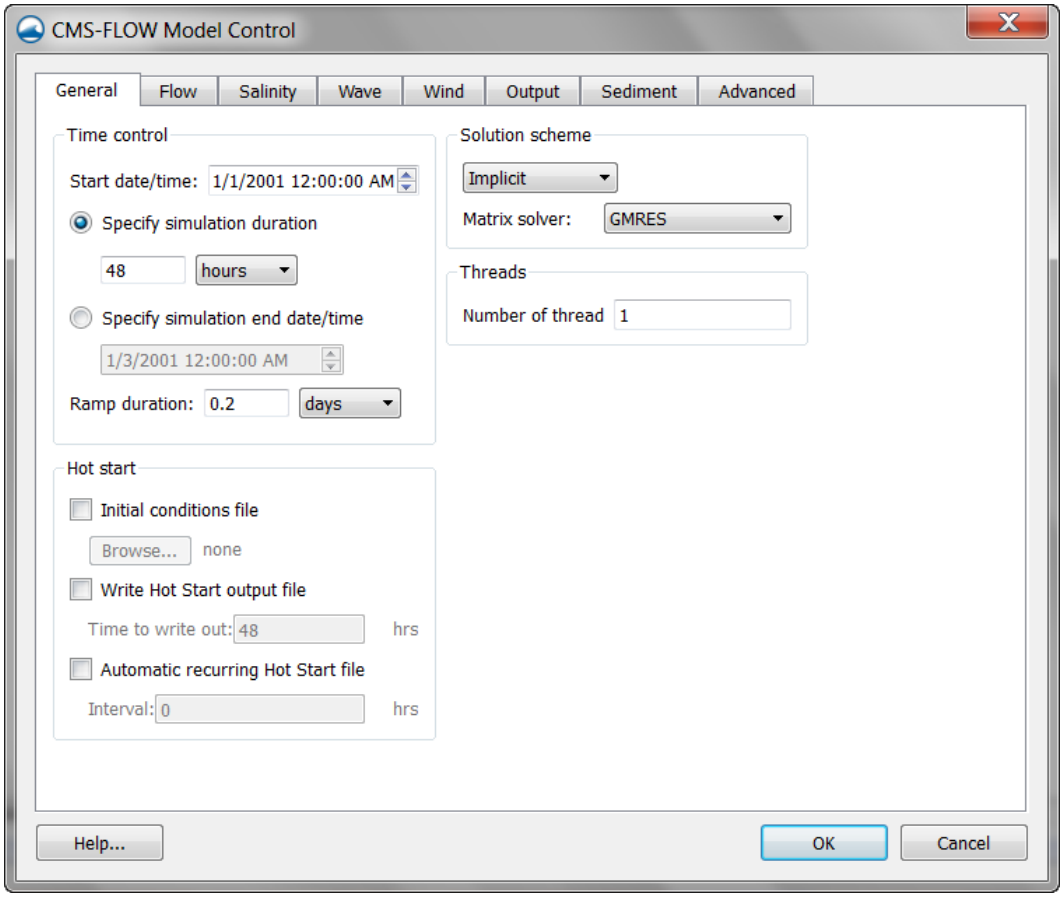


Figure 2-4. General Tab of th *CMS-Flow Model Control* window in SMS 11.1.

Geospatial Information

Horizontal Coordinate Systems and Conventions

CMS-Flow uses a local coordinate system in which all vector values are positive along the I and J axis (Figure 4-2). All output vector arrays are specified in the local coordinate system. Any input that is specified on the local grid must be specified in the local coordinate system (e.g. initial condition for currents, interpolated wave forcing, etc). If input vector arrays are specified on a different grid, such as a spatially variable wind field or waves on a CMS-Wave grid, then the vectors are assumed to follow the coordinate system of their native grid. The grid is always created in SMS with the origin is by default always at the lower left hand corner of the grid.

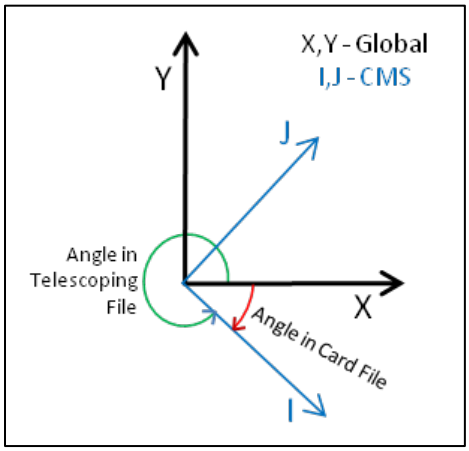


Figure 2-5. CMS-Flow local coordinate system and grid angle conventions used in the Card and Telescoping Grid Files. The I and J axis indicate the CMS grid axis, while the X and Y axis indicate the global coordinate system.

In the SMS, the term “projection” refers to a horizontal map projection such as “State Plane” or “Geographic”. Within the software, a global projection can be associated with a project. Individual datasets may also have their own projections, which if properly associated with their original reference, can be displayed without the global projection reference. Previous versions of the SMS software referred to projections as “coordinate systems” and reprojection as “coordinate conversion”.

The projection for the project can be specified from the *Edit* | *Projection* pull-down menu command. Changing the projection does not alter the coordinates of the project data. To change the projection of the individual

project data, the dataset must be reprojected to the common reference system. The CMS models operate in a Cartesian coordinate system such as State Plane or UTM.

Currently in SMS 11.1 the horizontal projection information in the grid file (*_grid.h5). The projection of the CMS-Flow simulation may be different from that of the grid and is specified in the CMS-Flow Control File. Specifying the projection information in the Control File is only necessary if any model input (such as wind or parent grids) is specified in any other projections. Otherwise the CMS assumes that all the horizontal projections are the same and no internal conversions are necessary. The table below describes the CMS-Flow cards used to specify the horizontal projection.

Table 2-13. CMS-Flow cards related to the single tidal water level boundary condition.

Input	Card Format	Notes
Start of Horizontal Projection Block	[begin=(HORIZONTAL_PROJECTION_BEGIN, HORIZ_PROJ_BEGIN), version=4.0, block=**HorizProj**, children=(**HorizDatum**,**HorizCoord**, **HorizZone**,**HorizUnits**)]	Begins the block structure for specifying the horizontal projection.
Horizontal Datum	[card=DATUM, parent=**HorizProj**] [arg=**HorizDatum**, type=char, options=(NAD27,NAD83,LOCAL), default= LOCAL]	Specifies the horizontal datum.
Horizontal Coordinate System	[card=SYSTEM] [arg=HorizCoord, type=char, options=(STATE_PLANE,UTM, GEOGRAPHIC,LOCAL), default=LOCAL, parent=**HorizProj**]	Specifies the horizontal coordinate system.
Horizontal Zone	[card=ZONE] [arg=HorizZone, type=int, default=none, parent=**HorizProj**]	Specifies the horizontal zone (only valid for UTM and State Plane Coordinate Systems).
Horizontal Units	[card=UNITS] [arg=**HorizUnits**, type=char, options=(METERS,FEET,DEGREES, RADIANS), default=METERS, parent=**HorizProj**]	Specifies the units of the horizontal coordinate system.
End of Horizontal Projection Block	[end=(HORIZONTAL_PROJECTION_END, HORIZ_PROJ_END), version=4.0, block=**HorizProj**, childs=(**HorizDatum**,**HorizCoord**, **HorizZone**,**HorizUnits**)]	Ends the block structure for specifying the vertical projection.

Example 2-2. Specifying the horizontal projection information.

```
HORIZONTAL_PROJECTION_BEGIN !can also use HORIZ_PROJ_BEGIN
   DATUM            NAD83         !NAD27 | NAD83 | LOCAL
   SYSTEM           STATE_PLANE   !UTM | STATE_PLANE | GEOGRAPHIC | LOCAL
   UNITS            METERS        !METERS | FEET
   ZONE             3104          !Long Island, New York, Only if necessary
HORIZONTAL_PROJECTION_END !can also use HORIZ_PROJ_END
```

A list of the supported horizontal datums, coordinate systems, or units are described in the following tables.

Table 2-14. Horizontal datums currently supported by CMS.

Datum	Description
LOCAL	Local datum. Set as the default datum. May be used when a coordinate system is not defined or working with idealized cases.
NAD27	Geographic coordinate system.
NAD83	Universal Transverse Mercator (UTM) coordinate system. 2D Cartesian coordinate system.

Table 2-15. Horizontal coordinate systems currently supported by CMS.

Coordinate Systems	Description
LOCAL	Local coordinate system. Set as the default coordinate system. May be used when a coordinate system is not defined or working with idealized cases.
UTM	Universal Transverse Mercator (UTM) coordinate system. 2D Cartesian coordinate system.
STATE_PLANE	State Plane coordinate system (also SPS or SPCS). Is divided into 124 geographic zones or coordinate systems designed for specific regions of the United States. Each State contains at least one zone.
GEOGRAPHIC	Geographic latitude and longitude coordinate system.

Table 2-16. Horizontal coordinate units (HorizUnits) currently supported by CMS.

Horizontal Units (HorizUnits)	Description
METERS	Units of meters only valid for State Plane and UTM coordinate system.
FEET	Units of feet only valid for State Plane and UTM coordinate system.
DEGREES	Units of degrees only valid for the Geographic coordinate system.
RADIANS	Units of radians only valid for the Geographic coordinate system.
SECONDS	Units of seconds only valid for the Geographic coordinate system.

Notes:

- In the CMS V4.1, it is not possible to convert between datums NAD27 and NAD83, therefore all of the input horizontal projections must have the same datum.
- Since the difference between the International Foot and US Survey Foot is 6.0960×10^{-7} m, the difference can be ignored for the purposes of coastal modeling and no differentiation is made between them in the CMS.

Vertical Coordinate System and Conventions

Water depths in CMS are positive and land elevations are negative. The vertical reference Datum in CMS is a local datum. Therefore, any vertical datum can be used by the user. However, it is important to note that the default initial water surface elevation is set to zero with respect to the local datum.

In choosing the vertical datum for CMS it is important to understand the differences between tidal and geodetic datums. It is common to use both types of datums. However, the most appropriate datum type depends on the project application. Tidal datums are a standard elevation defined by a tidal statistic such as the arithmetic mean of mean high water and mean low water over a tidal epoch known as the Mean Tide Level or MTL. Therefore, tidal datums vary spatially and in time. This is an important difference with respect to Geodetic datums such NAVD88 which are fixed reference elevations determined by geodetic leveling. Tidal datums should not be used for projects where the tidal datums vary spatially. It is common for bays to have tidal setup or super elevation of the mean water level caused by wind, fresh water inflow, and bottom friction. This will cause a difference in the MTL in the bay with respect to the ocean as illustrated in Figure 4-3.

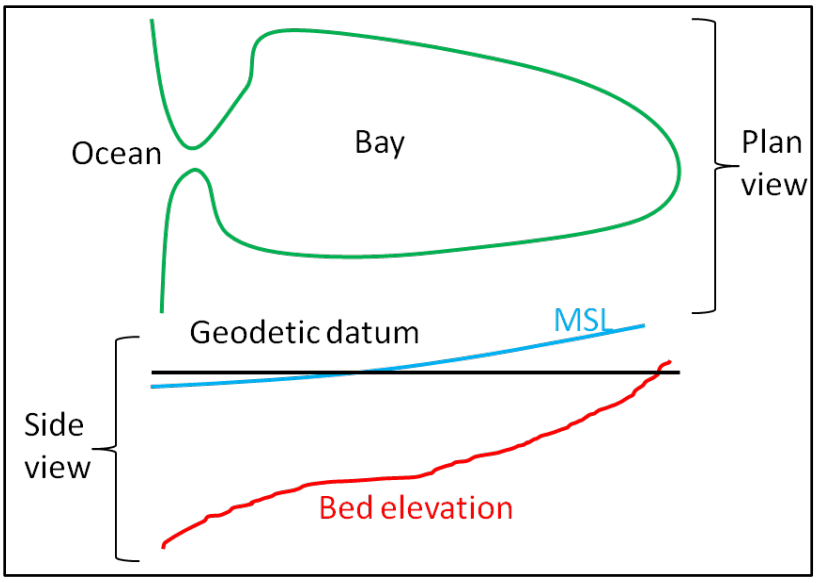


Figure 2-6. Schematic of showing the difference between MSL (tidal) and geodetic datums.

Currently, in CMS V4.1 the vertical projection information is only specified for metadata purposes and no internal conversions are performed yet. In the future, the goal is to be able to read in the vertical datum conversion conversions either from the model input files or to get them from an external database such as VDatum (http://vdatum.noaa.gov/welcome.html). A description of the cards used to specify the vertical projection is provided in the table below.

Table 2-17. CMS-Flow cards related to the single tidal water level boundary condition.

Card	Arguments	Description
Begin Vertical Projection Block	[begin=VERTICAL_PROJECTION_BEGIN, block=**VertProj**, version=4.0, childs=(**VertDatum**,**VertUnits**, **VertOffset**)]	Begins the block structure for specifying the vertical projection.
Vertical Datum	[card=DATUM, parent=**VertProj**] [name=**VertDatum**, type=char, options=(NGVD27,NAVD88,LOCAL, MSL,MTL,MLW,MLLW,MHW,MHHW), default=LOCAL]	Specifies the vertical datum.
Vertical Units	[card=UNITS, parent=**VertProj**] [name=**VertUnits**, type=char, options=('m','ft'), default='m']	Specifies the units of the vertical coordinate.
Vertical Offset	[card=OFFSET, comment="positive is upwards"] [name=**VertOffset**, type=float, limits=none, parent=**VertProj**]	Specifies the vertical offset from the datum specified. Useful for referencing a local datum to a known vertical datum. Positive is upwards and negative is downwards.
Ends Vertical Projection Block	[end=VERTICAL_PROJECTION_END, block=**VertProj**,version=4.0, childs=(**VertDatum**,**VertUnits**, **VertOffset**)]	Ends the block structure for specifying the vertical projection.

Example 2-3. Vertical projection specification.

```
VERTICAL_PROJECTION_BEGIN !Optional
  DATUM           NAVD88       !NGVD29 | NAVD88 | LOCAL
  UNITS           'm'        !'m' | 'ft'
  OFFSET          0.0 'm'         !Positive is upwards
VERTICAL_PROJECTION_END
```

The vertical offset is useful for referencing a local datum to a known vertical datum as in the example above. In CMS, the default initial water level

(can be specified otherwise using a hot start file) and therefore it is common practice to use a vertical datum which is close to MSL. However, because MSL is a tidal datum and therefore varies spatially, it is more accurate to use a geodetic datum with an offset, so that the zero reference elevation of the local datum is close to MSL.

Table 2-18. Horizontal datums currently supported by CMS.

Datum	Description
LOCAL	Local datum. Set as the default datum. May be used when a coordinate system is not defined or working with idealized cases.
NGVD29	National Geodetic Vertical Datum of 1929
NAVD88	National Geodetic Vertical Datum of 1988
MSL	Mean Sea Level
MTL	Mean Tidal Level
MLW	Mean Low Water
MLLW	Mean Lower Low Water
MHW	Mean High Water
MHHW	Mean Higher High Water

Table 2-19. Horizontal coordinate units currently supported by CMS.

Units	Description
METERS	Units of meters only valid for State Plane and UTM coordinate system.
FEET	Units of feet only valid for State Plane and UTM coordinate system.

Notes:

1. If the project site is a small section of coastline with a small harbor or structures, then it is generally ok to use a tidal datum.
2. If the project site includes an estuary, it is recommended to use a geodetic datum.
3. When using a geodetic datum in combination with tidal constituent forcing, it is recommended to offset the entire bathymetry so that the geodetic datum matches the offshore MSL. The reason for this is that tidal constitu-

ent boundary assumes that the reference datum is MSL. In CMS V4.1 it is possible to specify the MSL with respect to the CMS-Flow vertical datum.

4. When using a geodetic datum in combination with water level time series, it is important to make sure the time series has the same geodetic datum as the bathymetry. In addition, an initial water level may be specified which is close to MSL (see the Hot Start section for details).
5. The units of the horizontal coordinate system should always be set to meters. By default, in SMS the horizontal units are set to feet. It is recommended to change the units to meters and save this setting by clicking on the menu *File* | *Save Settings*. This will save the horizontal projection and will be set every time SMS opens.
6. A good merged bathymetric dataset is often the result of spending 50% of the time it takes to get the model up and running just on checking and cleaning up the bathymetry alone. If a significant amount of time is spent paying attention to detail, there will be less instability problems in the hydrodynamic model which are difficult to trace back to the source.
7. Always double check that you have calculated your datum corrections very carefully by analyzing and comparing to nearby datasets.
8. When merging bathymetry, try both methods to combine the datasets and evaluate for depth consistency and dataset coverage.
9. Save often, and occasionally save new versions of your merge-testing in new folders. If you delete too much of one dataset, its useful to have the older, datum converted and unaltered datasets.

Cartesian Grids

SMS supports regular and nonuniform Cartesian grids as well as regular and stretched telescoping grids (see Figure 4-4).

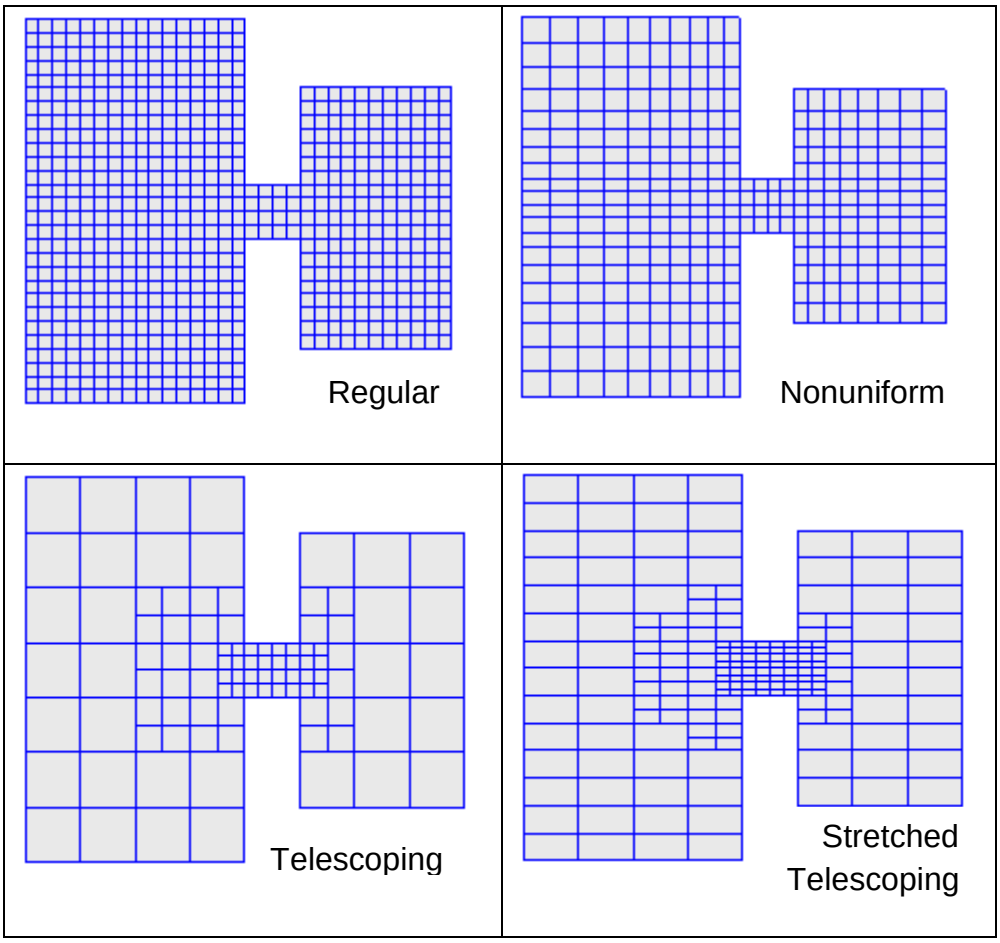


Figure 2-7. Types of Cartesian grids supported by the SMS interface and CMS-Flow.

Regular Cartesian grids are the easiest to generate making them useful for feasibility type studies, simple field cases, or test runs for more complicated field cases. Nonuniform Cartesian grids allow local refinement by gradually varying the grid spacing. These grids are offer more flexibility than regular Cartesian grids for a relatively low additional cost in generation. Nonuniform Cartesian grids are available for both CMS-Flow and CMS-Wave. For large complex modeling domains, Telescoping grids offer the most flexibility by providing local refine through the subdivision or splitting of cells into four. In many cases, the hydrodynamics can be aligned with one of the Cartesian coordinates. For these cases, the number of grid cells can be reduced by using a stretched Telescoping grid. Currently in

SMS, the stretched Telescoping grids can only have a constant aspect ratio between the grid resolution in the *x* and *y* directions.

When a CMS-Flow project is saved, SMS writes an XMDF *Grid File* named "*_grid.h5". If the grid is non-telescoping, this file contains the *x* and *y* coordinates as well as the water depths. If a telescoping grid is saved, then the SMS saves multiple-grids in the *CMS-Flow XMDF Grid File* which correspond to different levels of resolution of the telescoping grid. This information is only used by SMS. The information used in CMS when saving a telescoping grid is saved in the *Telescoping Grid File* with the extension "*.tel". The *Telescoping Grid File* contains the cell coordinates, resolution, connectivity and water depths. A description of *XMDF Grid File* and the *Telescoping Grid File* along with Matlab scripts for reading these files are provided in Grid File (*_grid.h5) and Telescoping Grid File (*.tel) sections of Appendix A: Description of Input Files.

All Cartesian grids can be characterized by a grid angle (orientation), origin coordinates. These grid parameters are saved to the *Card File*. The table below provides a description of the CMS-Flow cards used to specify the grid information in CMS.

Table 2-20. CMS-Flow cards related to the grid files and datasets.

Input	Format	Notes
XMDF Grid File	[card=GRID_FILE] [name=**GridFile**, type=*char*, default="**CaseName**_grid.h5"]	Specifies the XMDF grid file
Telescoping Grid File	[card= TELESCOPING, CMS_version=4.0, SMS_version=11.0] [name=**TelFile**, type=*char*, default="**CaseName**.tel"]	Specifies the telescoping grid file name.
Bathymetric Dataset	[card=BATHYMETRIC_DATASET] [name=**BathyFile**, type=*char*, default=**GridFile**] [name=**BathyPath**, type=*char*, default="**CaseName**/Datasets/Depth"]	Specifies the bathymetric dataset file and path.
Grid Angle or Orientation	[card=GRID_ANGLE] [name=**GridAngle**, type=*float*] [name=**GridAngleUnits**, type=*char*, default='deg']	Specifies the grid angle measured counterclockwise from the East direction to the grid *i*-axis (*x*-axis).
Grid Origin (x-coordinate)	[card=GRID_ORIGIN_X] [name=**GridOriginX**, type=*float*] [name= **GridOriginUnits**, type=*char*, default='m']	Specifies the x-coordinate for the Cartesian grid
Grid Origin (y-coordinate)	[card=GRID_ORIGIN_Y] [name=**GridOriginY**, type=*float*] [name=**GridOriginUnits**, type=*char*, default='m']	Specifies the y-coordinate for the Cartesian grid
Cell Center Latitudes	[card=CELL_LATITUDES] [name=**LatFile**, type=*char*, default=**MPfile**] [name=**LatPath**, type=*char*, default="**CaseName**/Datasets/Depth"]	Specifies the latitudes at the cell centers.
Average Latitude	[card=AVERAGE_LATITUDE] [name=**AvgLat**, type=*float*] [name=**AvgLatUnits**, type=*char*, default='deg']	Specifies the average latitude for the grid which is used for the Coriolis parameter.

The example below shows the Cartesian grid parameters defined in the CMS-Flow Model Control File.

Example 2-4. Grid geometry specification.

```
!Grid Geometry
BATHYMETRY_DATASET    "Flow_Shark_grid.h5" "Flow_Shark/Datasets/Depth"
GRID_ANGLE            13.0 deg    !Clock-wise from X-axes
GRID_ORIGIN_X         186102.986504 m
GRID_ORIGIN_Y         145761.404499 m
CELL_LATITUDES        "Flow_Shark_mp.h5" "PROPERTIES/Model Params/Lats"
```

If specified and average latitude is calculated for the grid and used to estimate the Coriolis parameter. By default the cell latitudes are stored in the XMDF *Model Parameters File*. For additional details on this file see XMDF CMS-Flow *Model Parameters File* (*_mp.h5).

Notes:

1. The grid angle specified in the *Card File* and the *Telescoping Grid File* use different conventions (unfortunately). The angle in the Card file is measured counter-clockwise from the East direction to the grid *x*-axis, while the angle in the *Telescoping Grid File* is measured clock-wise also from the East direction to the grid *x*-axis.
2. By default the *XMDF Grid File* also contains the bathymetry and other spatial datasets including the bottom friction, hard bottom, D50, etc. However, these datasets can be specified in separate files.
3. If a telescoping grid is used, the bathymetry is included by default in both the *XMDF Grid File* and the *Telescoping Grid File*. However, the depths in the in Telescoping Grid have -999 values for inactive cells. These cells are internally removed from the computational grid. Therefore, when using telescoping grids, only the depth values in the *Telescoping Grid File* are used by CMS.

The SMS Cartesian Grid Module has several tools for creating and editing Cartesian grids. A brief description of the tools is provided below

Create 2D Grid Frame

The 2D Cartesian Grid Frame (purple box) creation tool allows the user to visually specify the location and orientation of the grid in space. To create a grid frame, simple click three times to the desired length and width dimensions. SMS will complete the rectangle.

Select 2D Grid Frame

A 2D Cartesian Grid Frame can be modified with this selection tool. Once the frame is generated, the edges can be modified, the grid can be rotated (by selecting a small circle usually found in the lower left part of the grid), and the grid can be selected (middle black square) for manipulating location or opening the Grid Properties dialog box.

Select Cell

The *Select Cell* tool is used to select a grid cell. A single cell is selected by clicking on it. A second cell can be added to the selection list by holding the SHIFT key while selecting it. Multiple cells can be selected at once by dragging a box around them. A selected cell can be de-selected by holding the SHIFT key as it is clicked. When a single cell is selected, its *Z* coordinate is shown in the Edit Window. The *Z* coordinate can be changed by typing a new value in the edit field, which updates the depth function. If multiple cells are selected, the *Z* coordinate field in the Edit Window shows the average depth of all selected cells. If this value is changed, the new value will be assigned to all selected cells. With one cell selected, the Edit Window shows the cells *i*,*j* location. With multiple cells selected, the Edit Window shows the number of selected cells.

Select Row/ Select Column

The *Select Row* and *Select Column* tools are used to select cell rows and columns, respectively. Multiple rows and columns are selected in the same manner as selecting multiple individual cells: holding the SHIFT key, etc.

Insert Column/ Insert Row

When the *Insert Column* or *Insert Row* tools are active, clicking within a cell splits the row/column containing the selected cell, creating a new row or column in the grid. The *Z*-values of all split cells are the same as the original cells' values.

Drag Column/ Drag Row Boundary

The position of the edge of rows or columns in a grid can be changed with the *Drag Column* or *Drag Row* tools. These tools make one column/row narrower while making its neighbor wider. These tools allow for manual specification of the resolution in specific portions of the grid. Note that depth values are not adjusted, so significant dragging of boundaries should be avoided or depths should be re-interpolated after the boundaries are modified.

Create Cell String

The *Create Cell String* tool allows the modeler to group a string of cells together for the purpose of assigning boundary conditions. Cell strings are created automatically around water boundaries when a grid is generated. The user may create others as desired or delete and replace the automatically generated cell strings. When the *Create Cell String* tool is active, the modeler selects each cell to be added to the string. By holding down the SHIFT key, all boundary cells between the previously selected cell and the selected cell are added to the cell string.

Select Cell String

To specify a boundary condition, the modeler must create a cell string along the desired boundary cells, and then select the cell string while the *Select Cell String* tool is active. Specification of a boundary condition for the selected cell string is conducted through the *Assign BC* dialog, which is accessed through the CMS-M2D pull-down menu.

Editing Spatial Datasets

Spatial datasets including the bottom roughness may be edited manually by selecting and modifying cells, using the *Data Calculator*, or by interpolating from a scatter set. Figure 2-10 shows an example where the Manning's *n* coefficient was increased for Shark River Inlet to account for the presence of vertical piles. To manually edit a spatial dataset

1. Click on the *Select Cell* tool in the Cartesian Grid module toolbar.
2. Select a single or multiple cells by either:
   a. Clicking on a single cell.

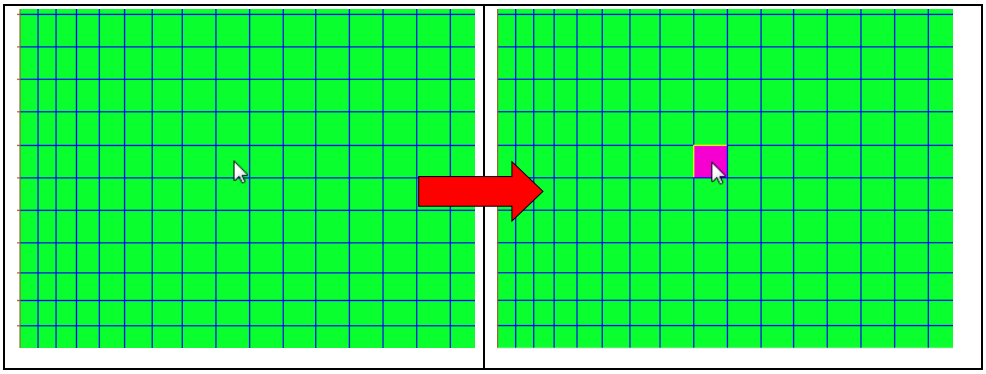


Figure 2-8. Selecting a single cell

   b. Use a selection box by single-left-clicking and holding to drag a selection box over a group of cells.

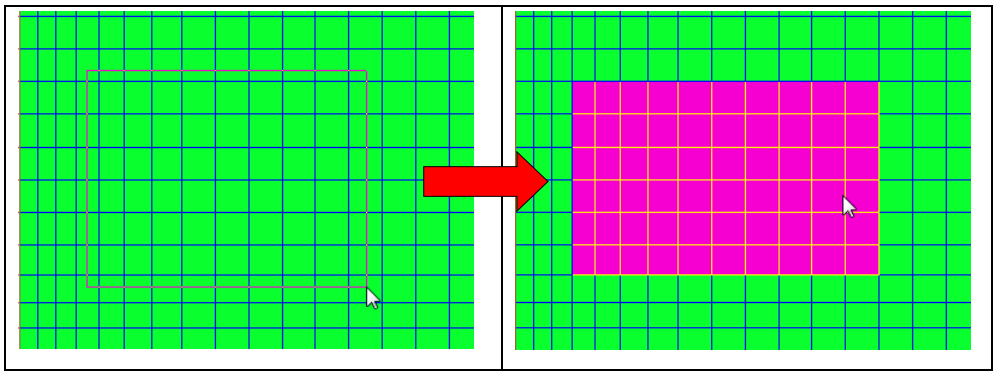


Figure 2-9. Selecting cell with a rectangle.

c. Use a selection polygon by clicking on the menu *Edit* and selecting the submenu *Select with Poly* to select a group of cells with a polygon (see Figure below).

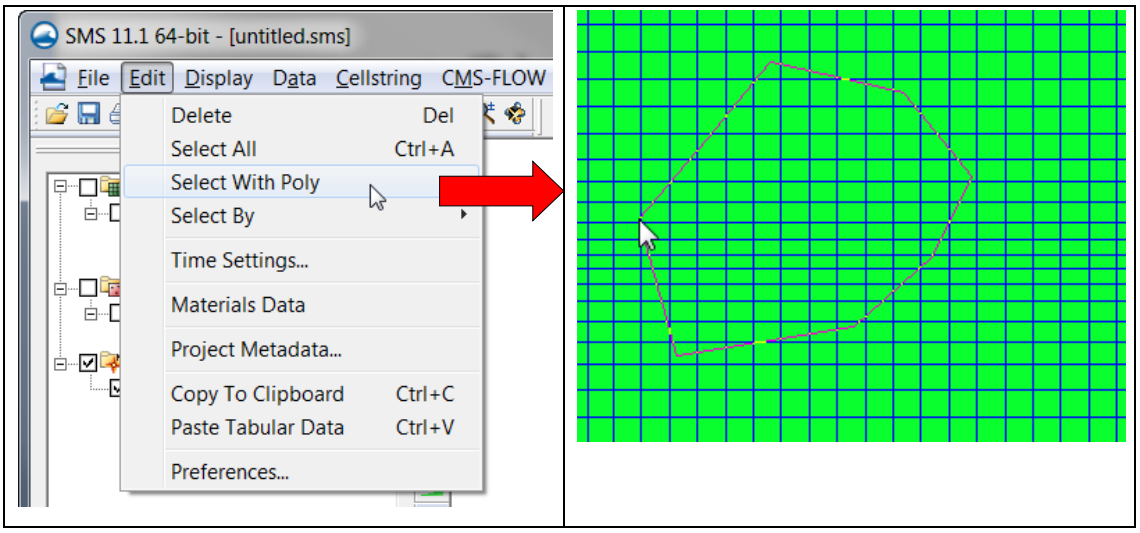


Figure 2-10. Selecting cells with a polygon in SMS 11.1.

d. Select cells by their value by clicking on the menu *Edit* and selecting the submenu *Select By* and then clicking the *Data Set value...* option (Figure 2-11).

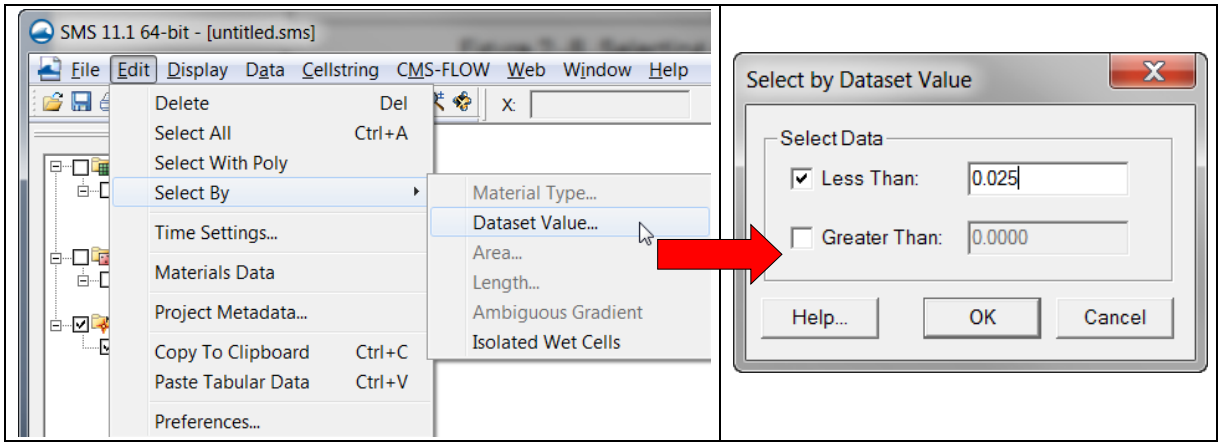


Figure 2-11. Selecting cells with a polygon in SMS.

If the dataset being modified has been linked to a field such as the Manning's n or the hard bottom in the CMS-Flow Control Window then the dataset will appear as editable in the SMS Edit Window (see Figure 2-12) and the user may proceed to Step 3. If the dataset does not appear as editable (i.e. the "S:" field is grayed-out), the dataset will need to temporarily set to the *Elevation* dataset in order to modify it. To set the dataset as the Elevation dataset follow the steps in Section Mapping a Spatial Dataset as the Elevation Dataset.

3. Enter a value in the "S:" field of the SMS Edit window (see Figure 2-12).

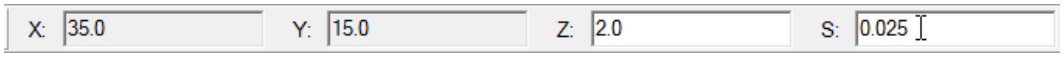


Figure 2-12. SMS Edit Window.

Figure 2-13 shows an example of a spatial dataset (Manning's n in this case) which has manually been modified.

Figure 2-13. Manning's *n* contours after modifying the areas under all three bridges.

Mapping a Spatial Dataset as the *Elevation* Dataset

The *Elevation* dataset is designated with the symbol Z preceding the dataset name. As the name indicates, the *Elevation* dataset is elevation of the computational grid and is used within SMS for example when visualizing a grid in three dimensions. In CMS-Flow the *Elevation* dataset is typically called "Depth". In some situations, it is useful to either temporarily designate a dataset as the *Elevation* dataset to be able to modify its values or to replace the *Elevation* dataset with a new one. To reset the *Elevation* dataset:

1. Click on the menu *Data* | *Map Elevation* (**Error! Reference source not found.**).

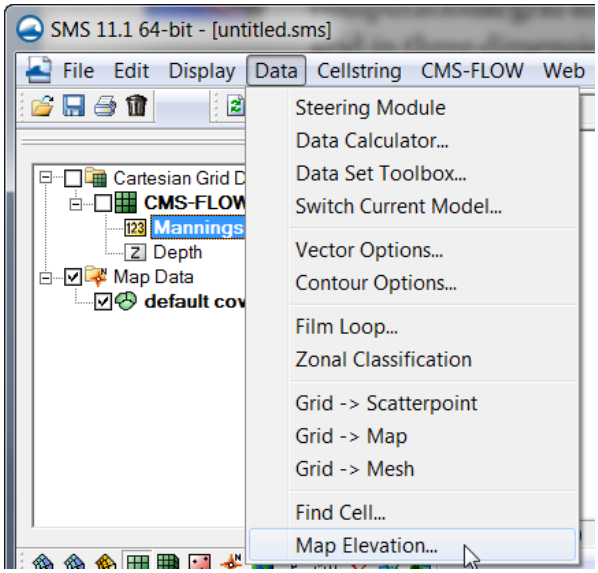


Figure 2-14. Selecting the Map Elevation menu option in SMS 11.1

2. In the SMS *Select Dataset* window, select the dataset to be converted as the *Elevation* dataset (Figure 2-15a) and then click on *Select* (Figure 2-15b). The select dataset friction dataset should now appear with the symbol Z.

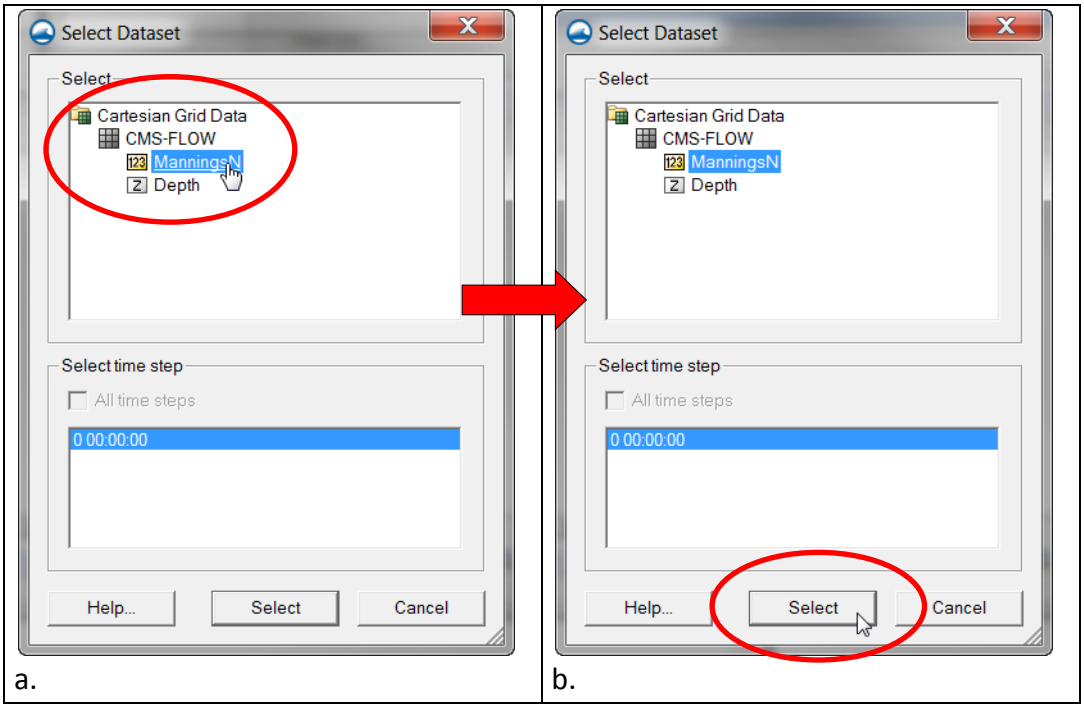


Figure 2-15. Selecting the Map Elevation menu option in SMS 11.1

Interpolating a Scatter Set to the Whole Grid

1. Select the scatter set.
2. Click on the menu Scatter | Interpolate to Cartesian Grid (see Figure 2-16).
3. In the Interpolation window, enter an extrapolation value, a name for the dataset and click OK (see Figure 2-17).

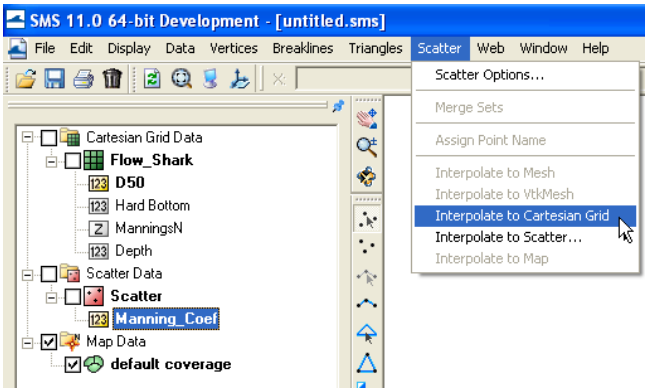


Figure 2-16. Interpolating a scatter set of Manning's *n* values to the CMS-Flow grid.

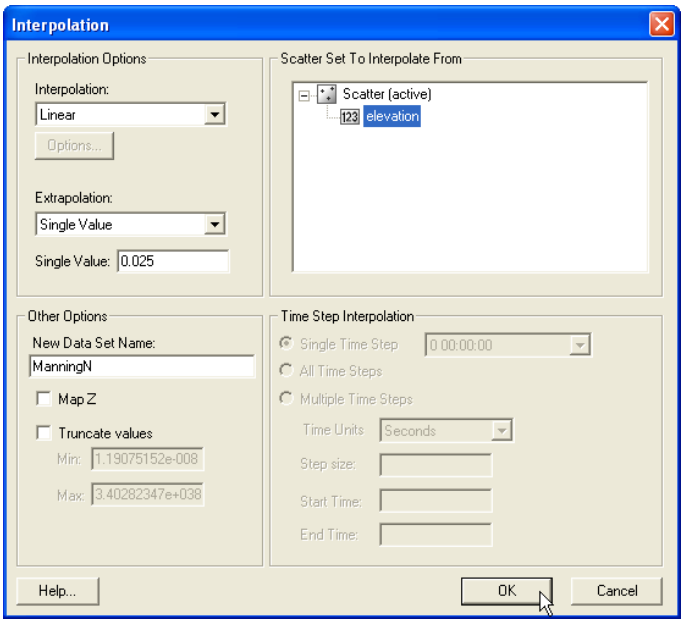


Figure 2-17. Interpolating a scatter set of Manning's *n* values to the CMS-Flow grid.

The dataset will be interpolated to the whole grid and grid cells outside of the scatter triangulation will be set to the extrapolation value.

Interpolating a Scatter Set to a Portion of Grid

In some cases, it is desirable to only interpolate a bottom roughness scatter set to a portion of the grid. However, it is only possible to interpolate portions of the grid from scatters for the *Elevation* dataset. The *Elevation* dataset is the dataset with a Z to the left. Therefore, it is necessary to *Map* the bottom friction dataset as the *Elevation* dataset, interpolate the scatter set, and then re-map the actual grid elevation (usually named *Depth*) dataset as the *Elevation* dataset. The instructions for *Mapping* a dataset as the *Elevation* dataset is outlines in Section Mapping a Spatial Dataset as the Elevation Dataset. To interpolate a bottom roughness dataset to a portion of a grid follow the instructions below.

1. Select the portion of the grid from which you want to interpolate the bottom friction dataset. This can be done by clicking and holding the left mouse button or by holding the Control key and single clicking so select the polygon and double clicking on the last polygon point (shown in Figure 2-18).
2. Once the correct cells are selected, right-click on the selected cells and select *Interpolate Bathymetry*.

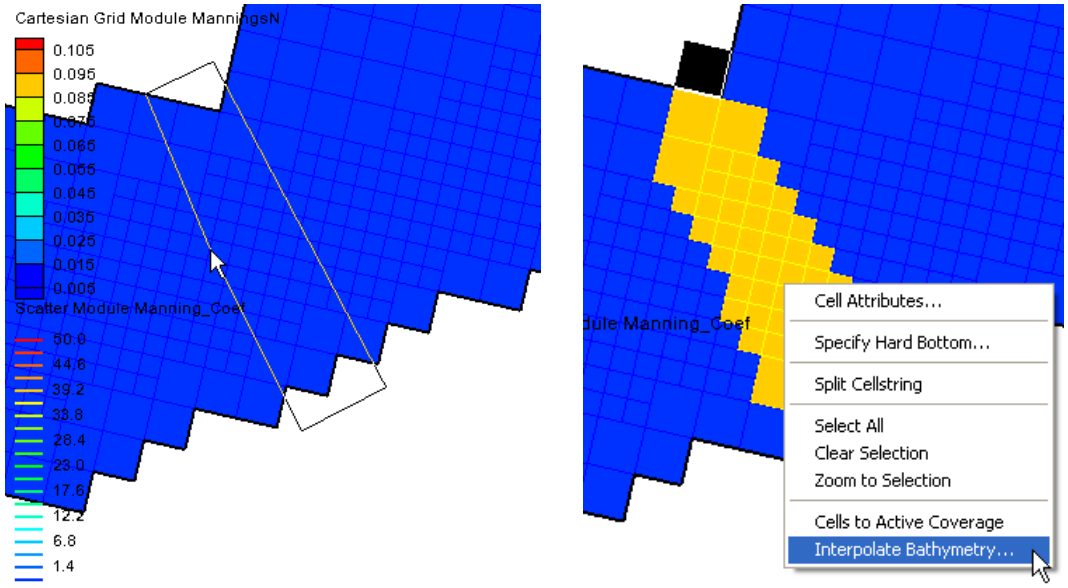


Figure 2-18. Mapping of a dataset (bottom friction in this case) as the Elevation dataset.

3. In the interpolation window, select and appropriate extrapolation value and click *OK* (see Figure 2-19).

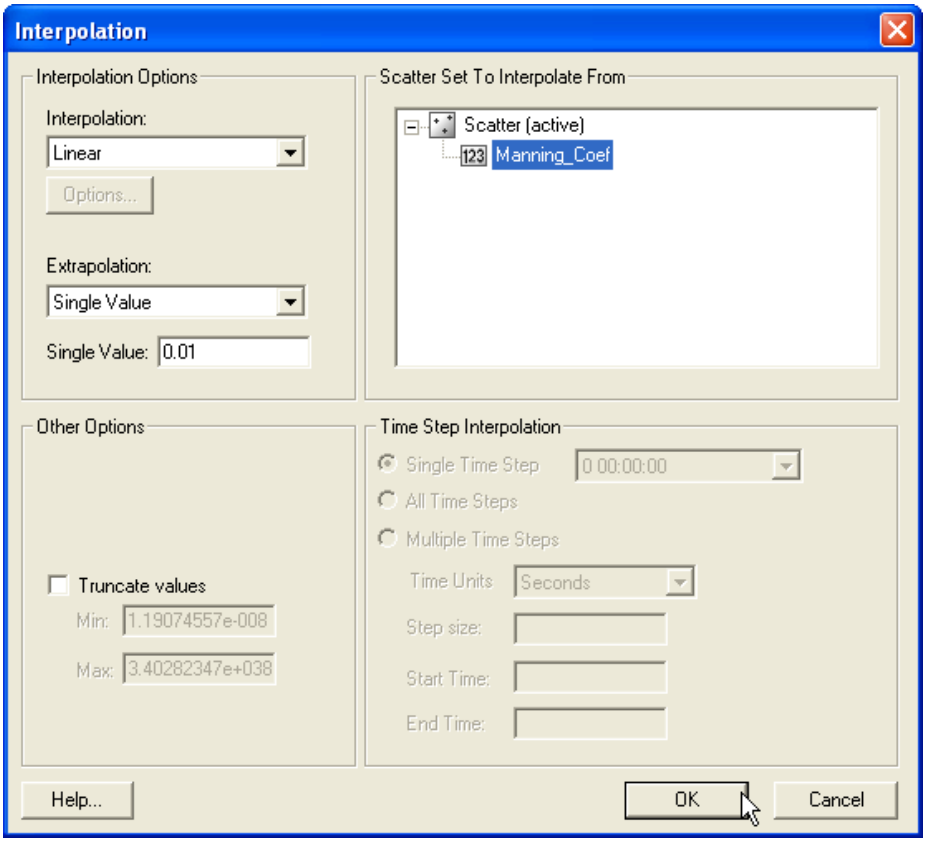


Figure 2-19. SMS Interpolation window.

Figure 2-20 shows an example of how the final bottom friction dataset looks like once a portion of the dataset has been interpolated from a scatter set.

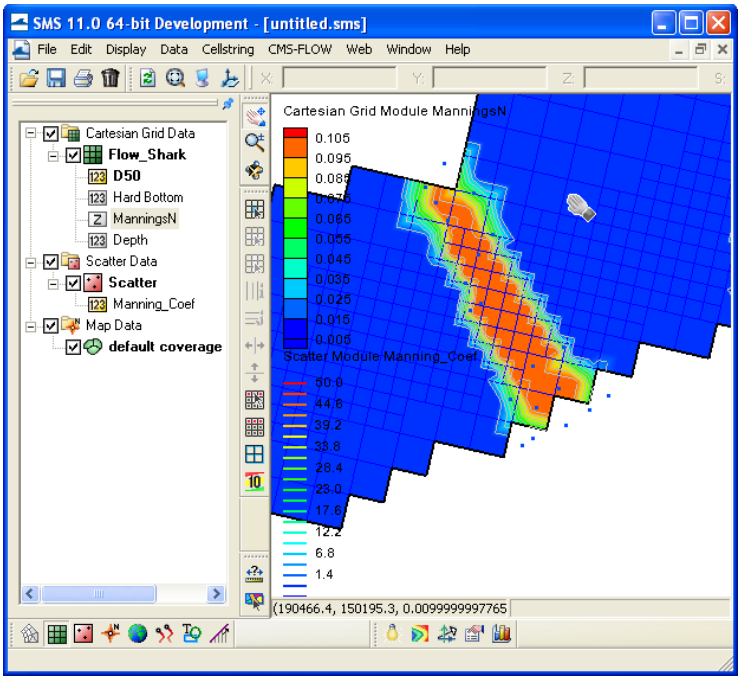


Figure 2-20. SMS *Interpolation* window.

4. Re-map the *Depth* dataset as the Elevation dataset by clicking on Data | Map Elevation and selecting the Depth dataset (Figure 2-21).

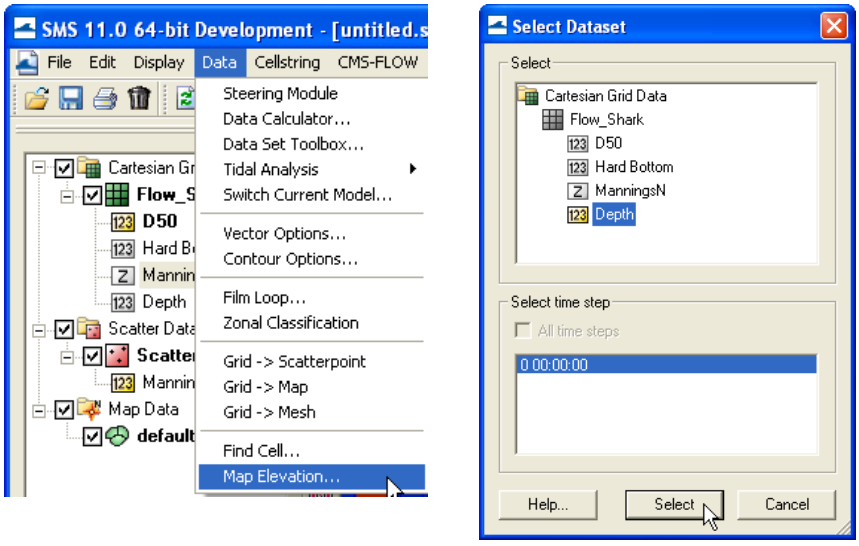


Figure 2-21. Mapping of *Depth* dataset as the *Elevation* dataset.

Exporting a Dataset

Exporting a spatially variable dataset such as bottom roughness (friction) datasets is useful for creating different project alternatives, scripting multiple runs, or for switching from different datasets for use as model input. Figure 2-22 illustrates how to export a dataset into an XMDF file. Multiple datasets can be exported into the same file. To export a dataset:

1. Select the CMS-Flow dataset(s) in the *SMS Project Explorer*.
2. Right-click and select the *Export* option. A window called *Export Data Set* will open
3. Under File Type select XMDF File.
4. *Under Time Steps*, select *All time Steps*, even if the dataset only has one time step.
5. Under *Filename* click on the folder icon and enter the location and filename for the dataset(s).

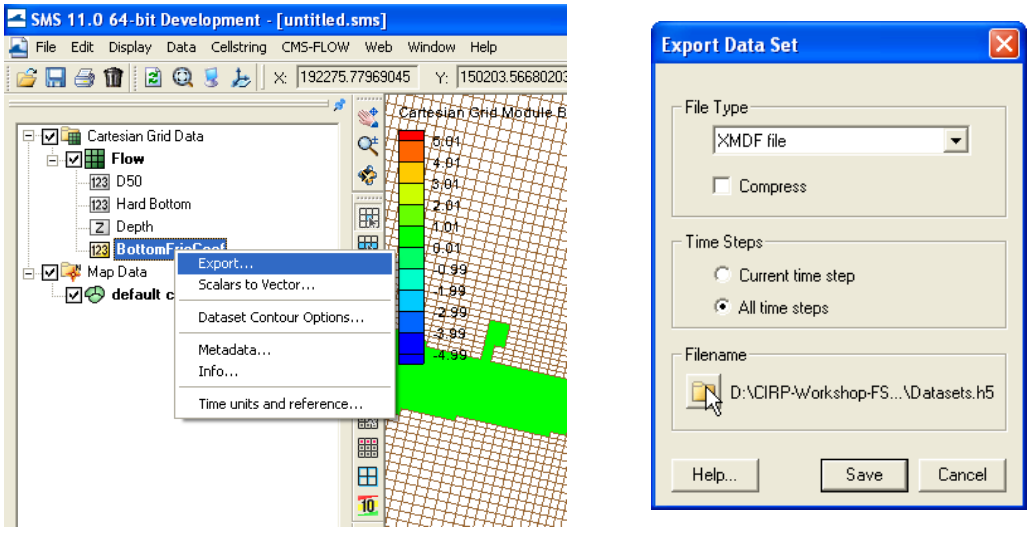


Figure 2-22. Manning's *n* contours after modifying the areas under all three bridges.

CMS Input Format Version Number

An important aspect for all numerical models is version control. In CMS, the version of the input files is specifies in the first card in the CMS-Flow *Card File*. A description of the card is provided in the table below.

Table 2-21. Example input parameters and card formats

Input	Format	Notes
CMS card file version	[card=CMS_VERSION] [name=**version**, type=*float*]	Specifies the version number of the CMS Card file.

It is important to note that the version of the input files is separate from the CMS version number. CMS is designed to be backward compatible and as long as the input file version is equal to or lower than the CMS version, then the CMS will run properly. The version of the CMS executable is written to the CMS-Flow Diagnostic File (CMS_DIAG.txt).

Example 2-5. CMS Input format version number.

```
CMS_VERSION             4.0        !Version of input card file
```

Timing

The starting time, time step, simulation duration, and ramp time are required input parameters for CMS. A description of the cards used to specify the timing parameters is described in the table below. The starting time is input as a Julian date (year and day) and time (hour).

The appropriate time step size is a function of the time stepping scheme (i.e. implicit or explicit), grid resolution, water depths, current velocities, and temporal variation of the solution. The explicit scheme time step is limited by the Courant-Friedrichs-Levy (CFL) condition

$$C = \frac{\Delta t\left(|U| + \sqrt{gh}\right)}{\min(\Delta x, \Delta y)} \le C_{\max} \tag{2-1}$$

where $C_{\max}$ is the maximum Courant number typically set to 0.4-0.5.

Since the implicit time stepping scheme is not limited by the CFL condition for stability, a different criteria is needed. For coastal applications where tides are important, it is generally recommended to use at least 35-40 time steps for tidal wave period. For a semidiurnal tide with a period of 12 hrs, then the maximum time step is approximately 18 to 20 min. However if smaller tidal wave periods of 4 to 6 hrs are considered, then a maximum time step of approximately 9-10 minutes is reasonable. When using the implicit scheme, the time step is changed during the simulation if the model detects convergence problems. The time step is reduced by half until the convergence criteria have been met. Once the model detects that the time step can be increased, the time step is doubled. The initial time step is used as the maximum time step.

The CMS does not interpolate global or point solutions in time, and therefore the initial time step should be chosen so that it is a multiple of the output intervals, or vice-versa, the output intervals should be chosen to be multiples of the initial step.

The simulation duration or length is specified in hours. The ramp period is the length of time during which the model forcing is gently increased or ramped from zero in order to avoid shocking the model at the beginning of the simulation.

Table 2-22. CMS-Flow Cards associated with timing.

Input	Format	Notes
Simulation Starting Julian Day	[card=STARTING_JDATE] [name=**StartJulDay**, type=*float*, default=none]	Deprecated. Will be replaced in future releases with STARTING_DATE_TIME
Simulation Starting Hour	[card=STARTING_JDATE_HOUR] [name=**StartJulDay**, type=*float*, default=none]	Deprecated. Will be replaced in future releases with STARTING_DATE_TIME
Simulation Starting Date and Time	[card=STARTING_DATE_TIME] [name=**StartDate0**, type=*char*, default=none, example="2004-12-24"] [name=**StartTime0**, type=*char*, default=none, example="23:34:12"] [name=**StartTimeZone**, type=*char*, default=none, example='UTC' , otpional=*true*,]	Specifies the simulation starting calendar date and time. Double quotes are optional
Simulation Run Duration	[cards=(DURATION_RUN,SIMULATION_DURATION), CMS_versions=(4.0,4.1))]] [name=**SimDur**, type=*float*, default=none, limits=(**SimDur**>0.0)] [name=**SimDurUnits**, type=*char*, options=**TimeUnits**, default='hrs']	Sets the duration of the simulation in hours.
Ramp Duration	[cards=(DURATION_RAMP,RAMP_DURATION), CMS_versions=(4.0,4.1))] [name=**RampDur**, type=*float*, default=none, limits=(**RampDur**>0.0)] [name=**RampDurUnits**, type=*char*, options=**TimeUnits**, default='hrs']	Specifies the length of the ramp period. For details on the ramp period see the Ramp Period section.

Example 2-6. General timing specification in CMS Versions 4.0 or earlier.

```
!General timing cards
STARTING_JDATE           80001
STARTING_JDATE_HOUR      0
DURATION_RUN             720.0 !hours
DURATION_RAMP            1.0   !days
```

Example 2-7. General timing specification in CMS Versions 4.1 and newer.

```
!General timing cards
STARTING_DATE_TIME       2004-12-24 23:34:12 UTC
SIMULATION_DURATION      720.0 hrs !Units optional
RAMP_DURATION            24.0 hrs
```

Tips:

1. The best way to determine the appropriate time step for an application is to conduct a sensitivity analysis. However, for most coastal applications an initial time step of 5-10 min is reasonable for the implicit scheme and a time step of 0.5-1.0 sec for the explicit scheme.
2. Use time steps that are multipliers of the output intervals. For example, it is common to output the hydrodynamics every 30 to 60 minutes, and so using a time step that is divisible by 30.

Notes:

1. The times in CMS-Flow are always referenced to Universal Time Coordinated (UTC). Prior to 1972, UTC was also called Greenwich Mean Time (GMT).

Hydrodynamics

The hydrodynamic parameters are divided into different groups: Water Properties, Bottom Friction, Wall Friction, Eddy Viscosity, Wave Mass Flux, and Wind and Atmospheric Pressure. Each group is discussed in detail in the following sections. A brief over is provided on what the options and parameters are, where to change them and other pertinent information.

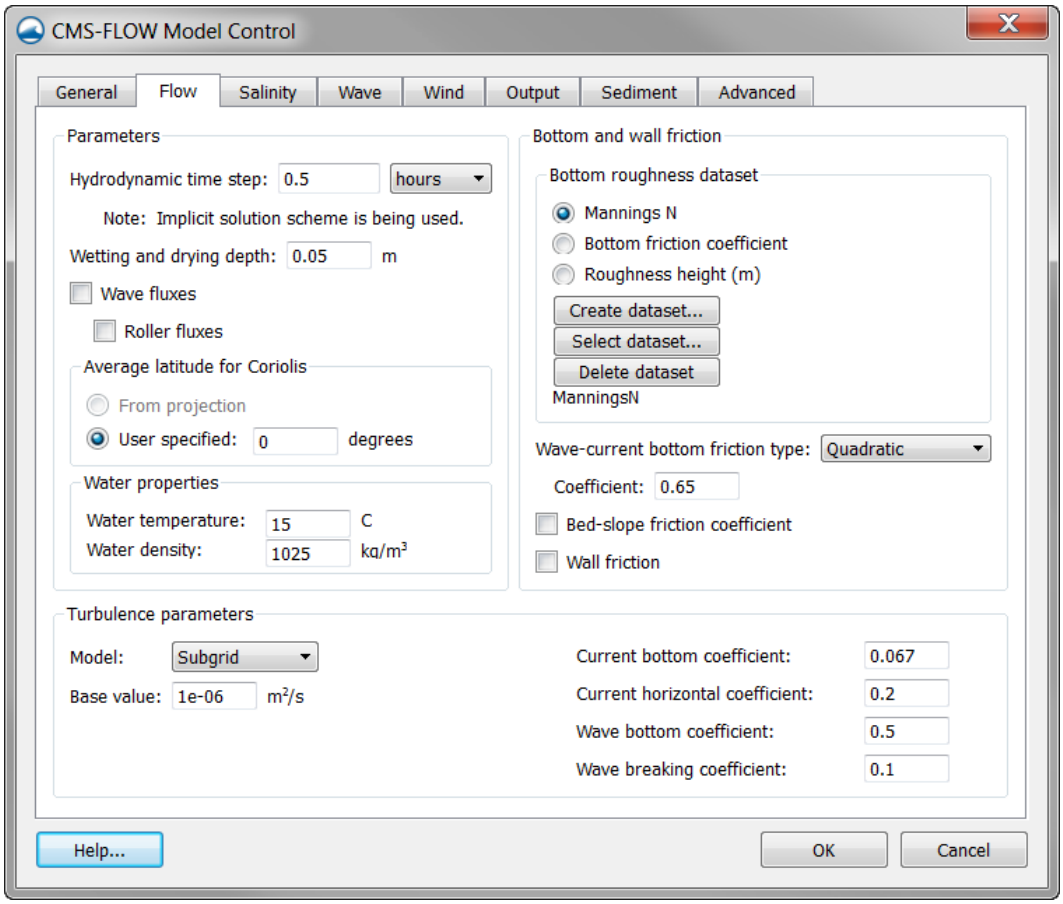


Figure 2-23. *Flow* tab of th *CMS-Flow Model Control* window in SMS 11.1.

Water Properties

The properties of sea water are a function of temperature, salinity (i.e. dissolved minerals) and pressure. The water density and dynamic viscosity (kinematic viscosity times the water density) are shown in the following two figures as a function of water temperature and salinity.

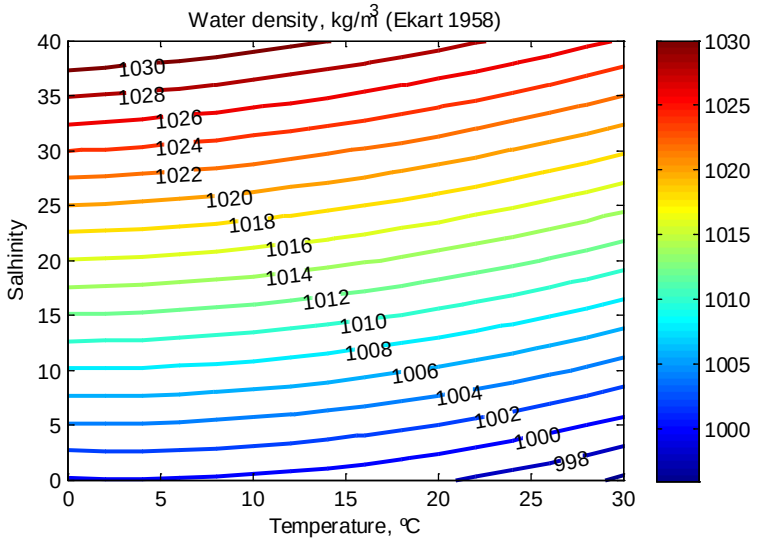


Figure 2-24. Water density as a function of temperature and salinity (at atmospheric pressure).

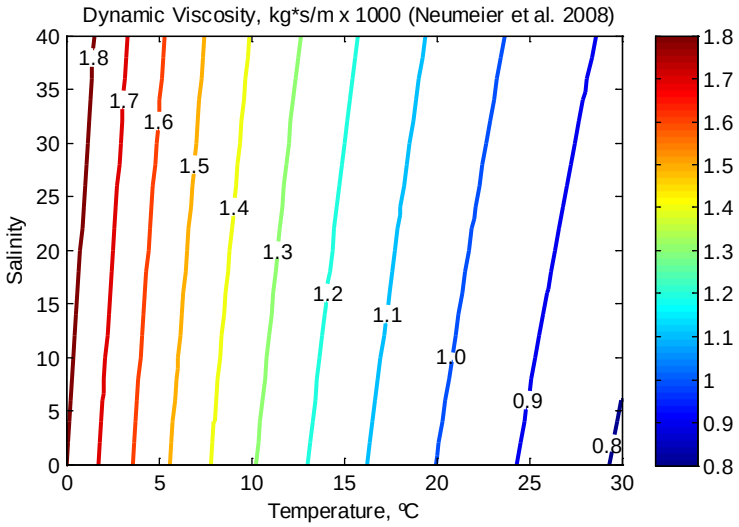


Figure 2-25. Water dynamic viscosity as a function of temperature and salinity (at atmospheric pressure).

CMS uses the water density and viscosity in various hydrodynamic and sediment transport calculations. The water density and viscosity are assumed to be constant and can be either user-specified or calculated internally based on the water temperature and salinity. The water temperature must always be specified, while either the water density or salinity need to be specified. If the salinity is not specified, then it is calculated from the water density. The salinity value used to calculate the water density and viscosity are independent of the salinity transport (initial condition and time-varying solution). Below is a table describing the CMS cards related to the specification of the water properties.

Table 2-23. CMS-Flow cards for specifying the water properties.

Input	Format	Notes
Water Temperature	[card=WATER_TEMPERATURE] [arg=**WaterTemp**, type=*float*, default=15.0] [units=**WaterTempUnits**, type=*char*, options=**TempUnits**, default='C']	Used in the calculation of the water kinematic viscosity and water density if not specified.
Water Salinity	[card=WATER_SALINITY, CMS_versions=(>4.10.21), SMS_versions=none] [arg=**WaterSalt**, type=*float*, default=35.0, optional=*true*] [units=**WaterSaltUnits**, type=*char*, options=**SaltUnits**, default='ppt']	Used in the calculation of the water kinematic viscosity and water density if not specified. Independent of salinity transport.
Water Density	[card=WATER_DENSITY, CMS_versions=(>3.50.00), SMS_versions=(>9.1)] [arg=**WaterDens**, type=*float*, default=1025.0] [units=**WaterDensUnits**, type=char, options=**DensUnits**), default='kg/m^3']	Assumed to be constant for the simulation. If not specified, calculated based on the water temperature and salinity.
Water Kinematic Viscosity	[card=WATER_KINEMATIC_VISCOSITY, CMS_versions=(>4.10.21), SMS_versions=none] [arg=**WaterVisc**, type=*float*, default=1.81e-6] [units=**WaterViscUnits**, type=char, options=**DensUnits**), default='m^2/s']	If not specified, calculated based on water temperature and salinity.

In SMS 11.1 and earlier versions, only the option to specify the water density and temperature were available (see Figure 2-23). Below is an example of the CMS cards used to specify the water density and temperature are

specified as when using the SMS interface. The default salinity value of 35 ppt is used to estimate the water viscosity. Therefore, if the user wishes to simulate a fresh water location, the salinity should also be modified.

Example 2-8. Water density and temperature specified (default method).

```
!Water properties
WATER_DENSITY     1025.0 'kg/m^3' !Units optional, default are kg/m^3
WATER_TEMPERATURE  15.0 'ºC' !Units optional default units are C
```

If the user wants the model to calculate the water density based on the water temperature and salinity, or to change the default salinity value used in the water density and viscosity calculations, then the user must manually edit the *CMS-Flow Control File*. Below is an example where the Card file is manually modified (advanced) and the water temperature and salinity specified.

Example 2-9. Water salinity and temperature specified (Advanced).

```
!Water properties
WATER_SALINITY     35.0 'ppt' !Advanced. Units optional, default are ppt
WATER_TEMPERATURE  15.0 'ºC' !Units optional default units are C
```

Notes:

1. CMS uses a constant density, and kinematic viscosity. The water density and kinematic viscosity can be either specified are calculated based on the water temperature and salinity. The salinity value used in the water properties section is independent of the salinity transport initial condition and solution.
2. The kinematic viscosity and water density y are used in the many of the sediment transport calculations.
3. The water properties are NOT calibration parameters and should be estimated as best as possible.
4. All water properties are written in the CMS_Diag.txt file for the user's reference.

Coriolis

The Coriolis force is a fictitious force which arises from the earth's rotation (Coriolis 1832). The Coriolis parameter is given by $f_c = 2\Omega \sin\phi$. Here Ω = 7.29x10-5 rad/s is the earth's angular velocity of rotation and ϕ is the latitude in degrees. Since the CMS uses a Cartesian coordinate system, the Coriolis parameter is approximated using either a constant (*f*-plane) or linearly varying (beta-plane). In the case of the *f*-plane approximation the constant value is based on the average-latitude of the domain ϕ_a as $f_c = 2\Omega \sin\phi_a$. In the case of the beta-plane approximation a linearly varying Coriolis parameter is used $f_c = f_a + \beta_a (y - y_a)$ where the subscript *a* indicates the average-latitude of the active computational cells, y is the linear northward distance, and β_a is a linear coefficient given by $\beta_a = 2\Omega \cos\phi_a / R_E$, in which R_E is the earth radius. The average latitude can either be automatically computed when a non-local horizontal projection has been set or it may also be specified in the interface. Below is an example of the Coriolis parameter variation (reference latitude is 35°).

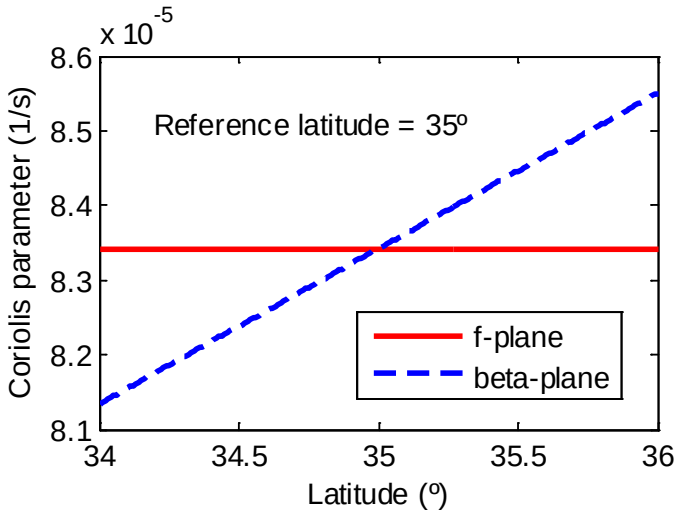


Figure 2-26. Example of the Coriolis parameter as a function of latitude using the f-plane and beta-plane approximations.

The table below describes the card used to specify the Coriolis approximation.

Table 2-24. CMS-Flow Card used to specify the Coriolis parameter approximation.

Input	Format	Notes
Coriolis Approximation	[cards=(CORIOLIS_APPROXIMATION, CORIOLIS_PLANE), version=(>4.00.00)] [arg=**CoriolisPlane**, type=*float*, options=(F,BETA), default=F]	Specifies the Coriolis approximation/plane.
Average Latitude	[cards=(AVERAGE_LATITUDE), version=(>4.00.00)] [arg=**LatAvg**, type=*float*, default=auto] [units=**LatAvgUnits**, type=char, options=**AngleUnits**), default='deg']	Specifies the average latitude used for calculating the Coriolis parameter.

Below are two examples of how to specify the Coriolis parameter options:

Example 2-10. Specifying the average latitude used to compute Coriolis.

```
!Note: Since the Coriolis plane is not specified, the default f-plane is
used
AVERAGE_LATITUDE    35.0 'degrees'
```

Example 2-11. Specifying the Coriolis approximation.

```
!Note: The average latitude is not specified and is calculated internally
CORIOLIS_PLANE    BETA   !{F} | BETA
```

Notes:

1. Independently of whether the f-plane or beta-plane approximation is used to calculate the Coriolis parameter, the average latitude is needed. The average latitude can be either estimated based on the grid projection or specified by the user.
2. The Coriolis parameter can be turned off (set to zero) by setting the “Average latitude for Coriolis” to “User specified” and entering 0.0 (see Figure 2-23).
3. The Coriolis parameter is NOT a calibration parameter. It is recommended to use the Coriolis parameter for all regional scale applications. For small project scale applications, the effects of the Coriolis parameter are less important.

Bottom Friction

The bed roughness may be may specified using one of the following parameters:

1. Manning's *n* coefficient (*n*) [s/m$^{1/3}$]
2. Bottom friction coefficient (c_b) [-]
3. Roughness height (Nikuradse roughness) (k_s) [m]

The Manning's *n* coefficient (also referred to as Gauckler-Manning coefficient) was originally first developed for open channel flow (Gauckler 1867). If the Manning's coefficient is specified, the bed friction coefficient is calculated internally at each time step from (Soulsby 1997)

$$c_b = gn^2 h^{-1/3} \tag{2-2}$$

If the input roughness is specified as a roughness height, the bed friction coefficient is calculated at each time step as (Graf and Altinakar 1998)

$$c_b = \left[\frac{\kappa}{\ln(z_0 / h) + 1} \right]^2 \tag{2-3}$$

where κ =0.4 is Von Karman constant, and z_0 is the bed roughness length which is related to the Nikuradse roughness (k_s) by $z_0 = k_s / 30$ (hydraulically rough flow).

Table 2-25 shows mean roughness heights from a large number of field measurements over natural sea beds.

Table 2-25. Mean values of Nikradse roughness heights for different bottom types (modified from Soulsby (1983).

Bottom Type	Nikuradse Roughness Height k_s [m]
Mud	0.006
Mud/Sand	0.021
Silt/Sand	0.0015
Unrippled Sand	0.012
Rippled Sand	0.18
Sand/Shell	0.009
Sand/Gravel	0.009
Mud/Sand/Gravel	0.009
Gravel	0.09

Table 2-26. Base Manning's n coefficients as a function of the median grain size for the upper flow regime (Benson and Dalrymple 1984).

Median Grain Size [mm]	Manning's Coefficient [$s/m^{1/3}$]
0.2	0.012
0.3	0.017
0.4	0.02
0.5	0.022
0.6	0.023
0.8	0.025
1.0	0.026

Notes:

1. The input roughness parameter is held constant throughout the simulation and not updated at each time step as a function of the hydrodynamics, bed composition, and bedforms. This is a common engineering approach which can be justified by the lack of data to initialize the bed composition and the large error in estimating the bed composition evolution and bedforms. In addition using a constant bottom roughness simplifies the model calibration. In future versions of CMS, the option to automatically estimate the bed roughness from the bed composition and bedforms will be added.
2. The bed roughness used for hydrodynamics may not be the same as that which is used for the sediment transport calculations because each sediment transport formula was developed and calibrated using specific methods for estimating bed shear stresses or velocities, and these cannot be easily changed. In the case of the Lund-CIRP transport formula the bed roughness is calculated as the sum of grain, transport, and bedform roughness's.
3. The Manning's n coefficient should not be confused with the Manning's number M used by other models. The two are related by $n = M^{-1}$.

Bottom Friction Dataset

The bottom roughness may be specified using a grid dataset referred to as the *Bottom Friction Dataset*. The dataset contain spatially variable or constant values and is the most general way of specifying the bed roughness. A description of the CMS-Flow cards related to the *Bottom Friction Dataset* are shown in the table below.

Table 2-27. SMS Cards related to the bottom friction datasets

Input	Format	Notes
Manning's *n* Dataset	[card=MANNINGS_N_DATASET] [name=**FricFile**, type=*char*, default=**GridFile**] [name=**FricPath**, type=*char*, default=**"CaseName/Datasets/ManningsN"**]	Grid file name and dataset path for the input Manning's N Dataset
Bottom Friction Coefficient Dataset	[card=BOTTOM_FRICTION_COEF_DATASET] [name=**FricFile**, type=*char*, default=**GridFile**] [name=**FricPath**, type=*char*, default=**"CaseName/Datasets/BottomFricCoef"**]	Grid file name and dataset path for the input bottom friction coefficient dataset
Roughness Height Dataset	[card=ROUGHNESS_HEIGHT_DATASET] [name=**FricFile**, type=*char*, default=**GridFile**] [name=**FricPath**, type=*char*, default=**"CaseName/Datasets/RoughnessHeight"**]	Grid file name and dataset path for the input roughness height dataset

Example 2-12. Specifying the bottom roughness using a Manning's coefficient dataset.

```
MANNINGS_N_DATASET   "Flow_grid.h5" "Flow/Datasets/ManningsN"
```

In SMS versions 10.1 and earlier, the bottom roughness was specified using the Manning's *n* coefficient in the *Cells* tab of the *CMS-Flow Model Control* window. In SMS 11.0 the bottom roughness is specified in the *Bottom Friction Dataset* section of within the *Input* tab of the *CMS-Flow Model Control* window (see Figure 2-27). The *Bottom Friction Dataset* is specified at every computational (ocean) cell and is required for each model simulation. In SMS 11.1, the bottom friction dataset is specified in the *Flow* tab of the *CMS-Flow Model Control* window (see Figure 2-23). As in SMS 11.0, the bottom roughness can be specified as a roughness height, bottom friction coefficient, or in Manning's *n* coefficient.

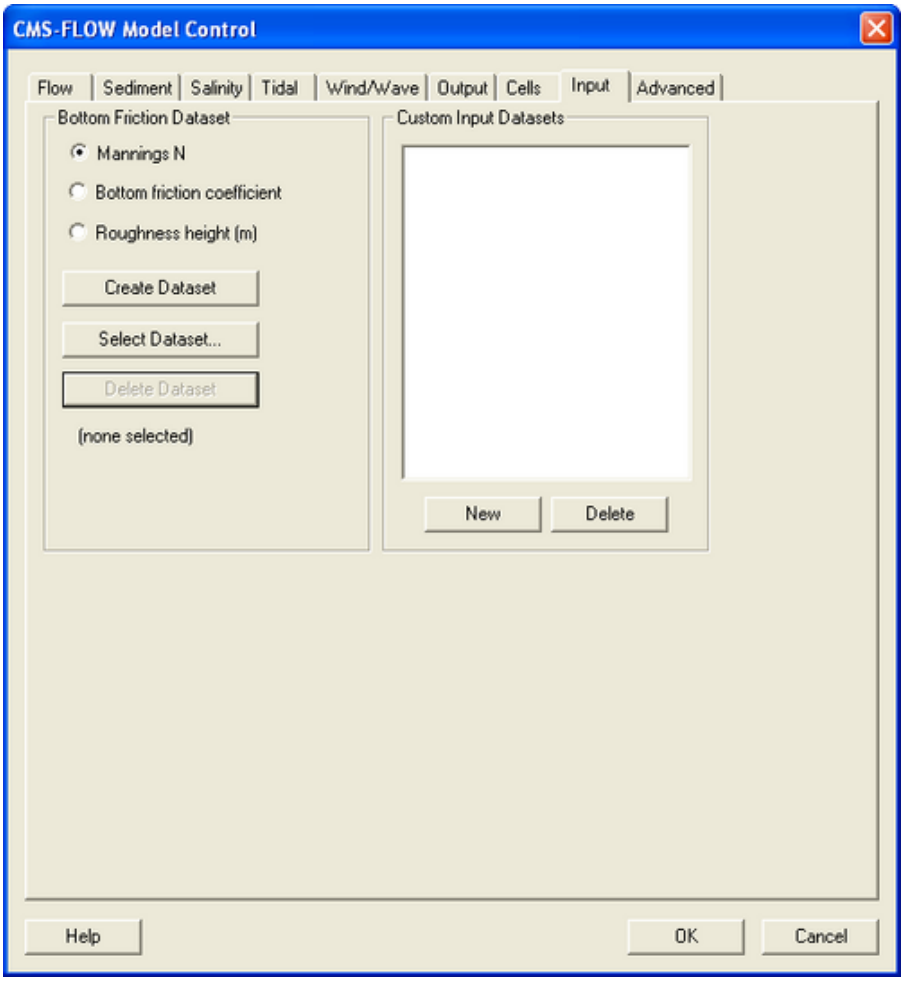


Figure 2-27. *Bottom Friction Dataset* specification via the *Input* tab of the *CMS-Flow Model Control* window in SMS 11.0

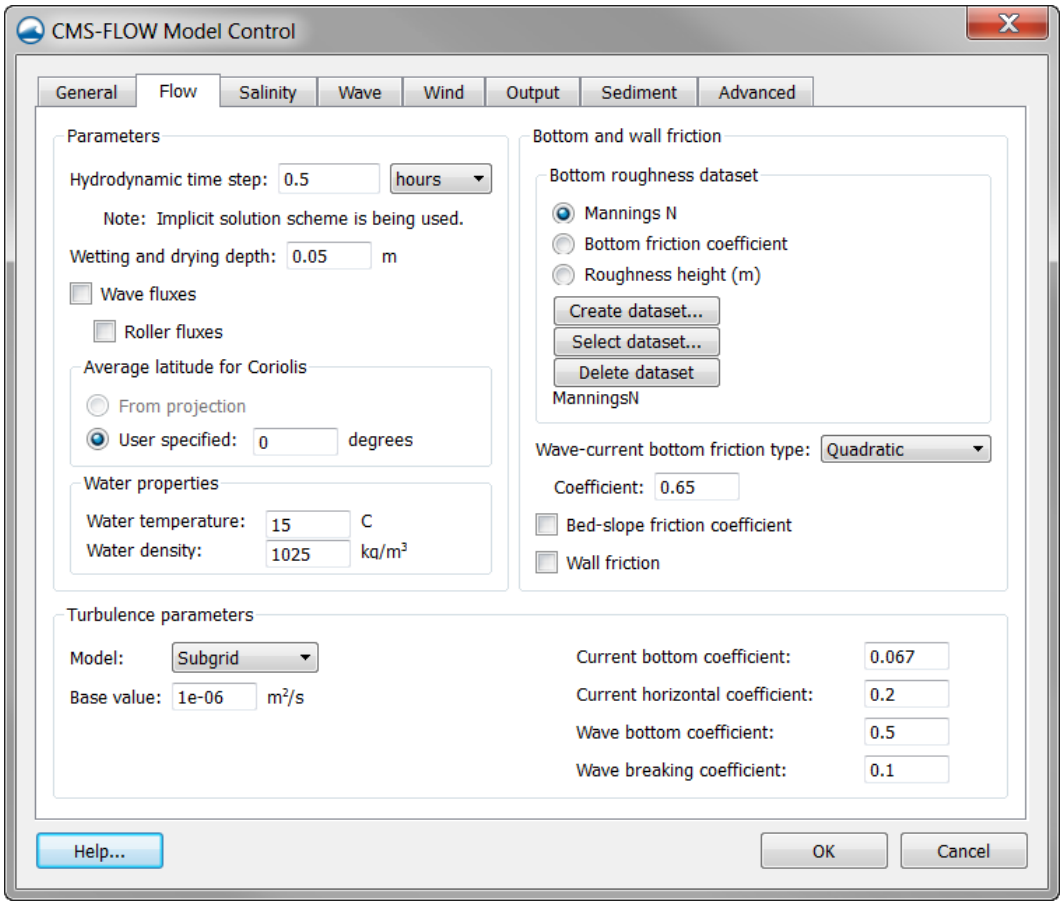


Figure 2-28. Bottom friction specification via the *Flow* tab of th *CMS-Flow Model Control* window in SMS 11.1.

Creating a Bottom Friction Dataset

Each time a new CMS-Flow simulation is created a bottom friction (roughness) dataset needs to be created using the *Create Dataset* button. The bottom friction parameter (related to Manning *n*) is spatially varying (cell-specific) over the grid domain. The default value upon grid creation is 0.025. At times a user may desire to represent locations where added friction is needed due to structures, vegetation, or increased turbulence due to rapid changes in current speed. Once the *Bottom Friction Dataset* is created and the *CMS-Flow Model Control* window closed, a new auxiliary dataset will appear in the *SMS Project Explorer* (see Figure 2-29).

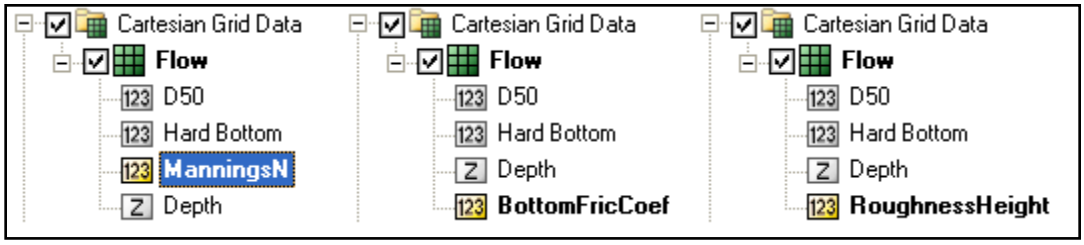


Figure 2-29. Three SMS Project Explorer showing the Manning's *n*, Bottom Friction Coefficient and Roughness Height datasets.

Bottom Friction Value

Additional advanced cards (in Table 2-28) are available for setting the bottom friction to a constant for the whole grid. These cards are useful for running sensitivity analysis for a wide range of values in cases which can be approximated with a single constant bottom friction value.

Table 2-28. Advanced Cards to set bottom friction dataset to a constant value

Input	Format	Notes
Manning's *n* Coefficient Dataset	[card=MANNINGS_N_CONSTANT] [arg=**cbotfric**, type=*float*, default=**0.025**] [units=**cbotfricunits**, default='s/m^(1/3)']	Specifies a constant input *Manning's n* coefficient. Over-rides any previous bottom friction cards.
Bottom Friction Coefficient Dataset	[card=BOTTOM_FRICTION_COEF_CONSTANT] [arg=**cbotfric**, type=*float*, units=*none*, default=**0.006**] [units=**cbotfricunits**, default='']	Specifies a constant input *Bottom Friction Coefficient*. Over-rides any previous bottom friction cards.
Roughness Height Dataset	[card=ROUGHNESS_HEIGHT_CONSTANT] [arg=**cbotfric**, type=*float*, default=**0.01**] [units=**cbotfricunits**, default='m']	Specifies a constant input *Roughness Height* in m. Over-rides any previous bottom friction cards.

Example 2-13. Specifying a constant Manning's coefficient.

```
MANNINGS_N_CONSTANT 0.025 's/m^(1/3)' !Units are optional
```

Wave-enhanced Bottom Friction

In the presence of waves, the mean (short-wave averaged) bottom shear stress which is used in the momentum equations is enhanced:

$$\tau_{bi} = \lambda_{wc}\tau_{ci} \tag{2-4}$$

where:

τ_{bi} = combined wave and current mean bed shear stress [Pa]

λ_{wc} = nonlinear bottom friction enhancement factor ($\lambda_{wc} \geq 1$) [-]

τ_{ci} = current-related bed shear stress [Pa].

The nonlinear bottom friction enhancement factor (λ_{wc}) is calculated using one of the following formulations (name abbreviations are given in parenthesis):

1. Wu et al. (2010)quadratic formula (QUAD)
2. Soulsby (1995) empirical two-coefficient data fit (DATA2)
3. Soulsby (1995) empirical thirteen-coefficient data fit (DATA13)
4. Fredsoe (1984) analytical wave-current boundary layer model (F84)
5. Huynh-Thanh and Temperville (1991) numerical wave-current boundary layer model (HT91)
6. Davies et al. (1988) numerical wave-current boundary layer model (DSK88)
7. Grant and Madsen (1979) analytical wave-current boundary layer model (GM79)

For further details on the above methods, the reader is referred to the above references. The default is a simple quadratic formula but can be changes using the *Advanced Card* described in Table 2-29.

Table 2-29. Advanced Card used for specifying the wave-current mean stress formulation

Input	Format	Notes
Wave Current Mean Stress	[card=(WAVE_CURRENT_MEAN_STRESS, WAVE-CURRENT_MEAN_STRESS)] [arg=**awavcur**, type=*char*, options=(QUAD,DATA2,DATA13,F85, HT95,DSK88,GM79),default=QUAD]	Defines the model used for calculating the mean bottom shear stress used in hydro
Wave Bottom Friction Coefficient	[card=WAVE_BOTTOM_FRICTION_COEFFICIENT] [arg=**cfricwav**, type=*float*, default=0.5, range=(0.3,0.7)]	Wave bottom friction coefficient used for quadratic combined wave-current mean bed shear stress calculation.

Example 2-14. Specifying the model for combined wave-current bottom friction.

```
WAVE_CURRENT_MEAN_STRESS            QUAD !DATA2, DATA13, F84, etc.
WAVE_BOTTOM_FRICTION_COEFFICIENT    0.5 !Dimensionless, only for QUAD
```

Bed-slope Friction Factor

The bed slope friction factor accounts for the increased surface area over sloping beds as compared to the horizontal area. This feature may be turned ON or OFF through the Advanced Card described in

Table 2-30. Advanced Card used for activating the bed slope friction factor

Input	Format	Notes
Bed Slope Friction Factor	[card=(BED_SLOPE_FRICTION_FACTOR, BED_SLOPE_FRICTION_FACTOR)] [name=**fricbedslope**, type=*char*, options=(ON,OFF), default=OFF]	Specifies whether to include the bed slope friction factor or not in the calculation of the bed friction

Example 2-15. Specifying the bed-slope friction factor.

```
BED_SLOPE_FRICTION_FACTOR           ON !ON | {OFF}
```

Wall Friction

Two boundary conditions are available for the tangential flow at closed boundaries:

1. Free-slip: No tangential shear stress (no wall friction);
2. Partial-slip: Tangential shear stress (wall friction) calculated based on the log law.

Wall friction enhances the flow drag perpendicular to any closed boundary. Assuming a log law for a rough wall, the partial-slip tangential shear stress is given by

$$\tau_{wall} = \rho c_{wall} U_{\|}^2 \tag{2-5}$$

where $U_{\|}$ is the magnitude of the wall parallel current velocity, and c_{wall} is the wall friction coefficient equal to

$$c_{wall} = \left[\frac{\kappa}{\ln(y_P / y_0)} \right]^2 . \tag{2-6}$$

Here, y_0 is the roughness length of the wall and is assumed to be equal to that of the bed (i.e., $y_0 = z_0$). The distance from the wall to the cell center is y_P.

Table 2-31. SMS Cards related to the wall friction

Input	Format	Notes
Wall Friction	[card=USE_WALL_FRICTION_TERMS] [name=**wallfric**, type=*char*, options=(ON,OFF), default=OFF]	Turns on or off wall friction

Example 2-16. Specifying the wall friction.

```
USE_WALL_FRICTION_TERMS      ON !ON | {OFF}
```

The wall friction may be turned ON or OFF in the *Flow* tab of the *CMS-Flow Model Control* window (see Figure 2-30). The default in SMS is for

the wall friction to be *ON*. Table 2-31 shows the CMS card used activating the wall friction.

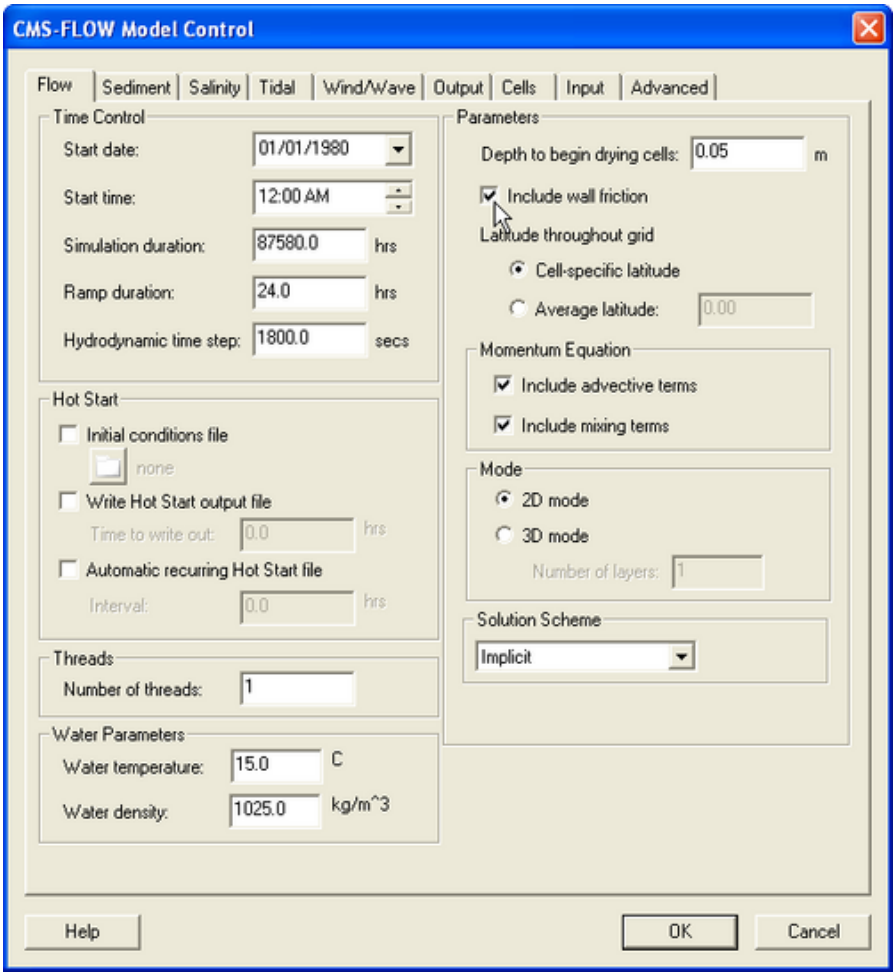


Figure 2-30. Wall friction specification in the *CMS-Flow Model Control* window of SMS 11.0.

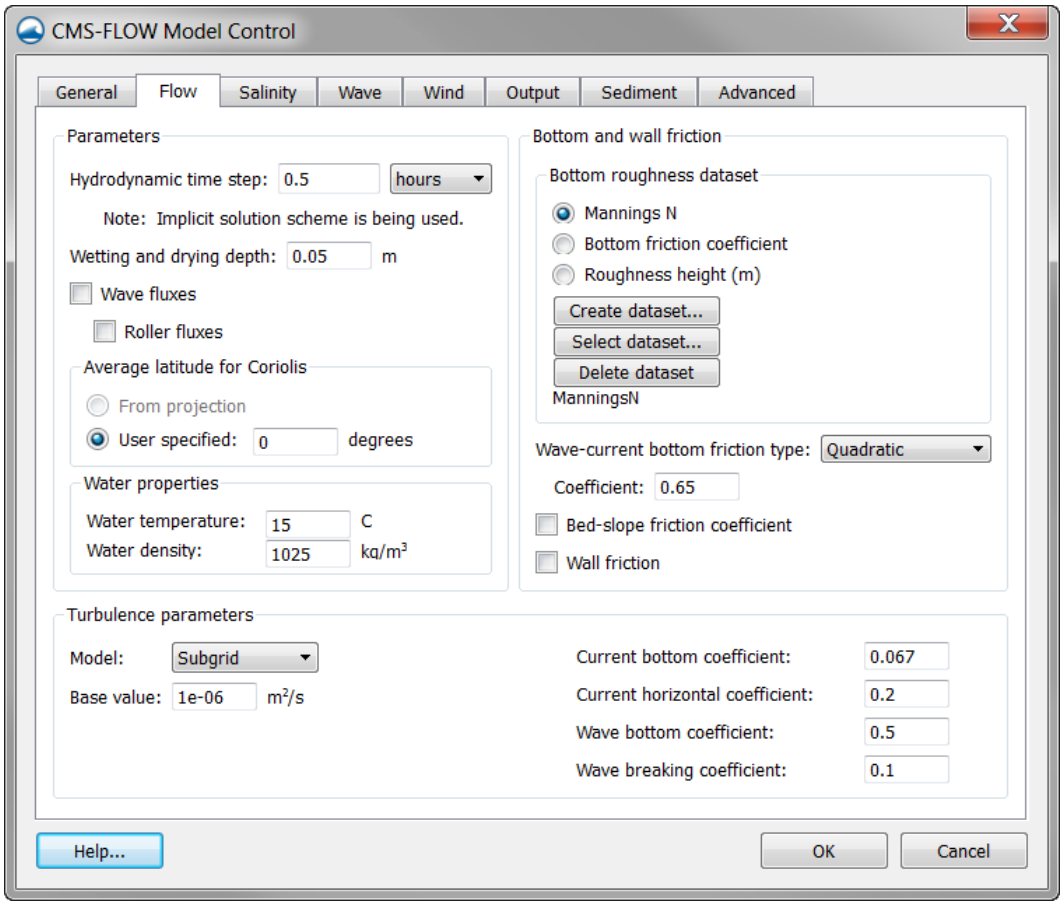


Figure 2-31. Wall friction specification in the *CMS-Flow Model Control* window of SMS 11.1.

Eddy Viscosity

The term *eddy viscosity* arises from the fact that small-scale vortices or eddies on the order of the grid cell size are not resolved, and only the large-scale flow is simulated. The eddy viscosity is intended to simulate the dissipation of energy at smaller scales than the model can simulate. In the nearshore environment, large mixing or turbulence occurs due to waves, wind, bottom shear, and strong horizontal gradients. Therefore, the eddy viscosity is an important parameter which can have a large influence on the calculated flow field and resulting sediment transport. In CMS-Flow, the total eddy viscosity (ν_t) is equal to the sum of three parts: 1) a base value (ν_0); 2) the current-related eddy viscosity (ν_c); and 3) the wave-related eddy viscosity (ν_w) defined as follows:

$$\nu_t = \nu_0 + \nu_c + \nu_w . \tag{2-7}$$

The base value (ν_0) is approximately equal to the kinematic viscosity (~1.81×10^{-6} m^2/s) but may be changed by user. The other two components (ν_c and ν_w) are described in the sections below.

There are four algebraic models for the current-related eddy viscosity: 1) Falconer Equation, 2) depth-averaged parabolic, 3) subgrid, and 4) mixing-length. The default turbulence model is the subgrid model but may be changed by the user. The wave-related eddy viscosity consists of two components representing turbulence generated at the bed due to oscillatory flow, and at the surface due to wave breaking.

In SMS 11.0 all of the eddy viscosity options and parameters are entered via Advanced Cards and only the option is given to turn *On* or *Off* the mixing terms (see Figure 2-32).

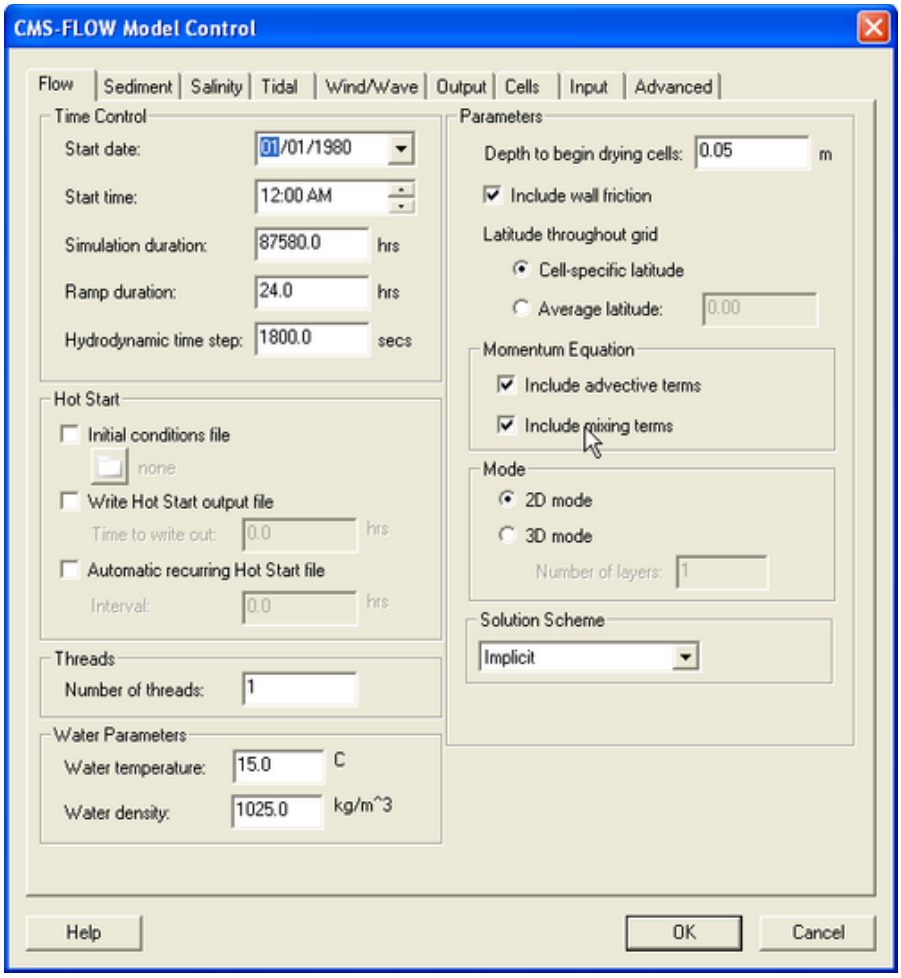


Figure 2-32. Toggle for turning *On* and *Off* the mixing terms in the SMS 11.0 interface.

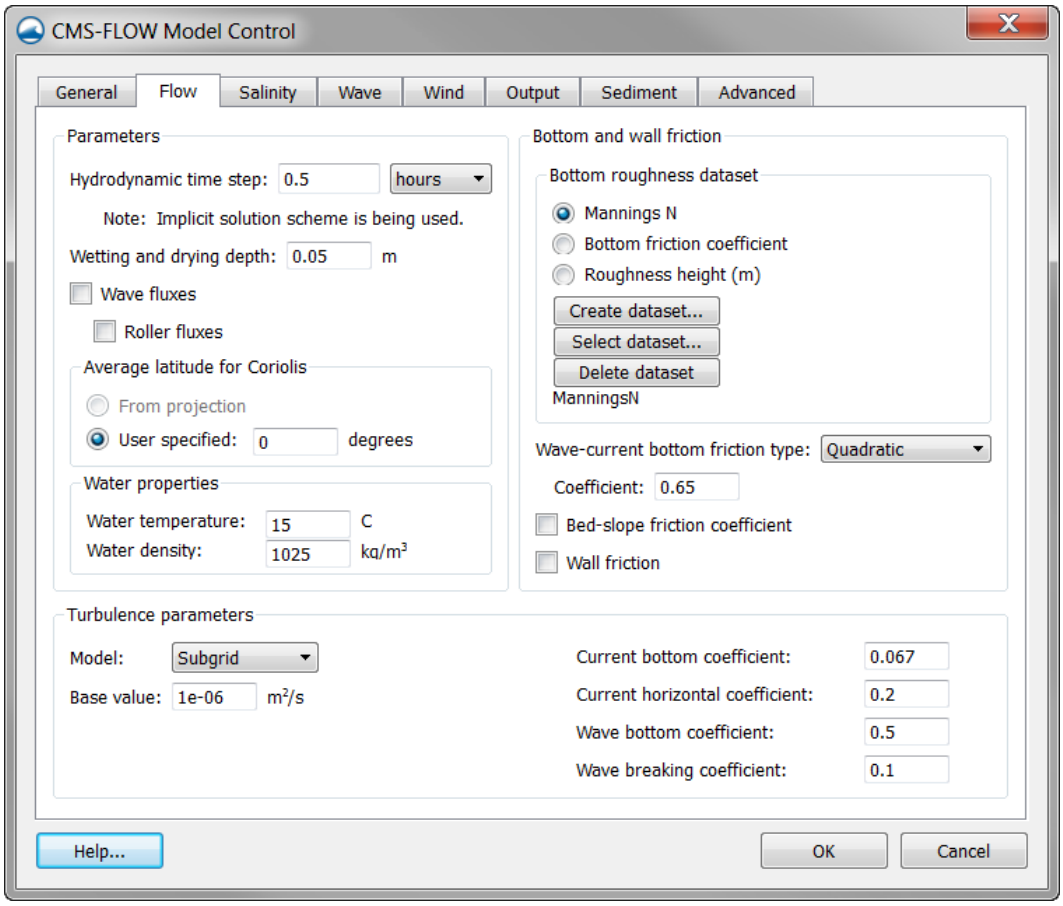


Figure 2-33. Eddy viscosity parameter specifiecation within the *Flow* tab of th *CMS-Flow Model Control* window in SMS 11.1.

A description of the CMS-Flow Cards related to eddy viscosity is shown, below, in the table below.

Table 2-32. CMS card used to set turbulence model and eddy viscosity terms

Input	Format	Notes
Mixing Terms	[card=USE_MIXING_TERMS] [name=**mturbul**, type=*char*, options=(ON,OFF), default=ON]	Deprecated. Turns On and Off the mixing terms in the momentum and transport equations.
Turbulence Model	[card=TURBULENCE_MODEL] [name=**mturbul**, type=*char*, options=(SUBGRID,FALCONER, PARABOLIC,MIXING-LENGTH, default=SUBGRID]	Specifies the turbulence model used. For details on the different turbulence models see section Eddy Viscosity.
Eddy Viscosity Constant	[card=EDDY_VISCOSITY_CONSTANT] [name=**cviscon**, type=*float*, default=**1.0E-6**,	Constant contribution or base value of eddy viscosity

	range=(**cviscon**>=0.0)]	
Eddy Viscosity Bottom	[card=EDDY_VISCOSITY_BOTTOM] [name=**cviscon**, type=*float*, default=**0.0667**, range=0.02-0.1]	Coefficient related to the contribution to eddy viscosity from the bottom shear
Eddy Viscosity Horizontal	[card=EDDY_VISCOSITY_HORIZONTAL] [name=**cvishor**, type=*float*, default=**0.2**, range= 0.1-0.2]	Coefficient related to the contribution to eddy viscosity from horizontal velocity gradients
Eddy Viscosity Wave	[card=EDDY_VISCOSITY_WAVE] [name=**cviswav**, type=*float*, default=**0.5**, range=0.2-1.0]	Coefficient related to the wave bottom friction contribution to eddy viscosity
Eddy Viscosity Breaking	[card=EDDY_VISCOSITY_BREAKING] [name=**cviswavbrk**, type=*float*, default=**0.05**, range=0.05-0.1]	Coefficient related to the wave breaking contribution to eddy viscosity

Notes:

1. The eddy viscosity can be calibrated by looking at several features of the flow. Increasing the eddy viscosity

   a. Increases the width of the longshore current.
   b. Increases the spread of the ebb and flood tidal jet currents.
   c. Decreases the offshore extent of the ebb jet current.
   d. Smoothes out small gyres and recirculation zones.

2. The eddy viscosity affects the sediment mixing since the sediment mixing coefficient is calculated as the eddy viscosity divided by the Schmidt number.
3. The eddy viscosity can be used to dampen out model (by increasing its value). However, this should be done with caution and is only recommended as a last resort. Usually, model instabilities are linked to bathymetry or boundary problems and can be resolved by other means.

Wave Mass Flux

The wave mass (or volume) flux has two contributions; one due to Stokes drift (mass transport above the trough level) and the second due to the surface roller. The Stokes drift component is due to the oscillatory wave motion and is always present. The surface roller flux is due to the breaking waves and is only present in the surf zone. The mean wave mass flux velocity, or simply the mass flux velocity for short, is defined as the mean wave volume flux divided by the local water depth and is approximated here as (Phillips 1977; Ruessink et al. 2001; Svendsen 2006)

$$U_{wi} = \frac{(E_w + 2E_{sr})w_i}{\rho hc} \tag{2-8}$$

where

E_w = wave energy $= \frac{1}{16}\rho g H_s^2$ [N/m]

H_s = significant wave height [m]

E_{sr} = surface roller energy density [N/m]

w_i = wave unit vector $= (\cos\theta, \sin\theta)$ [-]

c = wave speed [m/s]

θ = mean wave direction, [rad]

The first component is due to the wave energy (E_w is the Stokes velocity), while the second component is due to the surface roller (E_{sr} is only present in the surfzone). Therefore the total flux velocity V_i may be written as

$$V_i = U_i + U_{wi} \tag{2-9}$$

where U_i is the depth-averaged current velocity. Turning on the *Wave Mass Flux* activates the wave flux velocity. This velocity always includes the Stokes velocity. However, the surface roller component is optional and can be turned on separately, but only if the wave mass flux is on and the surface roller model is also activated. The table below describes the CMS cards used to activate the wave mass flux and surface roller mass flux.

Table 2-33. CMS card used to turn on or off the wave volume flux velocities

Input	Format	Notes
Wave Mass Flux	[card=WAVE_MASS_FLUX] [name=**WaveFlux**, type=*char*, options=(ON,OFF), default=OFF]	Turns on and off the wave volume flux velocities
Roller Mass Flux	[card=ROLLER_MASS_FLUX] [name=**RolFlux**, type=*char*, options=(ON,OFF), default=OFF]	Turns on or off the contribution to the wave volume flux velocities due to the surface roller in the surf zone. Only used if surface roller model is turned on.

Example 2-17. Wave mass flux specification.

```
WAVE_MASS_FLUX        ON   !{OFF} | ON
ROLLER_MASS_FLUX      ON   !{OFF} | ON, Only applicable if wave flux is on
```

As mentioned by Svendsen (2006), the Stokes drift component of the wave mass flux from linear wave theory is expected to be an overestimation in the surf zone since nonlinear waves are less “bulky” and therefore transport less mass. For this reason, it can be argued in favor of neglecting the surface roller component in the surf zone to compensate for the overestimation of the Stokes velocity. By default, the wave mass flux is off.

Notes:

- It is recommended to use the wave mass flux without the roller component. This usually leads to the best results.

Wind and Atmospheric Pressure Forcing

The wind and atmospheric pressure forcing are specified in the *Wind* tab of the *CMS-Flow Model Control Window* (see figure below). There are four main options for the wind and atmospheric pressure forcing

1. None
2. Spatially constant time-series
3. Meteorological Stations
4. Temporally and spatially varying

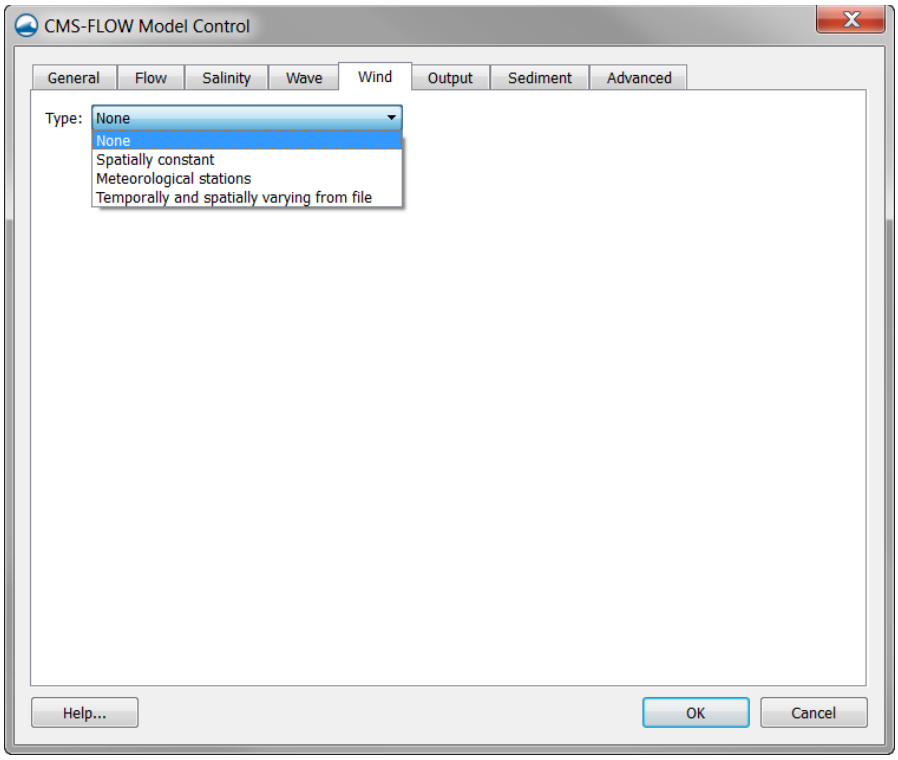


Figure 2-34. CMS-Flow Wind tab in the CMS-Flow Model Control window in SM 11.1.

Spatially Constant Wind Time series

Spatially constant wind is specified in the Wind/Wave tab of the CMS-Flow Model Control window (see Figure 2-35).

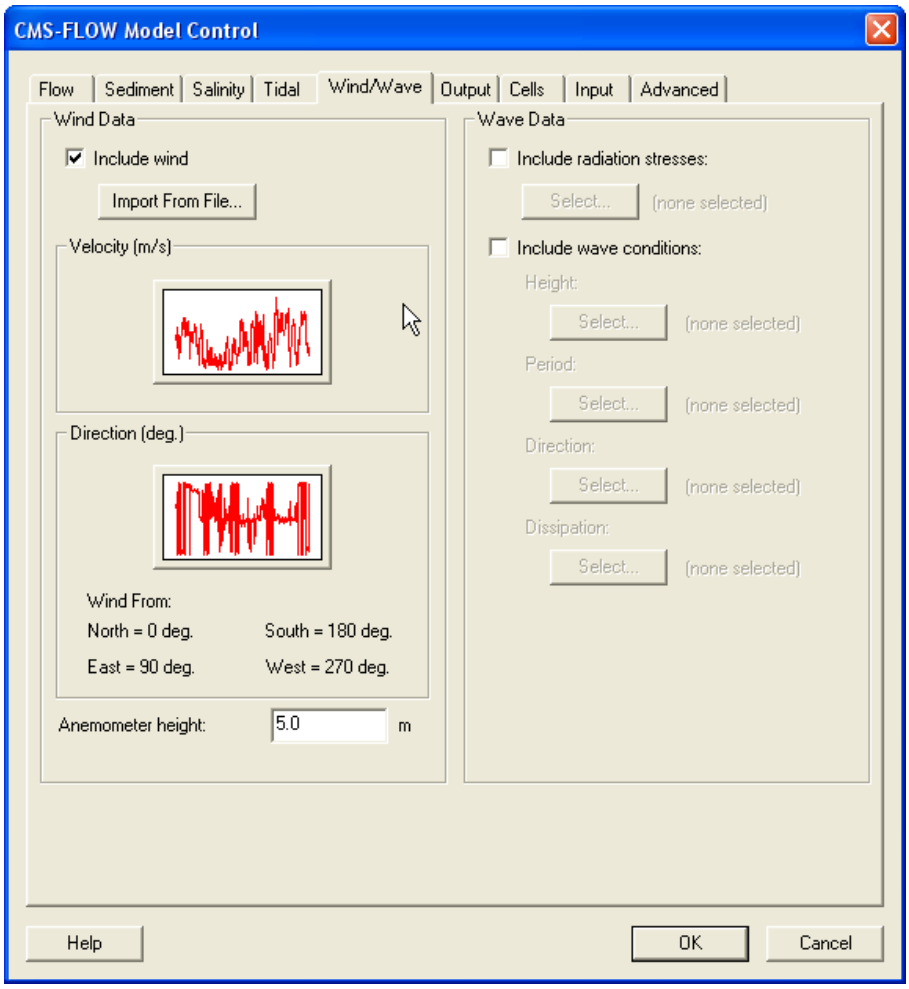


Figure 2-35. CMS-Flow Wind/Wave tab in the CMS-Flow Model Control window in SM 11.0.

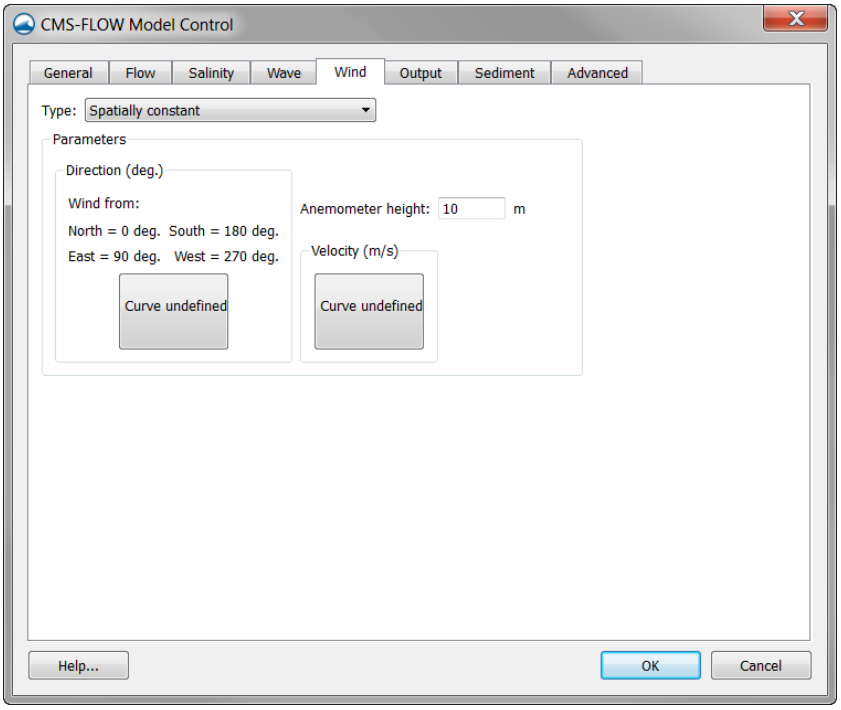


Figure 2-36. CMS-Flow Wind tab in the CMS-Flow Model Control window in SM 11.1.

Table 2-34. CMS cards used for spatially constant wind forcing.

Input	Format	Notes
Wind Time Series Curve	[card=WIND_INPUT_CURVE] [name=**WindFile**, type=*char*, default=**MPFile**] [name=**WindPath**, type=*char*, default= "PROPERTIES/Model Params/WindCurve"]	File name and path of the spatially constant wind time series.

Spatially Variable Wind and Atmospheric Pressure

CMS-Flow V4.0 (and higher) has the option to use spatially variable wind and atmospheric pressure forcing. Currently, this feature is specified in the advanced card section. There are currently two input options for spatially variable winds and atmospheric pressure: (1) Single ASCII Wind/Pressure File and (2) Multiple Oceanweather Files. These input formats and their cards are described in the following sections. When using spatially variable winds and atmospheric pressure it is good to keep in mind the following notes.

Notes:

- Interpolation File: When using spatially variable winds CMS will output a file named *Intpcoef_wndfl.bin*. This file contains the coefficients used to interpolate the winds and atmospheric pressure to the CMS-Flow grid. If the same flow and wind grids are run, the model will automatically detect the interpolation file and read in the coefficients instead of calculating them. If the either grid is changed in size, the model will automatically detect this and recalculate the coefficients. However, if the grids are changed but stay the same size, the model will not be able to detect that they have changed. Therefore, whenever any changes are made to the flow or wind grids, it is best to delete the prior interpolation file.
- Ramp Period: During the ramp period, the ramp is applied to both the wind shear stresses and spatial variations of the atmospheric pressure.

Wind and Atmospheric Pressure File

This file format is equivalent to the ADCIRC fort.22 format with NWS=6. The winds and atmospheric pressure are specified on a rectangular grid at constant temporal intervals. The wind grid is assumed to be in the same horizontal coordinate system as the CMS grid. The wind data is spatially interpolated using an inverse distance method and temporally interpolated using linear interpolation. The grid is assumed to vary from North to South and West to East so that north-west corner is the grid origin. The spatial extents of the wind/pressure grid must be consistent with the flow grid. The wind file is written from West to East starting at the North-West Corner. The file is read in free format. An example of section of a single ASCII Wind File is shown below:

Example 2-18. Excerpt of a Wind and Atmospheric Pressure File:

```
7.5835 1.2324 0.00
7.5835 1.2324 0.00
7.5835 1.2324 0.00
...
```

Example 2-19. FORTRAN code used to read the file is shown below:

```
!For each wind snap shot
do i=1,nwindi
  do j=1,nwindj
    read(wunit,*,end=444) wndspdx(i,j),wndspdy(i,j),atmpres(i,j)
  enddo !j
enddo !i
```

Example 2-20. Specifying a wind and atmospheric pressure file and parameters.

```
WIND_PRESSURE_SINGLE_FILE       "wind.dat"
WIND_PRESSURE_GRID_PARAM        201 201 50000.0 -50000.0 500.0 500.0
WIND_PRESSURE_TIME_INCREMENT    360000.0  !seconds
```

A description of the cards related to winds and atmospheric pressure are described in the table below.

Table 2-35. CMS-Flow cards related to winds and atmospheric pressure

Input	Format	Notes
Wind Pressure Single File	[card=WIND_PRESSURE_SINGLE_FILE] [name=**WindFile**, type=*char*]	File name and path of the spatially constant wind time series.
Wind Pressure XY file	[card=WIND_PRESSURE_XY_FILE] [name=**windlocfile**, type=*char*]	Name of coordinate file (*.xy).
Wind Pressure Time Increment	[card=WIND_PRESSURE_TIME_INCREMENT] [name=**wtiminc**, type=*float*, units=*sec*]	Specifies the the met file time series increment (SwTimeInc) in seconds
Wind Pressure Grid Parameters	[card=WIND_PRESSURE_GRID_PARAM] [name=**nwindi**, type=*int*, range= >0] [name=**nwindj**, type=*int*, range= >0] [name=**wlatmax**, type=*float*] [name=**wlonmin**, type=*float*] [name=**wlatinc**, type=*int*, range= >0] [name=**wloninc**, type=*int*, range= >0]	Specifies the grid parameters for the wind and pressure file. **nwindi** = number of latitude or y-coordinate. nwindj = number of longitude or x-coordinate **nwindj** = maximum latitude (decimal deg) or y-coordinate of data in met file (< 0 south of the equator). **wlatmax** = maximum latitude(decimal deg) or x-coordinateof data in the met file (< 0 west of Greenwich meridian). **wlonmin** = minimum longitude (decimal deg) or x-coordinateof data in the met file (< 0 west of Greenwich meridian). **wlatinc** = latitude or y-coordinate increment (decimal deg) of data in the met file (must be > 0). **wloninc** = longitude or x-coordinate increment (decimal deg) of data in the met file (must be > 0).

Notes:

- The wind is assumed to start at the start of the CMS simulation.
- The wind grid is assumed to be in the same horizontal coordinate system as the CMS grid.
- The wind velocities must be in m/s and the atmospheric pressure in Pa.
- If no atmospheric pressure data is available, a constant value may be specified. Since only the gradient of the atmospheric pressure is used in forcing the model, a constant atmospheric pressure is equivalent to no atmospheric pressure forcing.

Oceanweather Wind File

The Oceanweather format uses separate ASCII files to specify the wind speeds, atmospheric pressure and grid coordinates. The CMS-Flow cards used to specify the Oceanweather files are described below.

Table 2-36. CMS-Flow cards used to specify the Oceanweather wind, atmospheric pressure and grid coordinates files.

Card	Format	Notes
Oceanweather Wind File	[cards=OCEANWEATHER_WIND_FILE] [name=**OWWndFile**, type=*char*, default=none]	**OWWndFile** = Name of Ocean-weather, Inc. wind file (*.win)
Oceanweather Atmospheric Pressure File	[cards=OCEANWEATHER_PRES_FILE] [name=**OWPresFile**, type=*char*, default=none]	**OWPresFile** = Name of Ocean-weather, Inc. pressure file (*.pre)
Oceanweather Coordiante File	[cards=OCEANWEATHER_XY_FILE] [name=**OWXYFile**, type=*char*, default=none, optional=*true*]	**OWXYFile** = Name of Ocean-weather, Inc. coordinate file (*.xy)

Example 2-21. Excerpt of an Oceanweather wind file.

```
Oceanweather WIN/PRE Format                          2008090912     2008091406
iLat= 118iLong= 201DX=0.0150DY=0.0150SWLat=28.50000SWLon=-96.0000DT=200809091200
-4.8891   -4.8873   -4.8855   -4.8838   -4.8821   -4.8805   -4.8789   -4.8774
-4.8759   -4.8744   -4.8730   -4.8717   -4.8704   -4.8692   -4.8680   -4.8669
...
-2.1530   -2.1382   -2.1233   -2.1085   -2.0937   -2.0788   -2.0637   -2.0486
-2.0335   -2.0184   -2.0035   -1.9887   -1.9741   -1.9598   -1.9458   -1.9322
-1.9190   -1.9062   -1.8938   -1.8820   -1.8707   -1.8600
iLat= 118iLong= 201DX=0.0150DY=0.0150SWLat=28.50000SWLon=-96.0000DT=200809091215
-4.9107   -4.9090   -4.9074   -4.9058   -4.9043   -4.9028   -4.9014   -4.9001
-4.8988   -4.8975   -4.8963   -4.8952   -4.8942   -4.8932   -4.8923   -4.8914
-4.8906   -4.8898   -4.8889   -4.8881   -4.8873   -4.8866   -4.8860   -4.8854
...
```

The last two arguments of the header are the starting and end times in yyyymmddhh. The wind data are preceded by a header line with the grid information. iLat and iLong are latitude North and longitude West. DX and DY are the grid spacing in units of degrees. SWLat and SWLon are the coordinates of the south-west grid point. DT is the actual time in yyyymmddhhMM. First the wind velocities in the x-direction are read then the velocities in the y-direction. The wind speeds are written in m/s.

Example 2-22. FORTRAN code for reading the Oceanweather Wind File.

```
open(nunit,file=windfile,status='old')
read(nunit,*)   !skip  header
11 format(t6,i4,t16,i4)
read(nunit,11) nwindi,  nwindj
backspace(nunit)
 !...
13 format(68x,I4,4(I2))
read(nunit,13,end=333,err=333)  iyear,imonth,iday,ihour,imin
!...
12 format(8f10.0)
read(nunit,12,end=333,err=333) ((wndspdx(i,j),j=1,nwindj),i=1,nwindi)
read(nunit,12,end=333,err=333) ((wndspdy(i,j),j=1,nwindj),i=1,nwindi)
!...
333 close(nunit)
```

Oceanweather Pressure File

The file format for the Oceanweather pressure file (*.pre) is best described with an example.

Example 2-23. Excerpt of an Oceanweather Pressure File.

```
   Oceanweather WIN/PRE Format                          2008090912
2008091406
  iLat= 118iLong= 201DX=0.0150DY=0.0150SWLat=28.50000SWLon=-96.0000DT=200809091200
  1012.8565 1012.8564 1012.8563 1012.8563 1012.8563 1012.8563 1012.8563 1012.8564
  1012.8583 1012.8587 1012.8591 1012.8596 1012.8600 1012.8605 1012.8610 1012.8616
  ...
  1014.2051 1014.2062 1014.2073 1014.2084 1014.2096 1014.2107 1014.2119 1014.2130
  1014.2141 1014.2152 1014.2164 1014.2175 1014.2186 1014.2197
  iLat= 118iLong= 201DX=0.0150DY=0.0150SWLat=28.50000SWLon=-96.0000DT=200809091215
  1012.9089 1012.9089 1012.9089 1012.9089 1012.9090 1012.9091 1012.9092 1012.9094
  1012.9095 1012.9098 1012.9100 1012.9103 1012.9106 1012.9109 1012.9113 1012.9117
  ...
```

The last two arguments of the header are the starting and end times in yyyymmddhh. The atmospheric pressure data (in units of mbar) is preceded by a header line with the grid information. iLat and iLong are latitude North and longitude West. DX and DY are the grid spacing in units of degrees. SWLat and SWLon are the coordinates of the south-west grid point. DT is the actual time in yyyymmddhhMM.

Oceanweather Coordinate File

The file format for the Oceanweather coordinate file (*.xy) should contain the coordinates of the Oceanweather grid in the same coordinate system as the CMS-Flow grid. The order of the points should be from the south-west corner along each row. An example of an Oceanweather Coordinate File is shown below.

```
903301.20526 77626.22617
904769.21902 77663.98766
906237.22790 77701.93744
907705.23188 77740.07550
909173.23093 77778.40184
910641.22504 77816.91646
...
```

Notes:

- The first column is the horizontal coordinate, and the second is the vertical coordinate.
- The coordinates must be in the same coordinate as the CMS.

Tip:

- The easiest way to generate the Oceanweather Coordinate File is to make an ASCII file containing the coordinates in geographical coordinates, bring it into SMS, reproject on the CMS coordinate system and export the coordinates into a file to be used as the Oceanweather Coordinates File. A small Fortran program provided which converts the Oceanweather grid information to an *.xy file on the CIRP website: http://cirp.usace.army.mil/CIRPwiki/images/8/82/WINDLOC.rar

Meteorological Stations

Meteorological stations can be used to force the model. The locations of the stations are specified in the same coordinate system as the CMS-Flow (world coordinates). There are several options and parameters that can be set for each meteorological (met) station. The table below describes the CMS-Flow cards related to the met stations.

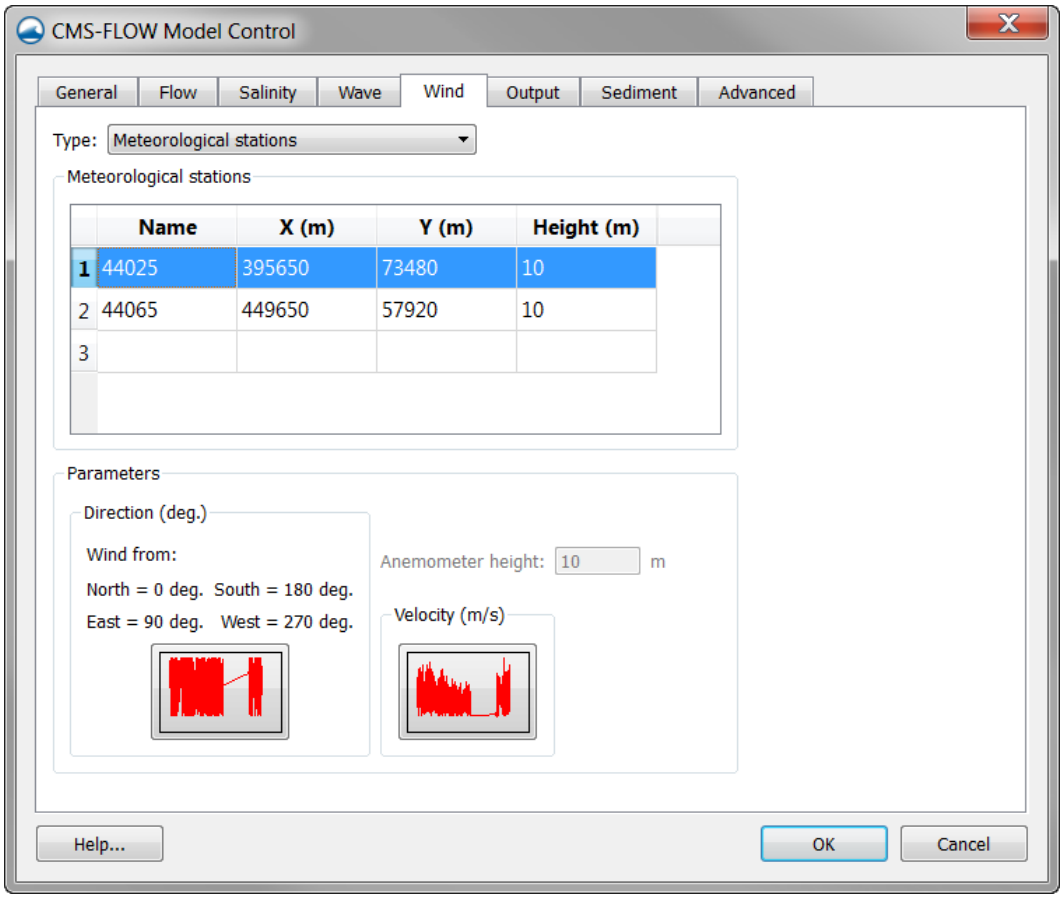


Figure 2-37. CMS-Flow Wind/Wave tab in the CMS-Flow Model Control window.

Table 2-37. CMS-Flow card for specifying the reference frame.

Input	Format	Notes
Meteorological Station Begin card	[card=MET_STA_BEGIN, version=4.1, block=MetStatBlock, childs=(NAME, WIND_INPUT_CURVE, COORDINATES, LATITUDE_LONGITUDE, WIND_VELOCITY_FACTOR, ANEMOMETER_HEIGHT, SMOOTHING_ITERATIONS, SMOOTHING_WIDTH, POWER_PARAMETER,MET_FILE)]	Starts the Meteorological Station Block
Name	[card=NAME] [arg=**metsta%name**, type=*char*, default=**MPFile**] [arg=**WindPath**, type=*char*]	Sets the name of the met station
Wind Input Curve	[card=WIND_CURVE] [arg=**metsta%file**, type=*char*] [arg=**metsta%path**, type=*char*]	File name and path of the spatially constant wind time series.
Coordinates	[card=COORDINATES] [arg=**metsta%xsta**, type=float] [arg=**metsta%ysta**, type=*float*]	Specifies the coordinates in local reference
Latitude / Longitude	[card=LATITUDE_LONGITUDE] [name=**metsta%ylat**, type=*float*] [name=**metsta%xlon**, type=*float*]	Specifies the coordinates in global coordinates
Wind Velocity Factor	[card=WIND_VELOCITY_FACTOR] [arg=**metsta%wndfac**, type=*float*, default=1.0]	Used to convert from different temporal averaging periods.
Anemometer Height	[card=ANEMOMETER_HEIGHT] [arg=**metsta%wndht**, type=*float*, units=*m*, default=10.0] [arg=**metsta%path**, type=*char*]	Defines the height of the anemometer in meters.
Smoothing Iterations	[card=SMOOTHING_ITERATIONS] [arg=**metsta%ismoothiter**, type=*int*, default=0]	Temporal smoothing iterations
Smoothing Width	[card=SMOOTHING_WIDTH] [arg=**metsta%ismoothwidth**, type=*int*, default=0, range=0 or odd]	Temporal smoothing window width. Must be odd number.
Power Parameter	[arg=POWER_PARAMETER] [arg=**metsta%pow**, type=*float*, range=1.0-5.0, default=1.0]	Power for Shepard interpolation.
Meteorological File name	[card=MET_FILE] [arg=**metsta%file**, type=*char*]	
Meteorological Station End card	[card=MET_STA_END, version=4.1, block=MetStatBlock]	Ends the Meteorological Station Block

Example 2-24. Specifying two meteorological stations.

```
MET_STATION_BEGIN
  STATION_NAME        44065
  COORDINATES         464000.0 53050.0
  WIND_CURVE  "Flow_mp.h5" "PROPERTIES/Model Params/Meteorological Stations/44065"
  ANEMOMETER_HEIGHT  10.0 m
MET_STATION_END
MET_STATION_BEGIN
  STATION_NAME        44025 !can be string, integer or floating
  COORDINATES         401490.0 93050.0
  WIND_CURVE  "Flow_mp.h5" "PROPERTIES/Model Params/Meteorological Stations/44025"
  ANEMOMETER_HEIGHT  10.0 m
MET_STATION_END
```

The wind components from meteorological stations is interpolated spatially using inverse-distance interpolation. A description of the interpolation procedure is provided in Appendix H. One advantage of the inverse-distance interpolation is the interpolation weights are independent of the interpolation function, and therefore only need to be calculated once and can be saved for computational efficiency.

Wind Reference Frames

Winds are specified in an Eulerian reference frame with respect to the solid earth. When the wind in the same direction of the currents the wind shear stress is lowered (Figure 2-38). When the wind and currents are in opposing directions, the wind shear stress is increased. For example, in the case of a current velocity of 1 m/s, with an opposing wind speed of 5 m/s, the Eulerian reference frame will give a surface stress proportional to (5 m/s)^2 = 25 m^2/s^2, while the Lagrangian reference frame will produce a surface proportional to (5-(-1) m/s)^2 = 36 m^2/s^2, which is an increase of 44%.

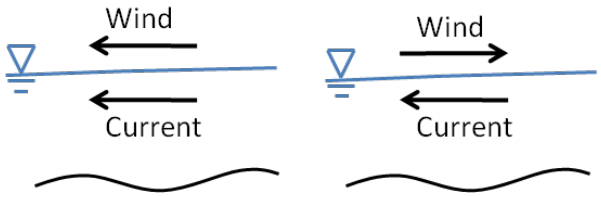


Figure 2-38. Schematic of wind and currents in same (left) and opposing directions (right).

The wind reference frame is specified with the advanced card described in the table below.

Table 2-38. CMS-Flow card for specifying the reference frame.

Card	Arguments	Default	Options	Description
WIND_REFERENCE_FRAME	character	EULERIAN	EULERIAN LAGRANGIAN	Sets the reference frame for wind forcing

Example 2-25. Wind reference frame specification.

```
WIND_REFERENCE_FRAME     LAGRANGIAN    !{EULERIAN} | LAGRANGIAN
```

Notes:

- By default, the wind reference frame is set to Eulerian. It is recommended field applications to change the wind reference frame to Lagrangian.
- The Lagrangian reference frame is especially important for cases with high wind velocities such as storms.

Wind Drag Coefficient

The wind drag coefficient is calculated with the Hsu (1988) formula. The drag coefficient may be modified to scale the wind surface stress. The table below described the cards related to the wind drag coefficient. For further details on the wind drag coefficient see section *Wind Surface Stress* in the technical report.

Table 2-39. CMS-Flow cards related to winds and atmospheric pressure

Card	Arguments	Default	Range	Description
WIND_DRAG_COEFFICIENT	real	0.4	>0	kappa in Hsu (1988) formula for the wind drag coefficient. Scales the wind drag coefficient (Not the wind drag coefficient).
WIND_DRAG_SCALE_FACTOR	real	1.0	0-1	Scaling factor for the wind drag coefficient

Example 2-26. Wind drag coefficient specification

```
WIND_DRAG_COEFICIENT    0.4  !default = 0.4. Depreciated
```

Example 2-27. Wind drag scale factor specification

```
WIND_DRAG_COEFICIENT   1.0   !default = 1.0
```

Anemometer Height

By default, the CMS assumes that the input wind velocities are specified at a 10-m height, if the height is different, the user may enter a different value as the Anemometer height. In SMS 11.0, the anemometer height is specified in the *Wind/Wave* tab of the *CMS-Flow Model Control* window. The wind velocities are internally converted to a 10-m reference height using the 1/7 rule (see Section Wind Surface Stress).

Table 2-40. CMS-Flow cards related to winds and atmospheric pressure

Input	Format	Notes
Anemometer height	[cards=ANEMOMETER_HEIGHT] [name=**WndHgt**, type=*float* default=10.0, range=(**WndHgt**>0)] [name=**WndHgtUnits**, type=*char*, options=**distUnits**, default='m']	Height of wind speed

Example 2-28. Anemometer height specification

```
ANEMOMETER_HEIGHT    6.0 m    !default = 10.0 m, Units optional
```

Wind Velocity Factor

The wind velocity factor may be used for converting the input wind velocity units or converting between different wind velocity definitions. Wind measurements are usually obtained as mean or “maximum sustained” wind in a certain time period. Generally, for hydrodynamic modeling mean wind speeds should be used (e.g. 10-min or 30-min averages). If winds are obtained as maximum sustained winds, they may be converted to averages using the wind velocity factor. Sustained wind speeds may be converted using the wind velocity factor. The wind velocity factor may be specified using an advanced card described in the table below.

Table 2-41. CMS-Flow cards related to winds and atmospheric pressure

Input	Format	Notes
Wind Velocity Factor	[cards=WIND_VELOCITY_FACTOR] [name=**WndFac**, type=*float* default=1.0, range=(**WndFac**>0)]	Specifies a wind velocity factor which directly multiplies by the input wind speeds.

Example 2-29. Wind velocity factor specification

```
WIND_VELOCITY_FACTOR    0.88    !Optional. default = 1.0
```

Boundary Conditions

There are basically two types of lateral (horizontal) boundaries in CMS-Flow: closed and open. Closed boundaries are any boundary between wet and dry cells either at the edge or within the computational grid. Below is a summary of the boundary conditions available in CMS.

1. Wall (closed)
2. Flux (flux distributed along boundary)
    a. Single flux value (advanced cards)
    b. Single flux time-series
3. Water level
    a. Constant along boundary
        (1) Single value (advanced cards)
        (2) Single time-series
        (3) Harmonic series
        (4) Tidal constituents
    b. Variable along boundary
        (1) User-specified (advanced cards)
        (2) Extracted water levels from a larger domain simulation
            (a) Extracted using SMS
            (b) Automatically extracted in CMS (advanced cards)
        (3) Interpolated tidal constituents from a tidal database
            (a) Extracted using SMS
            (b) Automatically extracted in CMS (advanced cards)
4. Mixed water level and current velocity
    a. Variable along boundary
        (1) User-specified (advanced cards)
        (2) Cross-shore boundary (approximate 1D solution)
        (3) Extracted water levels from a larger domain simulation
            (a) Using SMS
            (b) Using CMS (advanced cards)
        (4) Interpolated tidal constituents from a tidal database
            (a) Using SMS
            (b) Using CMS (advanced cards)

The boundary condition types which indicate “advanced cards” are not implemented yet in the latest version of the interface (SMS 11.1) and therefore only available through the advanced cards. It is noted that the water level boundary conditions which are constant along the boundary may be corrected for the wind and wave forcing, or even spatial water level gradients may be added so that in fact the specified water level along the boundary is in general not constant.

The implementation of the mixed water level and current velocity boundary conditions depends on the boundary condition type and the flow conditions. In the case of the cross-shore BC, a water level BC is applied when the flow is inwards of the simulation domain and a flux boundary condition is applied with the flow is outwards. For the other types BC’s the water level and current velocities are applied using the deferred correction approach for the implicit model.

Cellstrings

The spatial location of boundaries conditions is specified using cellstrings which are a series of cell ID's corresponding to the boundary cells on the CMS-Flow grid. Once cellstrings have been created, boundary conditions can be assigned to the cellstring. Cellstrings without assigned boundary conditions or boundaries without cellstrings are assumed to be closed.

Creating and Deleting Cellstrings

When a CMS-Flow grid is generated, cellstrings are automatically generated along all of the boundaries. The user has the option to use any of these cellstrings or to create new cellstrings. Cell strings can either be manually created by clicking on the cells or by using the SMS interface to automatically detect and assign the cell strings.

Manually Creating a Cellstring

1. Select the *Create cellstring* tool .
2. Single-click on the first cell of the cellstring.
3. If necessary, single click on subsequent cells in the cellstring (around corners).
4. Double-click on the last cellstring to finish.

The figure below shows several examples of good and bad cellstrings.

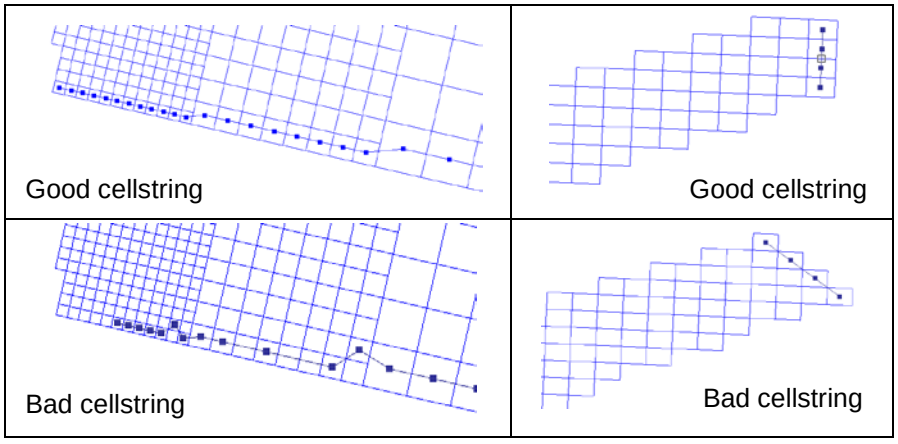


Figure 2-39. Examples of good and bad boundary cellstrings.

Automatically Generate a Cellstring

1. Click on *Cellstring* | *Generate Cellstrings* and select one of the options (Figure 2-40):

   - *Along Open Boundaries*
   - *Along Land Boundaries*
   - *Along Open and Land Boundaries*

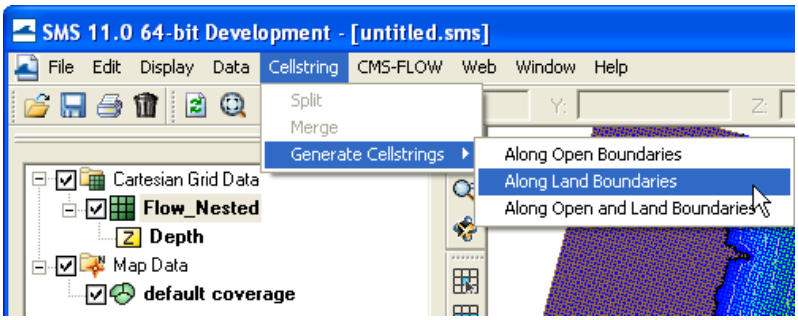


Figure 2-40. Automated Cellstring generation in SMS 11.0

Notes:

1. Cellstrings must be specified at boundaries. Internal boundaries are not allowed.
2. Cellstrings must be at least three cells wide.
3. Cellstrings must be specified at ALL open boundaries. Leaving any open boundaries without an assigned boundary will close that boundary and apply a wall boundary condition.
4. Land boundary conditions are not necessary at close boundaries (cells next to inactive cells). The CMS will automatically detect these land boundaries.
5. The orientation of cellstrings is important. Some inflow and all outflow boundary conditions assume a zero-gradient in the normal direction. If this assumption is not valid, the boundary may produce instability problems. It is recommended that the boundaries be placed in an orientation such that the spatial gradients normal to the cellstring or minimal.
6. Even though cell strings display along the cell centers, the actual boundary conditions are assigned at the cell faces.

Selecting Cellstrings

Once a boundary cellstring is created, the cell needs to be selected before assigning a boundary condition. If a cellstring is not selected, no boundary conditions can be assigned even if there is only one cellstring for the grid. There are basically two ways of selecting a cellstring in SMS. To assign this type of BC in SMS:

1. Enable the cellstring selection tool by left-clicking on the *Select a Cellstring* button in the *Cartesian Module Toolbar*.
2. Select a cellstring by either
    a. Right-clicking on a cellstring handle and select *Assign BC* (see left panel in the Figure below).
    b. Left-click and hold the button while dragging the mouse creating a selection box over the cellstring (see right panel of Figure 2-41).

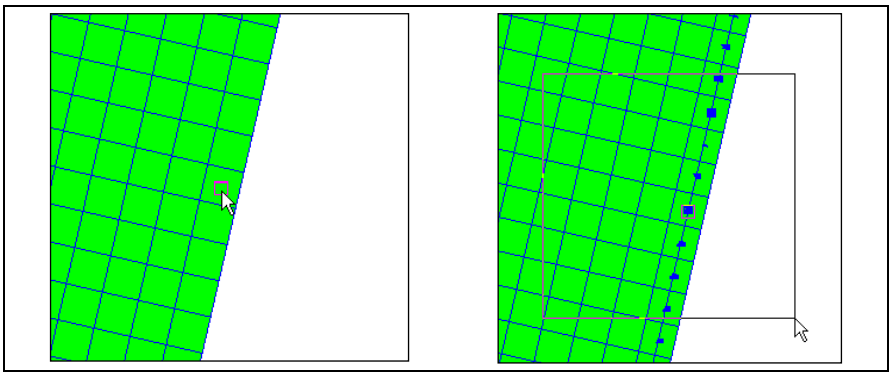


Figure 2-41. Selecting a celstring by either clicking on the cellstring handle (left) or dragging a selection box of the cellstring (right).

Deleting Cellstrings

To delete one or more cellstring:

1. Select one or more cellstrings (see section *Selecting Cellstrings* for details). and open the *CMS-Flow Boundary Conditions Window* (see section Assigning a Boundary Condition for details).
2. Either
   a. While the cellstring is selected right-click anywhere on the t the cellstring and select *Delete* from the list of items in the pop-up menu (Figure 2-42).

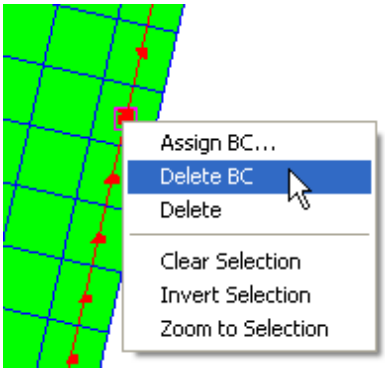


Figure 2-42. Deleting a cellstring by right-clicking on a cellstring and select *Delete BC* from the pop-up menu.

   b. Press the *Delete* key, and then confirm the deletion in the pop-up window (see figure below).

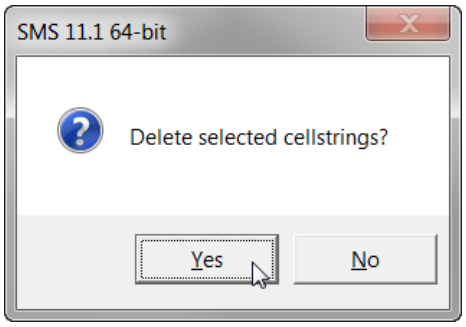


Figure 2-43. Deleting a cellstring in SMS by pressing the *Delete* key and selecting *Yes* in the pop-up window.

Notes:

- The same procedure outlined above can be used to delete multiple cellstrings at once.

Tips:

- The SMS interface has the option to turn on or off the display of unselected cell strings. To change this option click on *Display* | *Display Options* or the *Display Options Icon* , and under the *Cartesian Grid* options, select the check box for *Cellstrings*, and the appropriate boundary condition types.

Assigning Boundary Conditions

CMS-Flow has several types of boundary conditions which are listed and discussed below. The figure below shows the CMS-Flow Boundary Conditions Window in SMS. All CMS-Flow boundary conditions are forced at the edges of the domain by use of cellstrings defined with the Surface water Modelling System (SMS). Cellstrings can either be created manually or using the SMS tool called "Generate Along Boundary" which is found under the "Cellstring" menu.

Once a cellstring is selected, the boundary condition can be assigned in the *CMS-Flow Boundary Conditions Window* by:

1. Select a cellstring (see *Selecting Cellstrings* section)
2. Open the *CMS-Flow Boundary Conditions* window by either:
    a. Right-clicking on a cellstring handle and selecting the *Assign BC* option (see Figure 2-44).

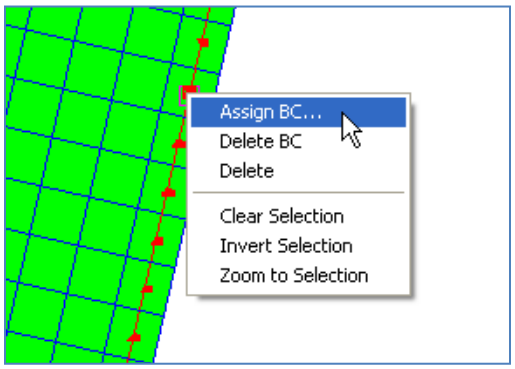


Figure 2-44. Opening the CMS-Flow Boundary Conditions window by left clicking.

    b. Click on the *CMS-Flow* menu, and select *Assign BC* (Figure 2-45).

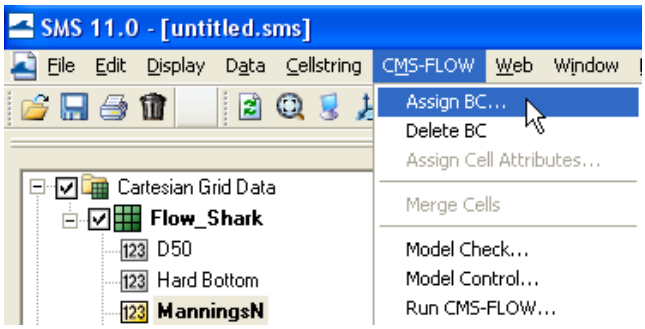


Figure 2-45. Opening the CMS-Flow Boundary Conditions Window through the CMS-Flow menu.

An example view of the how the *CMS-Flow Boundary Conditions Window* is shown in Figure 2-46 below. The sections below describe in detail each type of boundary condition and the information required.

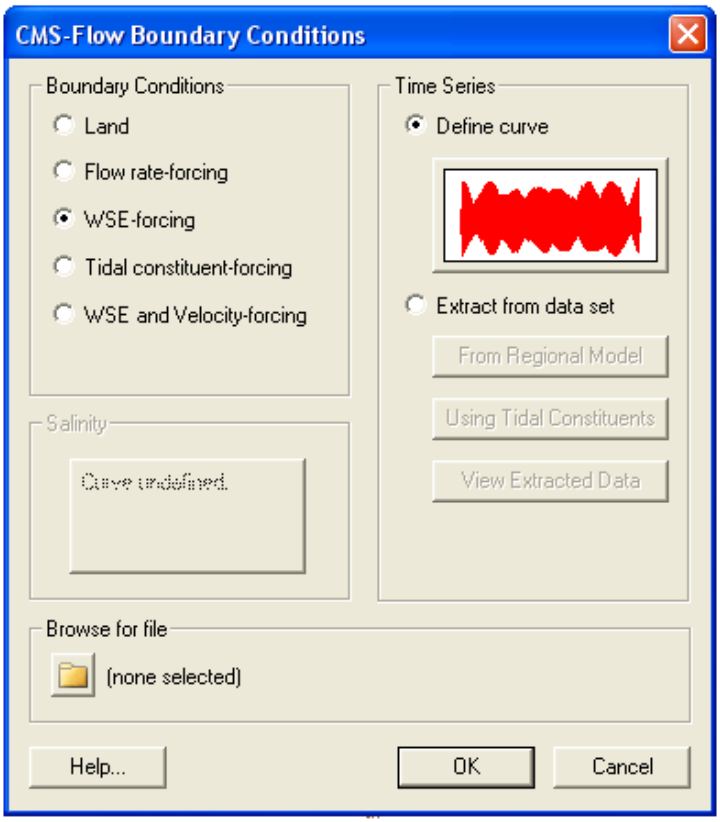


Figure 2-46. CMS-Flow Boundary Conditions Window in SMS 11.0.

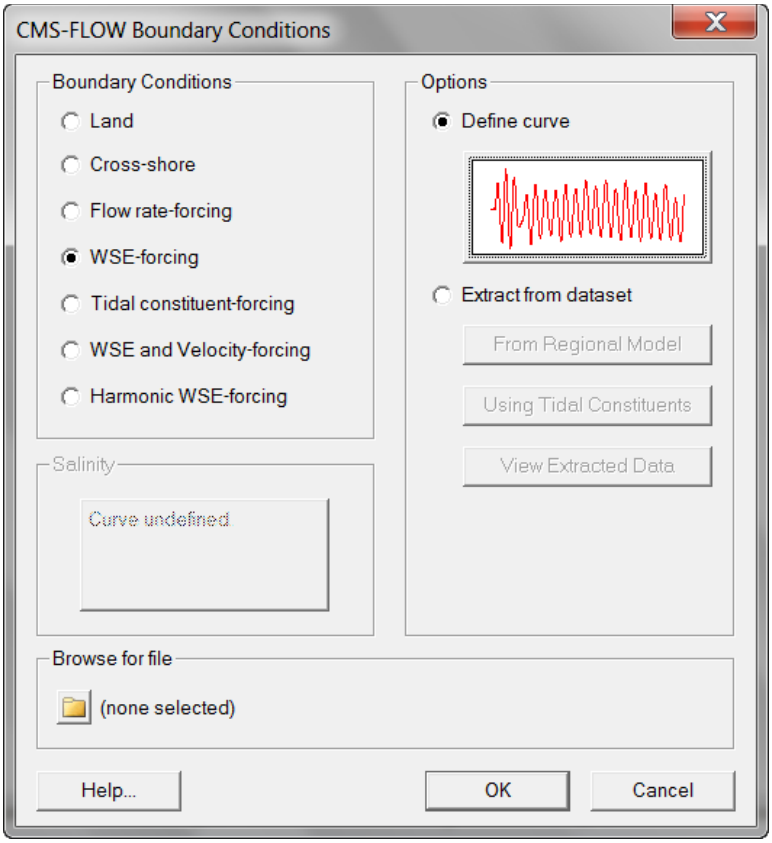


Figure 2-47. CMS-Flow Boundary Conditions Window in SMS 11.1.

Wall Boundary

The wall boundary condition is a closed boundary and is applied at any cell face between wet and dry cells. Any unassigned boundary cells at the edge of the model domain are assumed to be closed and are assigned a wall boundary. A zero normal flux to the boundary is applied at closed boundaries. Two boundary conditions are available for the tangential flow:

1. Free-slip: No tangential shear stress (wall friction)
2. Partial-slip: Tangential shear stress (wall friction) calculated based on the log-law

Assuming a log-law for a rough wall, the partial-slip tangential shear stress is given by

$$\tau_{wall} = \rho c_{wall} U_{\|}^2 \tag{2-10}$$

where $U_{\|}$ is the magnitude of the wall parallel current velocity and c_{wall} is the wall friction coefficient equal to

$$c_{wall} = \left[\frac{\kappa}{\ln(y_P / y_0) - 1} \right]^2 \tag{2-11}$$

Here y_0 is the roughness length of the wall and is assumed to be equal to that of the bed (i.e. $y_0 = z_0$). y_P is the distance from the wall to the cell center.

Notes:

- When SMS generates a new grid, cellstrings are automatically generated along all boundaries. The default boundary type is the wall boundary. It is recommended to delete all boundary cellstrings which are not open boundary conditions, since the wall boundary conditions are automatically assigned within the model. Deleting the cellstrings avoids confusion when assigning the boundary conditions.
- Since the wall boundary is a closed boundary and all unassigned boundaries are treated as closed, the wall boundary cellstring information does not need to written to the CMS-Flow Card File.

Flux Boundary

The flux boundary condition is typically applied to the upstream end of a river or stream and is specified as either a constant or time-series of total water volume flux. In a 2DH model, the total volume flux needs to be distributed across the boundary in order to estimate the depth-averaged velocities. This is done using a conveyance approach in which the current velocity is assumed to be related to the local flow depth h and Manning's n as (i.e. $U \propto h^r / n$). Here r is an empirical conveyance coefficient equal to approximately 2/3 for uniform flow. The smaller the r value the more uniform the current velocities are across the flux boundary. The water volume flux, q_i , at each boundary cell i is calculated as

$$\vec{q}_i = h\vec{U}_i = \frac{f_{Ramp} Q}{\left| \sum_i (\hat{e} \cdot \hat{n}) \frac{h_i^{r+1}}{n_i} \Delta l_i \right|} \frac{h_i^{r+1}}{n_i} \hat{e} \qquad (2\text{-}12)$$

where

i = subscript indicating a boundary cell

$\vec{q}_i$ = volume discharge at boundary cell i per unit width [m^2/s]

$\hat{e}$ = unit vector for inflow direction $= (\sin\varphi, \cos\varphi)$

φ = inflow direction measured clockwise from North [deg]

$\hat{n}$ = boundary face unit vector (positive outward)

Q = total volume flux across the boundary [m^3/s]

n = Manning's coefficient [$s/m^{1/3}$]

r =empirical constant equal to approximately 2/3

Δl =cell width in the transverse direction to flow [m]

f_{Ramp} = ramp function [-]

The total volume flux is positive into the computational domain. Since it is not always possible to orient all flux boundaries to be normal to the flux boundaries, the option is given to specify an inflow direction φ. The angle is specified in degrees clockwise from true North. If the angle is not specified, then the inflow angle is assumed to be normal to the boundary. The total volume flux is conserved independently of the inflow direction.

As mentioned above, the total water volume flux Q_B may be specified as a constant value or a single time-series curve. A description of the CMS-Flow cards used to specify the flux boundary condition information is provided in the table below.

Table 2-42. CMS-Flow cards related to the flux boundary condition.

Input	Format	Notes
Flux boundary	[card=QDRIVER_CELLSTRING] [name=**Qfile**, type=*char*, default=**none**] [name=**Qpath**, type=*char*, default=**none**]	Specifies the Boundary ID File and Path (if necessary). Outdated in CMS V4.1 and newer. Has been replaced by block structure format.

Example 1: Single line flux boundary specification

```
QDRIVER_CELLSTRING   "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
```

Step-by-Step Flux Boundary Specification in SMS

The flow rate boundary condition specifies a time series of water fluxes in units of m^3/s per cell. To assign a flux BC in SMS:

1. If not already created by SMS, create the cellstring at the flux boundary (see section *Creating and Deleting Cell Strings in SMS* for details).
2. Select the cellstring (see section *Selecting a Cellstring* for details)
3. Open the *CMS-Flow Boundary Conditions Window* (see section *Assigning a Boundary Condition* for details).
4. Enter a time series of total flux values on and select *OK* button.
5. Save the *SMS project File* (*.sms) or *CMS-Flow Simulation File* (*.cmcards).

Total flow rate specified is divided between the total number of cells in the cellstring with each assigned a portion of the total flux as function of the local water depth and bottom friction.

Notes:

- The flux boundary type may ONLY be specified along cellstrings which are straight (i.e. the cellstring may not wrap around corners).
- Cellstrings have to be at least three cells long.
- Positive fluxes are directed inward and negative fluxes are directed outward. By default the inflow/outflow angles are assumed to be normal to the boundary. The angle may be changed using the advanced block structure described in the following section.
- If the flux data file name and path are not specified, then they are assumed to be the same as for cellstring.

Water Level Boundaries

The water level at a boundary can be specified as:

1. Constant value
1. Single time-series curve
2. Cell-specific time-series curves (i.e. one time-series for each boundary cell)
3. Tidal constituents
4. Harmonic constituents
5. Extracted values from a parent grid simulation (nesting)
6. Tidal database

The general formula for the boundary water surface elevation is given by

$$\overline{\eta}_B = f_{Ramp}(\overline{\eta}_E + \Delta\overline{\eta} + \overline{\eta}_C + \overline{\eta}_G) + (1 - f_{Ramp})\overline{\eta}_0 \qquad (2\text{-}13)$$

where

$\overline{\eta}_B$ = boundary water surface elevation [m]

$\overline{\eta}_E$ = external boundary water surface elevation [m]

$\Delta\overline{\eta}$ = water surface elevation offset [m]

$\overline{\eta}_0$ = initial boundary water surface elevation [m]

$\overline{\eta}_C$ = correction to the boundary water surface elevation which is a function of the wind and wave forcing [m]

$\overline{\eta}_G$ = water surface elevation component derived from user specified gradients [m]

f_{Ramp} = ramp function [-]

The external water surface elevation may be spatially and temporally constant or variable. When a time series is specified, the values are interpolated using piecewise Lagrangian polynomials. By default, second order interpolation is used, but can be changed by the user. The water surface elevation offset $\Delta\overline{\eta}$ is assumed spatially and temporally constant and may be used to correct the boundary water surface elevation for vertical datums, surge, and sea level rise. The correction $\overline{\eta}_C$ is only applicable when $\overline{\eta}_E$ is spatially constant as in the case of a single water surface elevation time-series. The component $\overline{\eta}_G$ is intended to represent regional gradients

in the water surface elevation, is assumed to be constant in time, and is only applicable when $\overline{\eta}_E$ is spatially constant.

A small degree of relaxation is obtained by applying the water level forcing as a source term rather than assigning the water level at the boundaries. This technique is common practice in finite volume models and improves stability and convergence. When applying a water level boundary condition to the nearshore, local flow reversals and boundary problems may result if the wave-and wind-induced setup are not included. This problem is avoided by adjusting the local water level to account for the cross-shore wind and wave setup similar that described in Reed and Militello (2005).

Tidal Boundary

Tidal or astronomic water level predictions are based on the official United States National Oceanographic and Atmospheric Administration (NOAA) Tides and Currents website: http://tidesandcurrents.noaa.gov and using the National Ocean Service (NOS) (http://co-ops.nos.noaa.gov) prediction formula

$$\overline{\eta}_E(t) = \sum f_i A_i \cos\left(\omega_i t + V_i^0 + \hat{u}_i - \kappa_i\right) \tag{2-14}$$

where

i = subscript indicating a tidal constituent

A_i = mean amplitude [m]

f_i = node (nodal) factor [-]

ω_i = frequency [deg/hr]

t = elapsed time from midnight of the starting year [hrs]

$V_i^0 + \hat{u}_i$ = equilibrium phase [deg]

κ_i = phase lag or epoch [deg]

The nodal factor is a time-varying correction to the mean amplitude. The equilibrium phase has a uniform component V_i^0 and a relatively smaller periodic component. The zero-superscript of V_i^0 indicates that the constituent phase is at time zero. The table below provides a list of tidal constituents currently supported in CMS. More information on U.S. tidal constituent values can be obtained from NOAA and NOS.

Table 2-43. Tidal Constituents names (cname) and speeds (in degrees per solar hour) implemented in CMS.

Constituent	Speed	Constituent	Speed	Constituent	Speed	Constituent	Speed
SA*	0.041067	SSA*	0.082137	MM*	0.54438	MSF*	1.0159
MF*	1.098	2Q1*	12.8543	Q1*	13.3987	RHO1*	13.4715
O1*	13.943	M1*	14.4967	P1*	14.9589	S1*	15.0
K1*	15.0411	J1*	15.5854	OO1*	16.1391	2N2*	27.8954
MU2*	27.9682	N2*	28.4397	NU2*	28.5126	M2	28.9841
LDA2*	29.4556	L2*	29.5285	T2*	29.9589	S2	30
R2*	30.0411	K2	30.0821	2SM2*	31.0159	2MK3*	42.9271
M3*	43.4762	MK3*	44.0252	MN4*	57.4238	M4	57.9682
MS4*	58.9841	S4*	60.0	M6	86.9523	S6*	90.0
M8*	115.9364						

* Only available through advanced cards for CMS >v4.0

Table 2-44. CMS-Flow Cards related to the Tidal Boundary specification.

Input	Format	Notes
Begins a Tidal Constituents Boundary Block	[begin=TIDAL_CONSTITUENTS_BEGIN, name=**TCblock**]	Begins the block structure for a flux boundary condition
Tidal constituent card time series file	[card=TIDAL_CONSTITUENT_(**cname**), parent=**TCblock**] [name=**WSEfile**, type=*char*, default=**none**] [name=**WSEpath**, type=*char*, parent=**WSEblock**, default=**none**]	Specifies the WSE Data File Name (including path if different from Card File) and Dataset Path (for XMDF files only). Optional. Assumed to be the same as the cell-string if not specified. The file must contain a time-series for each boundary cell.
Ends Tidal Constituents Boundary Block	[end=TIDAL_CONSTITUENTS_END, name=**TCblock**]	Ends the block structure for a flux boundary condition
Tidal boundary cellstring	[card=TIDAL_CELLSTRING] [name=**Hfile**, type=*char*, default=**none**] [name=**Hpath**, type=*char*, default=**none**]	Specifies the Boundary ID File and Path (if necessary). Outdated in CMS V4.1 and newer. Has been replaced by block structure format.

In the example below, the amplitude and phase of tidal constituents are specified. The tidal constituents are assigned to the tidal boundary condition specified by the TIDAL_CELLSTRING card.

Example 2-30. Tidal boundary condition specification.

```
TIDAL_CONSTITUENTS_BEGIN
   TIDAL_CONSTITUENT_M2   0.75 211.0  ![Amplitude (m)] [Phase (deg)]
   TIDAL_CONSTITUENT_S2   0.18 231.0  ![Amplitude (m)] [Phase (deg)]
   TIDAL_CONSTITUENT_K1   0.15 185.0  ![Amplitude (m)] [Phase (deg)]
   TIDAL_CONSTITUENT_O1   0.24 191.0  ![Amplitude (m)] [Phase (deg)]
TIDAL_CONSTITUENTS_END
TIDAL_CELLSTRING   "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
```

Notes:

- The order of the tidal constituents does not matter.
- In the example above the tidal constituent cards are indented for better readability. He SMS interface version 11.1 and earlier do not indent these cards. The model is not sensitive to the amount of indentation.
- It is only possible to assign one tidal boundary condition using this card format. In order to have more than one tidal boundary condition, the advanced cards must be used described in Section “Advanced Boundary Specification”.

Harmonic Boundary

If a harmonic boundary condition is applied, then the node factors are set to 1.0 and the equilibrium arguments are set to zero. Similarly to the water level boundary condition, the local water level at the boundary is adjusted to account for the wind and wave setup in order to avoid local flow reversals or instabilities (Reed and Militello 2005).

$$\bar{\eta}_E(t) = \sum A_i \cos\left(\omega_i t - \kappa_i\right) \tag{2-15}$$

where

i = subscript indicating a tidal constituent

A_i = mean amplitude [m]

ω_i = frequency [deg/hr]

t = elapsed time from midnight of the starting year [hrs]

κ_i = phase [deg]

The nodal factor is a time-varying correction to the mean amplitude The equilibrium phase has a uniform component V_i^0 and a relatively smaller periodic component. The zero-superscript of V_i^0 indicates that the constituent phase is at time zero.

Table 2-45. CMS-Flow Cards related to the Harmonic Boundary specification.

Input	Format	Notes
Begins a Harmonic Boundary Block	[begin=HARMONIC_BEGIN, name=**Harmblock**]	Begins the block structure for a flux boundary condition
Harmonic component amplitude, phase, and speed	[card=HARMONIC_COMPONENT_(**HarmJ**), parent=**Harmblock**)] [arg=**HarmAmp(HarmJ)**, type=*real*, parent=**Harmblock**, default=**none**] [arg=**HarmPha(HarmJ)**, type=*real*, parent= **Harmblock**,default=**none**]	Specifies the harmonic component amplitude (m), phase (deg), and speed (deg/hr). **HarmJ** is an integer indicating the harmonic component.
Ends a Harmonic Boundary Block	[end=HARMONIC_END, name=**TCblock**]	Ends the block structure for a flux boundary condition
Harmonic boundary cellstring	[card=HARMONIC_CELLSTRING] [arg=**Hfile**, type=*char*, default=**none**] [arg=**Hpath**, type=*char*, default=**none**]	Specifies the Boundary ID File and Path (if necessary). Outdated in CMS V4.1 and newer. Has been replaced by block structure format.

Example 2-31. Harmonic boundary condition specification (SMS 11.1).

```
HARMONIC_BEGIN
   HARMONIC_COMPONENT_1  1.0  0.0 28.9841 ![Amp(m)] [Phase(deg)] [Speed(°/hr)]
   HARMONIC_COMPONENT_2  0.5 20.0 30.000  ![Amp(m)] [Phase(deg)] [Speed(°/hr)]
HARMONIC_END
HARMONIC_CELLSTRING   "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
```

Notes:

- The phases used in the harmonic boundary conditions are specified with respect to the start of the simulation.
- The speeds for the major tidal constituents can be found on the NOAA Tides and Currents Website: http:// http://tidesandcurrents.noaa.gov/.

Water Level Time Series Curve

For many coastal applications, a single water level time-series curve is sufficient. Water level measurements are usually obtained from a nearby tidal gauge. A water level time series can be specified in the *CMS-Flow Boundary Conditions* window (see Figure below).

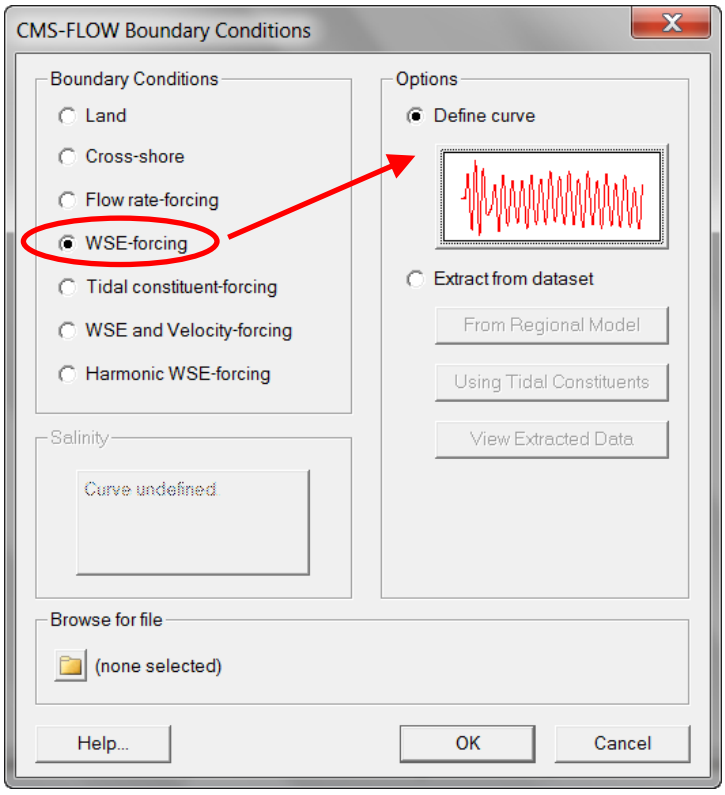


Figure 2-48. Single water level time-series specification in SMS 11.1.

Table 2-46. CMS-Flow cards related to the single water level time series boundary condition.

Input	Format	Notes
WSE time series boundary	[card=HDRIVER_CELLSTRING] [name=**Hfile**, type=*char*, default=**none**] [name=**Hpath**, type=*char*, default=**none**]	Specifies the Boundary ID File and Path (if necessary). Outdated in CMS V4.1 and newer. Has been replaced by block structure format.

Example 2-32. Single WSE time-series BC specification using a single card

```
HDRIVER_CELLSTRING  "Flow_Shark_mp.h5" "PROPERTIES/Model Params/Boundary_#1"
```

Note:

- Careful attention is needed when using a single water level time series. If the location of the measured water levels is far away from the model boundary, this may result in significant amplitude and phase errors. In some cases, if the errors or minor, it is possible to adjust the measured water levels by comparing measured and calculated water levels at other water level stations and adjusting the reference time and scaling the amplitude of the measured water level time series. However, this approach is very simplistic and caution should always. If the amplitude or phase errors are significant it not recommended to attempt any corrections and instead use a tidal database to force the model.

Tips:

- It is recommended to install and use HDFView to view the CMS XMDF files. HDFView can be used to open and even edit the boundary condition data without opening SMS. The figure below shows an example of a CMS-Flow Model Parameters File with a single water level time series curve specified. The data tree on the left-hand-side can be expanded to explore the datasets. In the figure below, the datasets for the water level times and values are shown.

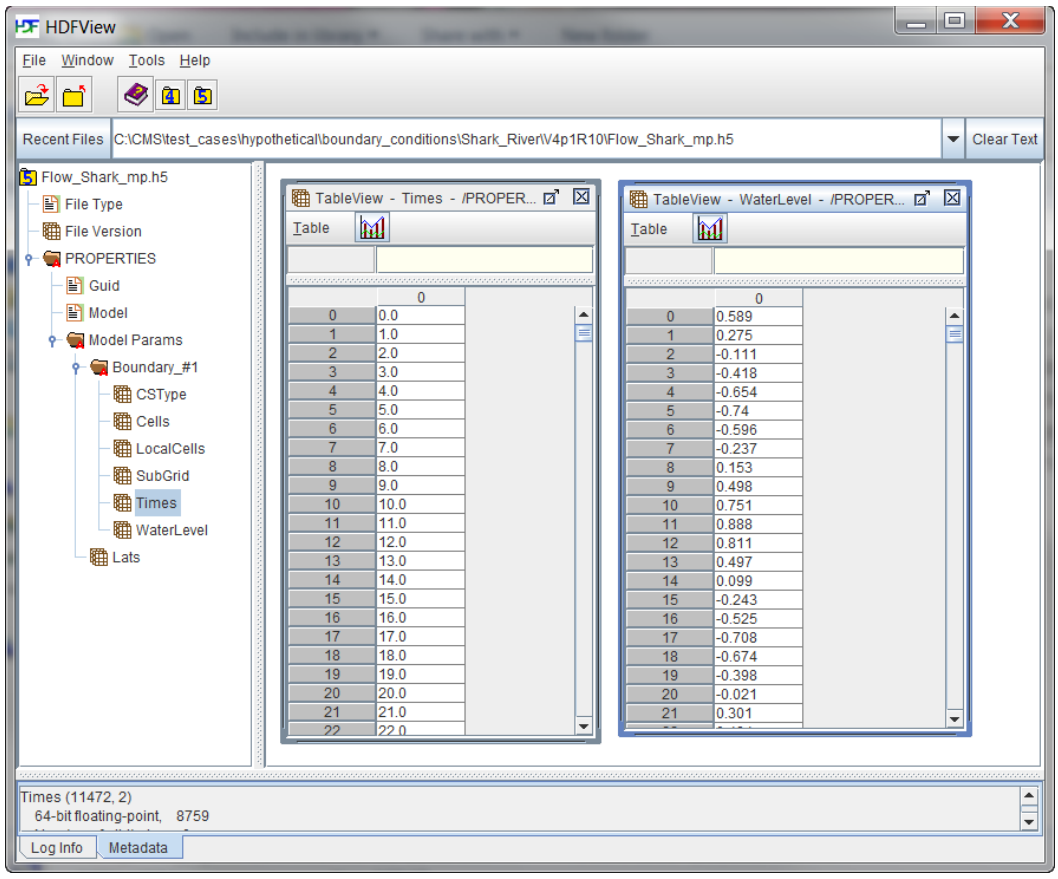


Figure 2-49. Single Water Level Time Series data structure within the CMS-Model Parameters File.

Step-by-Step Instructions: Assigning a Single Water Level Time Series

This is the option called *Define Curve* in SMS. In this BC, a single time series of water levels is specified and is applied at all of the cells along a boundary cell string. The time series curve may be specified by either importing an SMS *.xys file, copying tabular data into SMS, or manually entered the time series information in SMS. To assign this type of BC in SMS:

1. If not already done, load a CMS-Flow grid in SMS.
2. If not already created by SMS, create the cellstring at the boundary (see section *Creating and Deleting Cell Strings in SMS* for details).
3. Select the cellstring (see section *Selecting a Cellstring* for details) and
4. Open the *CMS-Flow Boundary Conditions Window* (see section *Assigning a Boundary Condition* for details). See Figure 2-50.

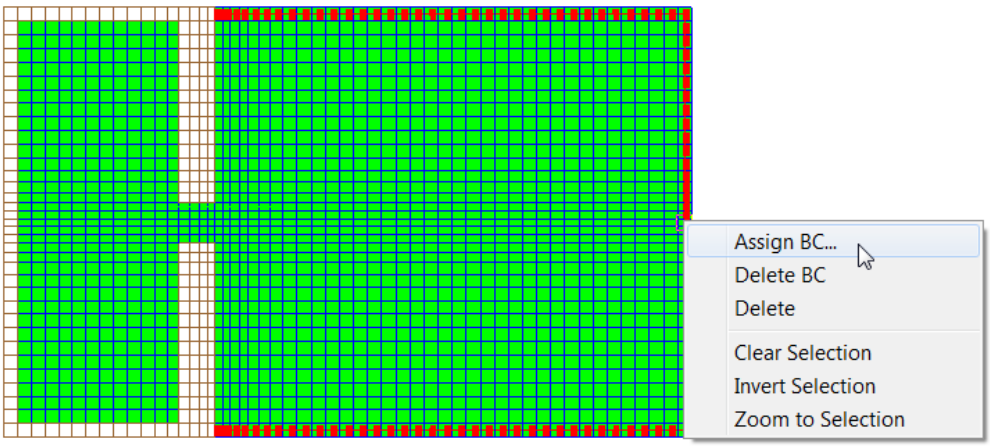


Figure 2-50. Selecting the cellstring along the ocean boundary, and clicking on Assign BC.

5. Select the *WSE-forcing* boundary condition, and click curve undefined (see Figure 2-50) under *Define Curve*.

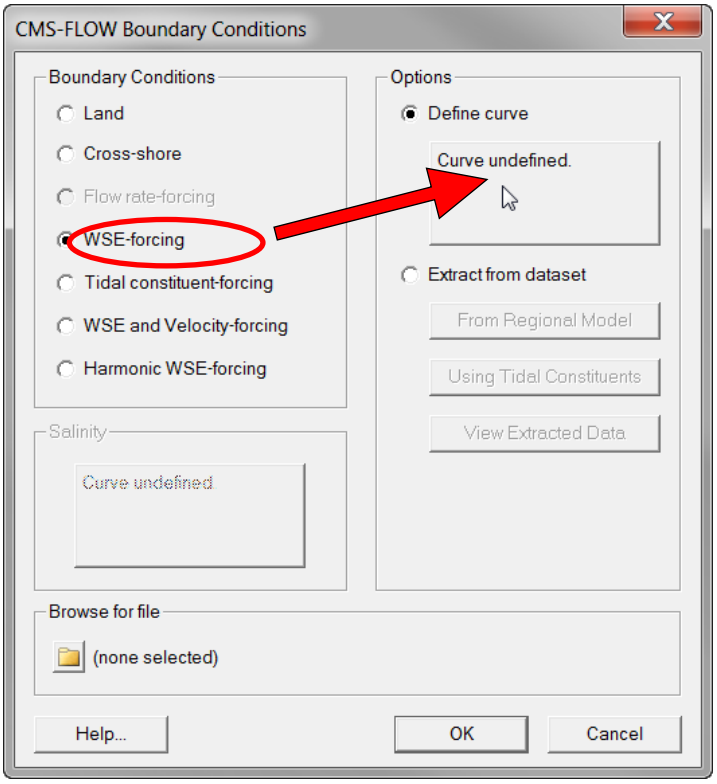


Figure 2-51. Defining the single water surface elevation curve.

6. The *XY Series Editor* will open.
7. Enter the time-series data in the left side of the window either by:
    a. Copy-pasting a time series from a spreadsheet,
    b. Manually entering the times and WSE values, or
    c. Importing an SMS xys file by clicking on the *Import...* button in which the SMS interface will prompt the user to select the appropriate file.

8. The time series should appear in the plot on the right-side of the window (see figure below).

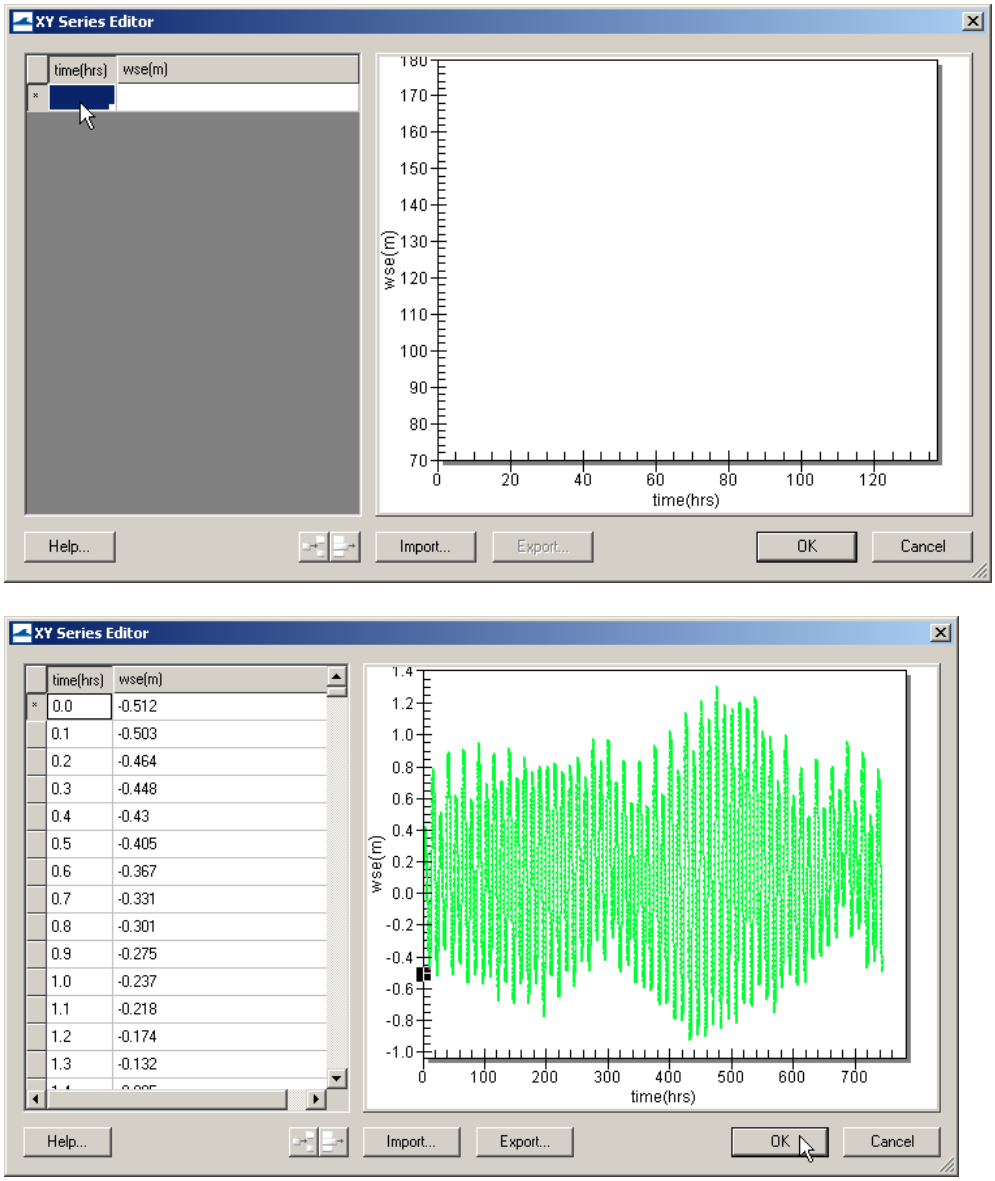


Figure 2-52. Top: is the blank *XY Series Editor*; Bottom: after the data is pasted in.

9. If the user selects to, the boundary time series may be saved to a file by click ingon the *Export...* button.
10. Save the *SMS project File* (*.sms) or *CMS-Flow Simulation File* (*.cmcards).

Notes:

- When importing an SMS xys file, it is important to note that the SMS reads the file and loads the data rather than linking to that file. If the file is changed, the boundary condition in SMS will NOT automatically update.
- The water level time series do not have to have regular spacing.
- The water level time series should cover the time extent of the simulation. If the time series does not the model will still run and use the nearest value from time series.

Tips:

- When specifying the time series forcing in SMS manually or by copying from an spreadsheet, it is recommended to save the time series forcing in SMS XY Series Files (*.xys). If a new grid is created, the file can be easily reloaded in SMS. In addition, these files serve as a record of the model forcing; serve as a backup in case the Model Parameters File becomes corrupt.
- For most coastal applications it a time series interval of about 5 to 30 minutes is sufficient.

Extracted Water Level Time Series Boundary

This BC type is named *Extract from dataset* in SMS 11.0 and 11.1 (see figure below). In this boundary condition type, an individual water level time-series is specified for each boundary cell. The SMS interface allows the user to extract these time series from a larger CMS or ADCIRC simulation or to synthesize them from interpolated tidal constituents from a tidal database.

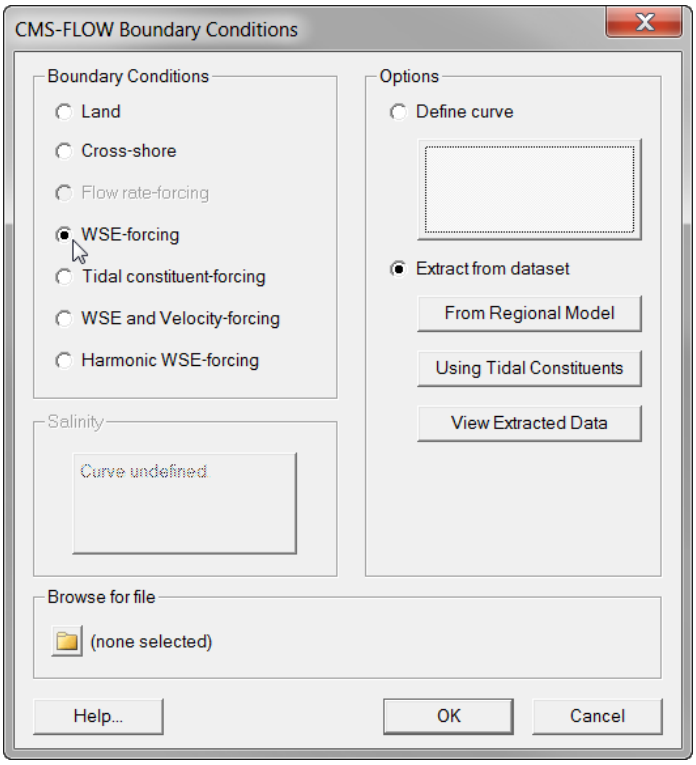


Figure 2-53. CMS-Flow Boundary Conditions window in SMS 11.1 with an assigned WSE-Forcing from an extracted dataset.

The CMS-Flow card used to specify the extracted water level boundary condition is described in the table below.

Table 2-47. CMS-Flow cards related to the extracted water level time series boundary condition.

Input	Format	Notes
Extracted WSE time series boundary	[card=MDRIVER_CELLSTRING] [name=**MHfile**, type=*char*, default=**none**] [name=**MHpath**, type=*char*, default=**none**]	Specifies the Boundary ID File and Path (if necessary).

The extracted water levels are stored in the *CMS-Flow Model Parameters* file. Below is an example of how the CMS-Flow card for the extracted water level times boundary.

Example 2-33. Extracted water level time series boundary specification using a single card.

```
MDRIVER_CELLSTRING    "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
```

Step-by-Step Instructions: Extracted WSE from a Regional Model

To assign this type of BC in SMS:

1. If not already done, load a CMS-Flow grid in SMS.

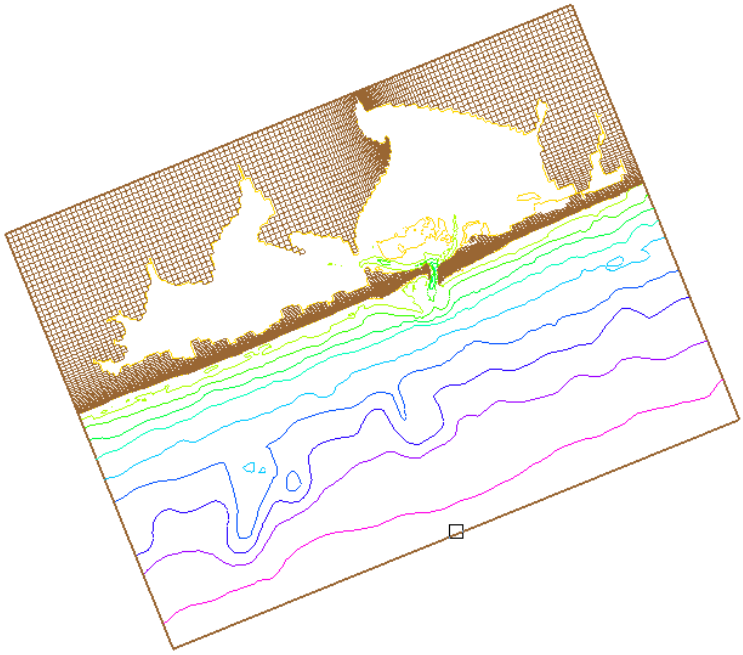


Figure 2-54. CMS-Flow grid for Shinnecock Inlet, NY.

2. If not already done, load in a larger-domain (regional) simulation files including the water level solution (typically either an CMS or ADCIRC simulation).

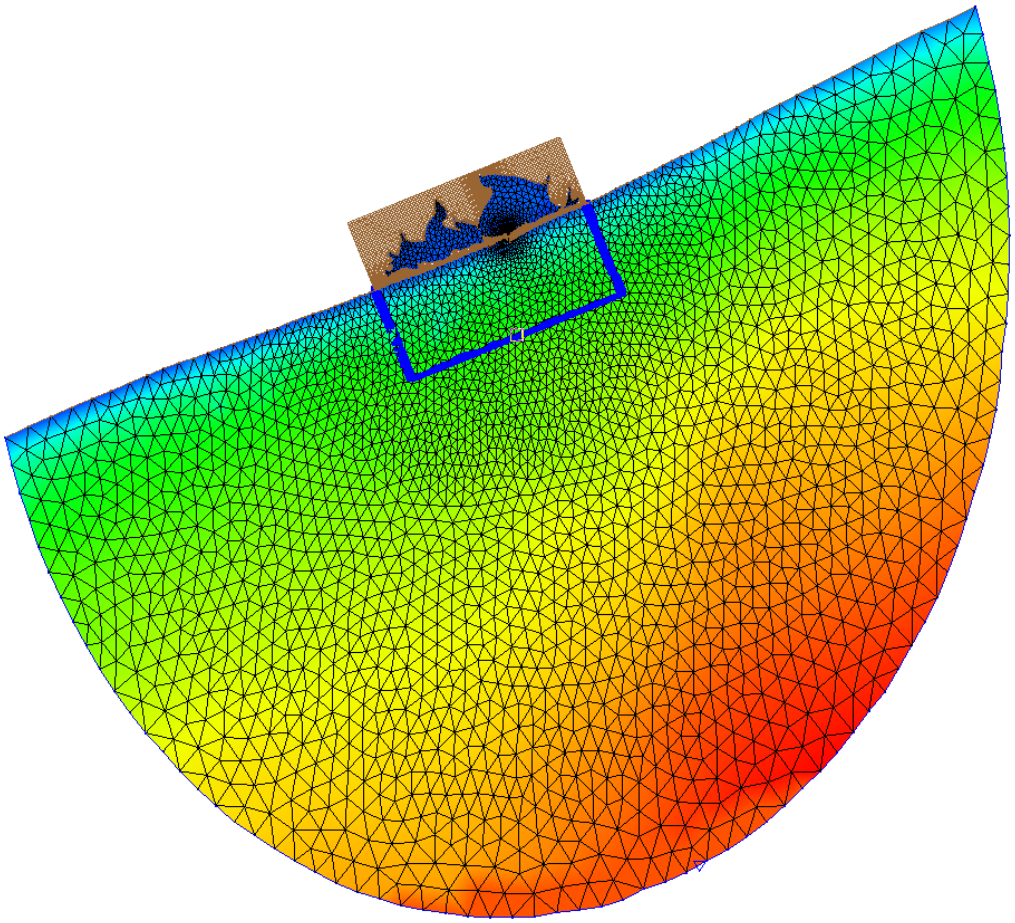


Figure 2-55. Local CMS-Flow grid and larger-domain ADCIRC mesh for Shinnecock Inlet, NY.

3. If not already created by SMS, create the CMS-Flow cellstring at the boundary (see section *Creating and Deleting Cell Strings in SMS* for details).
4. Select the CMS-Flow cellstring (see section *Selecting a Cellstring* for details).
5. Open the *CMS-Flow Boundary Conditions Window* (see section *Assigning a Boundary Condition* for details). See figure below.

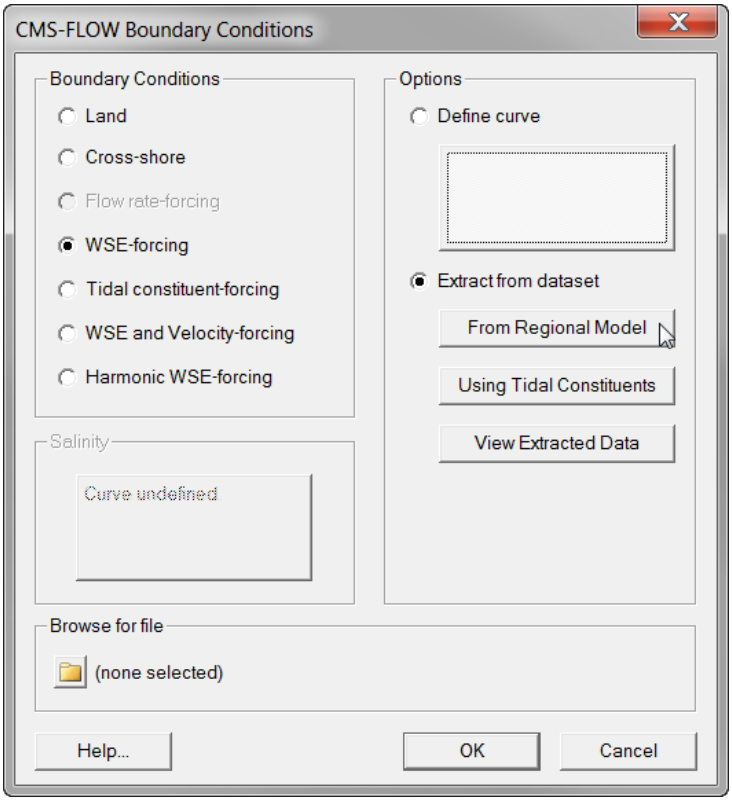


Figure 2-56. CMS-Flow Boundary Conditions window in SMS 11.1 with an WSE boundary condition.

6. Select *WSE-forcing* in the *Boundary Conditions* section on the left side of the window.
7. Select *Extract from dataset* in the *Options* section on the right side of the window.
8. Click on *From Regional Model*. The Extract Boundary Conditions window will appear (see example figure below).
9. Select the water level forcing dataset and the start and end times for the extraction.
10. Select the *Extract* button in the *Extract Boundary Conditions* window. The window will close.

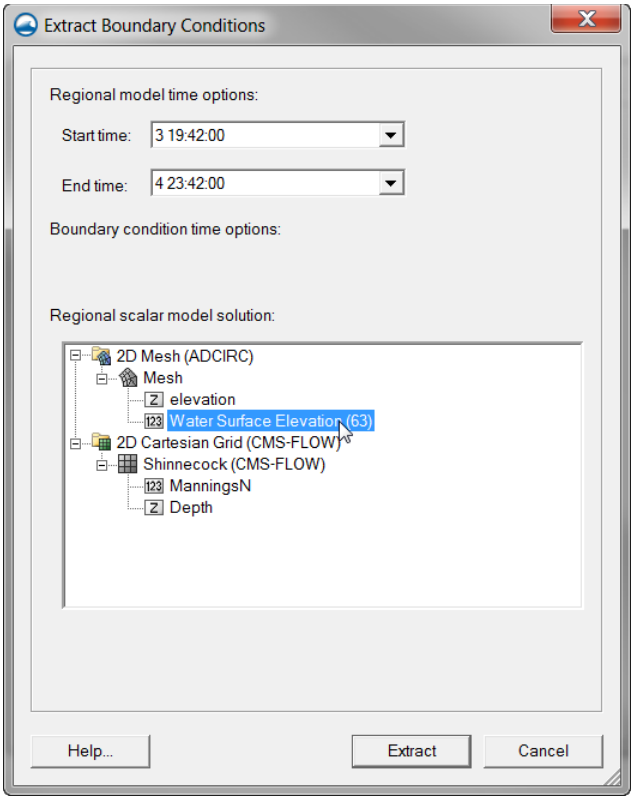


Figure 2-57. *Extract Boundary Conditions* window in SMS 11.1.

11. In the CMS-Flow Boundary Conditions window, select the *OK* button.
12. Save the *SMS project File* (*.sms) or *CMS-Flow Simulation File* (*.cmcards).

Notes:

- Due to their size, the tidal databases do come installed with SMS, and must be downloaded separately. Before using the tidal databases in SMS for the first time it is necessary to follow the steps below.
- If an ADCIRC simulation is used to extract water levels, make sure the ADCIRC grid is in the same horizontal projection as the CMS-Flow grid.

Step-by-Step Instructions: Extracted WSE using Tidal Constituents

To assign an extracted water surface elevation boundary using tidal constituents from a tidal database:

1. If not already done, load a CMS-Flow grid in SMS.
2. If not already created by SMS, create the CMS-Flow cellstring at the boundary (see section *Creating and Deleting Cell Strings in SMS* for details).
3. Select the CMS-Flow cellstring (see section *Selecting a Cellstring* for details).
4. Open the *CMS-Flow Boundary Conditions Window* (see section *Assigning a Boundary Condition* for details). See figure below.

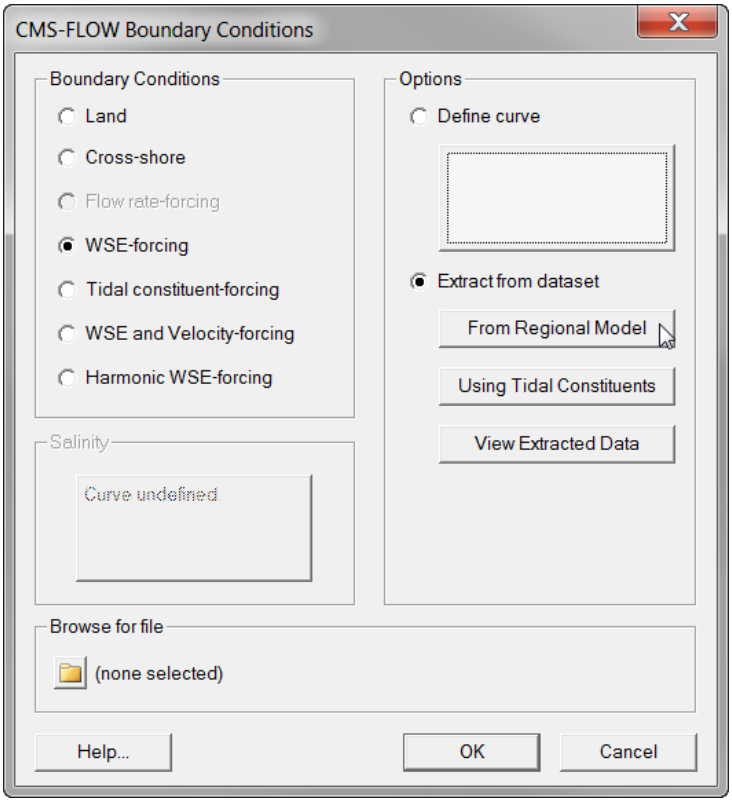


Figure 2-58. CMS-Flow Boundary Conditions window in SMS 11.1 with an WSE boundary condition.

5. Select *WSE-forcing* in the *Boundary Conditions* section on the left side of the window.
6. Select *Extract from dataset* in the *Options* section on the right side of the window.

7. Click on *using Tidal Constituents*. The Extract Boundary Conditions window will appear (see example figure below).
8. Select the water level forcing dataset and the start and end times for the extraction.
9. Select the *Extract* button in the *Extract Boundary Conditions* window. The window will close.

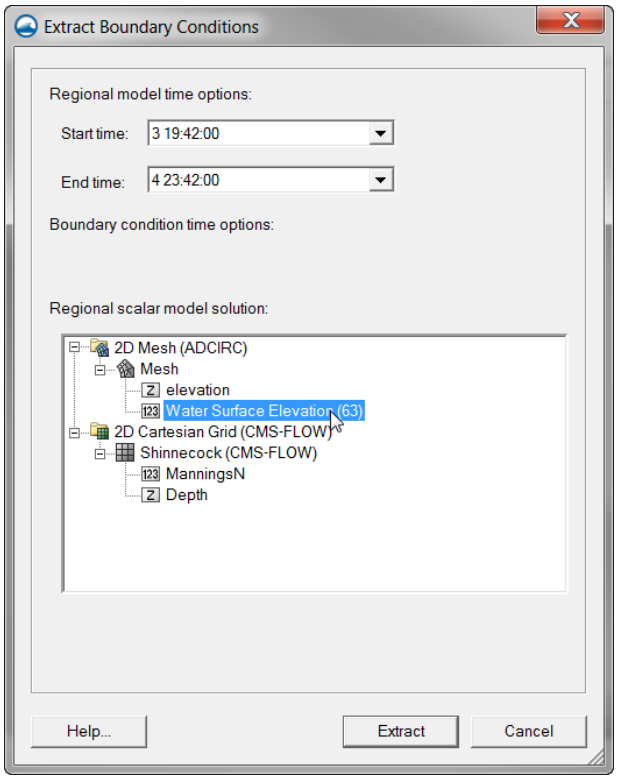


Figure 2-59. *Extract Boundary Conditions* window in SMS 11.1.

10. In the CMS-Flow Boundary Conditions window, select the *OK* button.
11. Save the *SMS project File* (*.sms) or *CMS-Flow Simulation File* (*.cmcards).

Extracted Water Level and Velocity Time Series Boundary

The water level and velocity at a boundary can be specified as:

1. Cell-specific time-series curves (i.e. one time-series for each boundary cell). This boundary type is used when extracting boundary conditions in SMS.
2. Interpolated values from a parent grid simulation (nesting).
3. Cell-specific cosine series using interpolated tidal constituents from a tidal database.

Example 2-34. Multiple water level and velocity boundary specification using two cards.

```
MDRIVER_CELLSTRING   "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
VDRIVER_CELLSTRING   "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
```

Cross-shore Boundary Condition

In the implicit flow solver, a cross-shore boundary condition is applied by solving the 1-D cross-shore momentum equation including wave and wind forcing (Wu et al. 2011a, 2011b). Along a cross-shore boundary, it is assumed that a well-developed longshore current exists. Thus, the alongshore (*y*-direction) momentum equation can be reduced to

$$\frac{\partial}{\partial x}\left(\nu_t h \frac{\partial V_y}{\partial x}\right) = \frac{1}{\rho}\left(\tau_{sy} + \tau_{wy} - \tau_{by}\right) \tag{2-16}$$

where τ_{sy}, τ_{wy}, and τ_{by} are the surface, wave, and bottom stresses in the long-shore direction, respectively. The equation above is solved iteratively for the longshore current velocity. The cross-shore (*x*) component of the velocity is assigned a zero-gradient boundary condition.

The water level due to waves and wind at the cross-shore boundary can be determined by assuming a zero alongshore gradient of flow velocity and negligible cross-shore current velocity. For this case, the cross-shore momentum equation reduces to

$$\rho g h \frac{\partial \overline{\eta}}{\partial x} = \tau_{sx} + \tau_{wx} \tag{2-17}$$

where τ_{sx} and τ_{wx} are the wind and wave stresses in the cross-shore direction.

Tidal Databases

SMS Supported Tidal Databases

SMS allows the user to generate multiple water level time series from spatially variable tidal constituents. The tidal constituents are spatially interpolated from one of several available datasets:

1. NE Pacific Tidal Database (ADCIRC)
2. NW Atlantic Tidal Database (ADCIRC)
3. LeProvost Tidal Database

Step-by-Step Instructions for Setting Up the Tidal Databases in SMS.

1. Download the tidal database. The ADCIRC database can be downloaded from http://www.unc.edu/ims/ccats/tides/tides.htm and LeProvost from http://sms.aquaveo.com/leprovost.zip.
2. Unzip the files and save them to the SMS directory (preferably).
3. Set the path to the tidal database by opening SMS and clicking on the *Edit* menu and selecting *Preferences...* (see Figure 4-16).
4. Enter the correct location (path) of the tidal database.

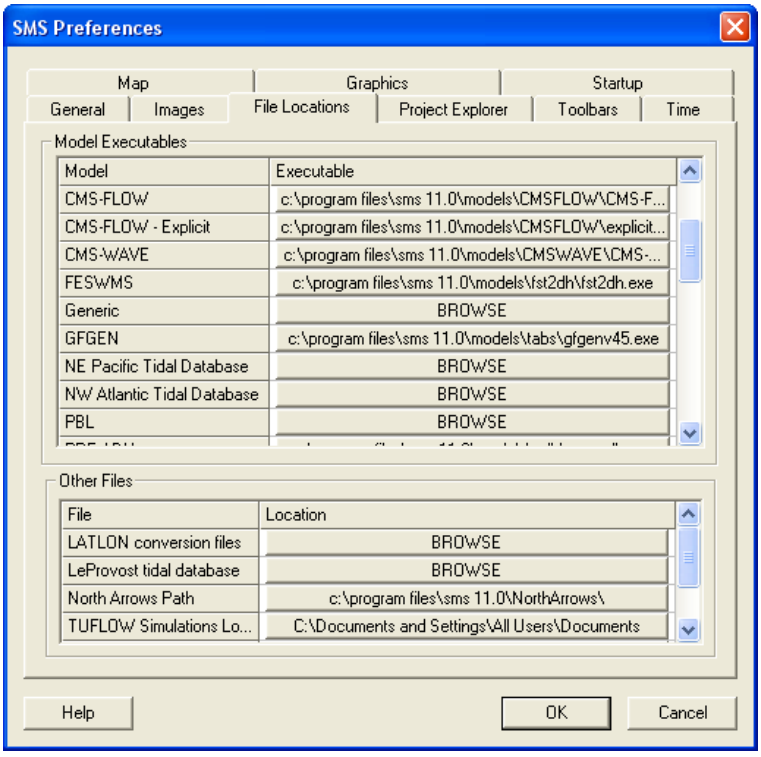


Figure 2-60. Setting the tidal database executable location in the SMS Preferences window.

When using the *SMS Boundary Extraction* utility, to extract water levels or water levels and current velocities from a boundary, the SMS creates the boundary time-series are these are written to the *Model Parameters File*.

Advanced Boundary Specification

Boundary Blocks

In CMS versions 4.0 and earlier both the cellstring information and the boundary information were stored together in the *Model Parameters File* (*_mp.h5), and therefore all of the information needed for each boundary was a path within the *Model Parameters File*. Although simple, this method makes it difficult for users to specify their own boundary time-series without going in SMS and resaving the whole CMS-Flow project. In addition, it makes it difficult to specify additional boundary parameters. For this reason, a new approach has been adopted in CMS V4.1 which is based on block structures. Block structures offer more flexibility for specifying input parameters, make the input file more readable, and reduce the complexity of the input cards by introducing context around them. The concept of a block structure is best illustrated with the following simple example:

Example 2-35. Boundary block specification.

```
BOUNDARY_BEGIN
   NAME   "My Offshore Boundary" !Optional
   !Other boundary information goes here
BOUNDARY_END
```

In the above example is incomplete since only a name is provided for boundary and no other boundary information is given such as the forcing conditions or the location of the boundary. These are covered in the following sections with detailed examples. A description of the boundary structure cards are provided in the table below.

Table 2-48. CMS-Flow cards related to the flux boundary condition.

Input	Format	Notes
Begins a Boundary Block	[begin=BOUNDARY_BEGIN, name=**BNDblock**]	Begins the block structure for a boundary condition
Boundary Block Name	[card=NAME, parent=**BNDblock**, optional=true] [name=**bndName**, type=*char*]	Specifies a boundary name. Optional.
Ends a Boundary Block	[end=BOUNDARY_BEGIN]	Begins the block structure for a boundary condition

Cellstrings

In CMS-Flow the spatial location of boundaries conditions is specified using cellstrings which are a list of cell ID's corresponding to the boundary cells in the CMS-Flow grid. Once cellstrings have been created, boundary conditions can be assigned to the cellstring. Cellstrings without assigned boundary conditions or boundaries without cellstrings are assumed to be closed.

Table 2-49. CMS-Flow card for specifying cellstrings.

Input	Format	Notes
Boundary Cellstrings	[card=CELLSTRING, parent=**BNDblock**] [arg=**bidFile**, type=*char*, default=**mpFile**] [arg=**bidPathID**, type=*char*, default=**none**]	Specifies the boundary file, and path/ID. The path is used for XMDF files and ID number for ASCII Boundary ID Files.

Example 2-36. A boundary block structure with the cellstring boundary information specified in the Model Parameters File (*_mp.h5).

```
BOUNDARY_BEGIN
   CELLSTRING   "Flow_mp.h5" "PROPERTIES/Model Params/Boundary_#1"
BOUNDARY_END
```

Example 2-37. A boundary block structure with the cellstring boundary information specified in an ASCII Boundary ID File (*.bid).

```
BOUNDARY_BEGIN
   NAME        "My Offshore Boundary" !Optional
   CELLSTRING   "Boundaries.bid"  1 ![file name] [cellstring ID]
BOUNDARY_END
```

The *Boundary ID File* (*.bid) is an ASCII file with all of the cell ID's listed for each boundary. This option allows users an alternate option to specifying boundaries which may be easier than editing the *Model Parameters File*. A simple example of a Boundary ID File is provided below.

Example 2-38. Boundary ID File.

```
2 !Number of strings
3 !Number cells in string 1
6
12
18
4 !Number of cells string 2
1 2 3 4
```

The lines with the number of cellstrings and the length of each cellstring must be on a separate line. However, the actual cellstring ID numbers may be on a single or multiple lines.

Notes:

- When working with advanced CMS-Flow features such as the boundary blocks, it is recommended to save a backup of the *CMS-Flow Model Control File* in case SMS removes or changes any cards specified in a text editor. When using many advanced cards, it is recommended to work in a text editor instead of SMS.
- Always check the *CMS-Flow Diagnostic File* (see Section Diagnostic File) and output screen to make sure that the input parameters and options are being set properly in the model.

Flux Boundary

Several advanced features for the flux boundary are available through the advanced cards. These features use a block structure format which allows greater flexibility in assigning boundary options. The flux boundary block and some of the basic boundary options are described in the table below.

Table 2-50. CMS-Flow cards related to the flux boundary condition.

Input	Format	Notes
Flux Boundary Block	`[begin=FLUX_BEGIN, name=Qblock]`	Begins the block structure for a flux boundary condition
Flux data units	`[card=(UNITS,FLUX_UNITS), parent=Qblock] [arg =Qunits, type=char, options=fluxunits, default='m^3/s/cell']`	Specifies the units of the input flux data.
Inflow direction	`[card=(DIRECTION, INFLOW_DIRECTION), parent=Qblock] [arg=QDir, type=float, optional=true, default=auto options=(-360<Qdir<=360 'deg')] [arg=QDirUnits, type=char, options=AngUnits, default='deg']`	Specifies in the inflow angle clockwise from North. The default value is inwards normal to the boundary.
Conveyance coefficient	`[card=(CONVEYANCE, CONVEYANCE_COEFFICIENT), parent=Qblock] [arg=Qconvey, type=real, range=(0.3<Qconvey<0.8), default=0.667]`	Specifies the conveyance coefficient used to distribute the total flux along the boundary.
Flux Boundary Block	`[end=FLUX_END, block=Qblock]`	Ends the block structure for a flux boundary condition

The boundary flux may be specified as a flux per boundary cell or as a total flux for the whole boundary. The flux along the boundary and is distributed using a conveyance approach. If the flux is specified per cell, then it is multiplied by the number of cells to compute a total flux and then distributed along the boundary using the conveyance approach. The inflow direction is assumed to be normal to the boundary if not specified. The boundary flux may be specified as a constant or time-series.

Notes:

- The inflow direction is only uses for inflow conditions, as the name indicates. If the flow is outward, the flow is allowed to exit the domain without modifying the flow direction.

Flux Value

For many cases, it is sufficient to specify a constant flux. The flux value is specified using the card described in the table below.

Table 2-51. CMS-Flow cards related to the flux boundary condition.

Input	Format	Notes
Flux value	[card=(VALUE,FLUX_VALUE)] [arg=**Qvalue**, type=*float*, optional=true] [units=**QvalueUnits**, type=*char*, options=**fluxunits**, default='m^3/s/cell']	Specifies the boundary flux value.

Below is an example in which a constant total flux is applied to a boundary. The units of the flux value are optional. The default units are m^3/s/cell.

Example 2-39. Constant boundary flux.

```
BOUNDARY_BEGIN
   CELLSTRING   "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
   FLUX_BEGIN
      VALUE       50.0 'm^3/s' !Constant water flux
   FLUX_END
BOUNDARY_END
```

Flux Time Series

The boundary flux may be specified using a time-series.

Table 2-52. CMS-Flow cards related to the flux boundary condition.

Input	Format	Notes
Flux time series file	[card=(CURVE,FLUX_CURVE), parent=**Qblock**] [arg=**Qfile**, type=*char*, default=**none**] [arg=**Qfath**, type=*char*, default=**none**]	Specifies the Flux Data File Name (including path if different from Card File) and Dataset Path (for XMDF files only). Optional. Assumed to be the same as the cellstring if not specified.

The simple example below is equivalent to using the card QDRIVER_CELLSTRING. It uses the flux curve specified in the *Model Parameters File*.

Example 2-40. Total flux time-series boundary.

```
BOUNDARY_BEGIN
   CELLSTRING    "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
   FLUX_BEGIN
      CURVE      "Flow_mp.h5" "PROPERTIES/Model Params/Boundary" !Optional
   FLUX_END
BOUNDARY_END
```

The example below illustrates the enhanced capabilities of the block structure format. In the example, a name is assigned to the boundary. The cellstring ID's are specified using a *Boundary ID File* (*.bid). The input total flux time series is specified in an XY Series file (*.xys). The units of the time series flux is specified as cubic meters per second. The inflow direction is assigned clockwise from true North. The conveyance coefficient determines how the total flux is distributed along the boundary. For the simple case provided below the water depths and Manning's coefficient are constant along the boundary, so the conveyance coefficient will not affect the results.

Example 2-41. Total flux time-series specification using an XY Series file (*.xys).

```
BOUNDARY_BEGIN
  NAME              "River"
  CELLSTRING        "Boundaries.bid"  1 ![file name] [cellstring ID]
  FLUX_BEGIN
    CURVE           "Flux.xys"  !Contains flux data
    UNITS           'm^3/s'     !Optional, default is m^3/s/cell
    DIRECTION       110.0 'deg' !Optional, -360<real<360 in degrees
    CONVEYANCE      0.5         !Optional, 0.5<real<1.0, default=0.667
  FLUX_END
BOUNDARY_END
```

The figure below shows an example of the inflow direction specified at 110 degrees.

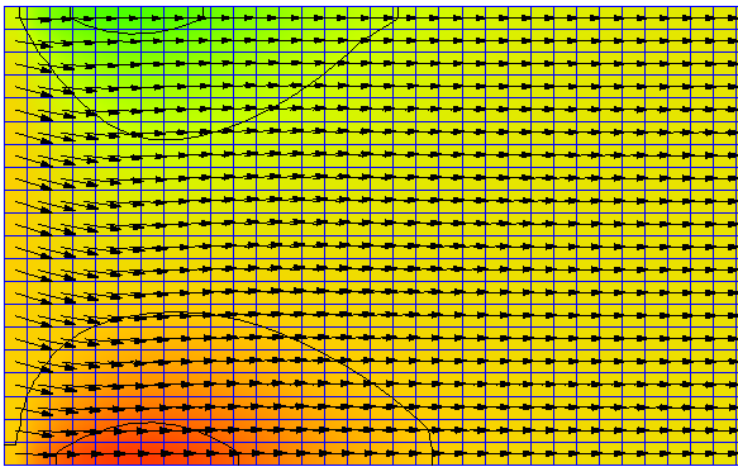


Figure 2-61. Example of a flux boundary specified at an oblique angle to the boundary. Contours indicate the current magnitude with warmer colors being larger current velocities.

It is also possible to use a Time Series Data or TSD file (*.tsd). The TSD file has the advantage that it uses a reference time so the input files is independent of the simulation starting time.

Example 2-42. Volume flux per cell time-series specification using a Time Series Data file (*.tsd).

```
BOUNDARY_BEGIN
  CELLSTRING        "Boundaries.bid"  1  ![file name] [cellstring ID]
  FLUX_BEGIN
    CURVE           "Flux.tsd"     !Contains flux data
    UNITS           'm^3/s/cell'   !Optional, default is m^3/s/cell
    DIRECTION       90.0 'deg'     !Normal to boundary default is 90
  FLUX_END
BOUNDARY_END
```

Water Level Boundary

Several advanced features for the water boundary are available through the advanced cards using the water level block structure format. The hierarchal block structures allow more flexibility in assigning input parameters, make the input file more readable and reduce the complexity of the input cards by introducing context around them. The syntax for the water level block structure is described in the table below.

Table 2-53. CMS-Flow cards related to the wse boundary condition block.

Input	Format	Notes
Begins a WSE Boundary Block	`[block=WSE_BEGIN, name=WSEblock]`	Begins the block structure for a flux boundary condition
Ends WSE Boundary Block	`[block=WSE_END, name=WSEblock]`	Ends the block structure for a flux boundary condition

The cards or input parameters which can be placed within the water level block depend on the boundary condition specification and is discussed in detail in the sections below. The boundary water levels may be specified as:

1. Single water level value
2. Multiple water level time series (one for each boundary cell)
3. Extracted water level time series from a parent simulation (nesting)
4. Tidal constituents (constant along boundary)
5. Harmonic constituents (constant along boundary)
6. Extracted water level constituents from a tidal database

The above options are described in detail in the following sections.

Single Water Level Value

The simplest water level boundary condition is a constant value. This water level boundary condition is useful for simulating steady-state problems. The input cards used for specifying a constant water level and boundary options are described in the table below.

Table 2-54. CMS-Flow cards related to the single water level time series boundary condition.

Input	Format	Notes
WSE value	[card=VALUE, parent=**WSEblock**] [name=**WSEval**, type=*float*, optional=true, default=0.0)]	Specifies constant WSE.
WSE gradients	[card=(WSE_GRADIENTS,GRADIENTS), parent=**WSEblock**] [name=**WSEgrad**, type=*float*, optional=true, default='0.0,0.0')]	Specifies WSE gradients which are superimposed on the boundary. Used to approximate regional circulation.
WSE correction or adjustment	[card=(ADJUSTMENT,CORRECTION), parent=**WSEblock**] [name=**WSEadjust**, type=*char*,options=(ON,OFF), default=ON]	Corrects the boundary water level for wind, atmospheric pressure, and wave forcing.

The following simple example illustrates how to specify a constant water level using the advanced block structure. This is the most simple BC which can be specified. No other options are specified and are left with default values.

Example 2-43. Constant WSE BC specification using a block structure.

```
BOUNDARY_BEGIN
   CELLSTRING   "Boundaries.bid"  1 ![file name] [cellstring ID]
   WSE_BEGIN
      VALUE       0.2 m   !constant water level, units optional
   WSE_END
BOUNDARY_END
```

In the second example, a spatially gradient is added to the water level boundary. In addition, the water level adjustment for waves, wind, and atmospheric pressure is turned off.

Example 2-44. Constant WSE BC specification using a block structure and additional options.

```
BOUNDARY_BEGIN
   NAME          "Constant Water Level"
   CELLSTRING    "Boundaries.bid"  1 ![file name] [cellstring ID]
   WSE_BEGIN
      VALUE           0.0 m   !constant water level, units optional
      GRADIENTS       1.E-5 0.0 !Optional, global coordinate system
      ADJUSTMENT      OFF      !Optional, turns on the forcing corrections
   WSE_END
BOUNDARY_END
```

Notes:

- If a water level gradient is applied, the gradient is centered at the middle cell of the cell string and not the center cell spatially.

Single Water Level Time Series

A single water level time-series is one of the most commonly boundary condition types used for coastal modeling.

Table 2-55. CMS-Flow cards related to the single water level time series boundary condition.

Input	Format	Notes
WSE time series file	[cards=(CURVE,WSE_CURVE), parent=**WSEblock**] [name=**WSEfile**, type=*char*, default=**none**] [name=**WSEpath**, type=*char*, default=**none**]	Specifies the WSE Data File Name (including path if different from Card File) and Dataset Path (for XMDF files only). Optional. Assumed to be the same as the cell-string if not specified.
WSE gradients	[card=(WSE_GRADIENTS,GRADIENTS), parent=**WSEblock**] [name=**WSEgrad**, type=*float*, optional=true, default='0.0,0.0')]	Specifies WSE gradients which are superimposed on the boundary. Used to approximate regional circulation.
WSE Temporal Interpolation Order	[card=TIME_INTERP_ORDER] [name=**WSEnti**, type=*int*, range=(1<**WSEnti**<3),default=2]	Order of Lagrangian polynomial used to interpolate time-series.
WSE Temporal Smoothing Iterations	[card=TIME_SMOOTH_ITER] [name=**WSEnsi**, type=*int*, range=(0<**WSEnsi**<10), default=2]	Temporal smoothing iterations. Smoothing is done by a central moving average.
WSE Temporal Smoothing Window Width	[card=TIME_SMOOTH_WIDTH] [name=**WSEnsw**, type=*int*, range=(0<**WSEnsw**<10), default=0]	Temporal smoothing window width. Should be an odd number Smoothing is done by a central moving average.

Example 2-45. Single WSE time-series BC specification using a block structure.

```
#HDRIVER_CELLSTRING "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
BOUNDARY_BEGIN
   CELLSTRING  "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
   WSE_BEGIN
      CURVE     "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
   WSE_END
BOUNDARY_END
```

Example 2-46. Single WSE time-series BC specification using a block structure and advanced options.

```
BOUNDARY_BEGIN
   NAME          "My offshore boundary"
   CELLSTRING    "Boundaries.bid"  1
   WSE_BEGIN
      CURVE           "WSE.xys"
      OFFSET          0.1 m   !Optional
      GRADIENTS       1.E-5 0.0 !Optional, global coordinate system
      TIME_INTERP_ORDER  2 !Temporal interpolation
      TIME_SMOOTH_ITER   1 !Temporal smoothing iterations
      TIME_SMOOTH_WIDTH  3 !Temporal smoothing window width
   WSE_END
BOUNDARY_END
```

Example 2-47. Constant WSE BC specification using a block structure.

```
BOUNDARY_BEGIN
   CELLSTRING    "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
   WSE_BEGIN
      CURVE       "WSE.tsd"
      OFFSET      0.1 m     !Optional, positive is upwards
      GRADIENTS   1.E-5 0.0 !Optional, global coord system, nondimensional
      ADJUSTMENT ON         !Optional, turns on the forcing corrections
   WSE_END
BOUNDARY_END
```

Multiple Water Level Time Series Boundary

In this boundary condition type, an individual water level time-series is specified for each boundary cell. The SMS 11.1 interface allows the user to extract these time-series from a larger CMS or ADCIRC simulation or to synthesize them from interpolated tidal constituents from a tidal database. Once these time-series are generated, the user can then go into the CMS-Flow Model Control File (*.cmcards file), and use the advanced block structure format to be able to specify additional input options. Specifically, the block structure allows the user to choose a spatial interpolation order, and spatial smoothing along the cellstring. The temporal interpolation is calculated using a piece-wise Lagrangian polynomial of order 1 to 3. The first order polynomial reduces to linear interpolation. The higher order interpolation methods improve the smoothness of the interpolation but can lead to oscillations if the parent simulation is noisy or the solution file output times are very large. This is why the interpolation order is limited to 3. Temporal and spatial smoothing is applied (optional) as central moving averages with a user-specified window width and number of iterations. The spatial smoothing is applied first and then the temporal smoothing to the input boundary water levels. The smoothed water levels are then interpolated to the model time step using the piece-wise Lagrangian interpolation. The boundary water levels may be user-specified in a Time Series Data (TSD) file (*.tsd) or extracted in SMS from a larger simulation or tidal constituent database. Each option is covered in the sections below with examples. The option is also available to output the extracted water levels to a TSD file. The cards used to specify the multiple water level time-series boundary are described in the table below.

Table 2-56. CMS-Flow cards related to the multiple water level time series boundary condition.

Input	Format	Notes
WSE Input Time series file	[card=(WSE_DATA,DATASET,DATA) parent=**WSEblock**] [name=**WSEfile**, type=*char*, default=**none**] [name=**WSEpath**, type=*char*, parent=**WSEblock**, default=**none**]	Specifies the WSE Data File Name (including path if different from Card File) and Dataset Path (for XMDF files only). Optional. Assumed to be the same as the cellstring if not specified. The file must contain a time-series for each boundary cell.
WSE Temporal Interpolation Order	[card=TIME_INTERP_ORDER, parent=**WSEblock**] [name=**WSEnti**, type=*int*, range=(1<**WSEnti**<3), default=2]	Order of Lagrangian polynomial used to interpolate time-series.
WSE Temporal Smoothing Iterations	[card=TIME_SMOOTH_ITER, parent=**WSEblock**] [name=**WSEnsi**, type=*int*, range=(0<**WSEnsi**<10), default=2]	Temporal smoothing iterations. Smoothing is done by a central moving average.
WSE Temporal Smoothing Window Width	[card=TIME_SMOOTH_WIDTH, parent=**WSEblock**] [name=**WSEnsw**, type=*int*, range=(0<**WSEnsw**<10), default=0]	Temporal smoothing window width. Should be an odd number Smoothing is done by a central moving average.
WSE Spatial Smoothing Iterations	[card=SPACE_SMOOTH_ITER, parent=**WSEblock**] [name=**WSEnsi**, type=*int*, range=(0<**WSEnsi**<10), default=2]	Spatial smoothing iterations. Smoothing is done by a central moving average.
WSE Spatial Smoothing Window Width	[card=SPACE_SMOOTH_WIDTH, parent=**WSEblock**] [name=**WSEnsw**, type=*int*, range=(0<**WSEnsw**<10), default=0]	Spatial smoothing window width. Should be an odd number Smoothing is done by a central moving average.

Example: SMS-generated Multiple Water Levels

In the example below the SMS interface is used to create a multiple water level time-series boundary and the CMS project files are saved. The CMS-Flow card file is then modified by commenting out the SMS generated card MULTI_HDRIVER_CELLSTRING, and placing the card information in the user-specified boundary block. It is noted that both the cellstring information and the extracted water levels are contained within the same path of the CMS-Flow Model Parameters File (*_mp.h5). This is the reason why both the CELLSTRING and DATASET cards contain the same file name and path. The temporal interpolation order is set to 2, and two iterations of simple moving average filter are applied along the cellstring with a window width of 3.

Example 2-48. Multiple water levels time series boundary specification using a block structure.

```
#MULTI_HDRIVER_CELLSTRING "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
BOUNDARY_BEGIN
   NAME          "Extracted Offshore BC"
   CELLSTRING    "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
   WSE_BEGIN
      DATASET    "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
      OFFSET     0.1 m     !Optional, default = 0.0 m
      TIME_INTERP_ORDER    2 !Optional, Temporal interpolation
      TIME_SMOOTH_ITER     1 !Optional, Temporal smoothing iterations
      TIME_SMOOTH_WIDTH    3 !Optional, Temporal smoothing window width
      SPACE_SMOOTH_ITER    2 !Optional, Spatial smoothing iterations
      SPACE_SMOOTH_WIDTH   3 !Optional, Spatial smoothing window width
   WSE_END
BOUNDARY_END
```

Example: User-specified Multiple Water Levels

In the example below, a multiple water level boundary condition is specified using all ASCII input files. The cellstring is specified using a BID file, and the boundary water levels using a TSD file.

Example 2-49. Multiple water levels time series boundary specification using a block structure.

```
BOUNDARY_BEGIN
   NAME          "All ASCII Offshore BC"
   CELLSTRING    "Boundaries.bid"  1 ![file name] [cellstring ID]
   WSE_BEGIN
      DATASET    "Offshore_wse.tsd" !Must be TSD file
   WSE_END
BOUNDARY_END
```

The TSD file containing the boundary water levels should contain a time series for each boundary cell and covering the whole simulation time period. The format of the TSD file is illustrated in the two examples below.

Example 2-50. Excerpt of the Offshore_wse.tsd file used to specify multiple water level boundary condition in the above example.

```
TIME_SERIES
"Boundary WSE"  "Unassigned"     7      100  "01/01/2011 00:00:00"
      0.000     0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
   1800.000     0.0020    0.0021    0.0022    0.0023    0.0024    0.0025
   3600.000     0.0073    0.0074    0.0075    0.0076    0.0077    0.0078
   5400.000     0.0842    0.0843    0.0844    0.0845    0.0846    0.0847
   7200.000     0.1731    0.1732    0.1733    0.1734    0.1735    0.1736
   9000.000     0.2546    0.2547    0.2548    0.2549    0.2550    0.2551
   ...
```

Notes:

- If the number of time steps is set to zero (forth entry in second line of TSD file), CMS will automatically determine the number from the input file. Therefore, the parameter may be considered optional when using as an input for CMS.

Extracted Water Levels from a Parent Simulation

CMS supports automated one-way nesting within parent CMS and ADCIRC models. Additional support for other models is currently in development. When using this feature, the CMS automatically extracts the boundary condition information from the parent model solution. The user only has to specify the boundary cellstring and the location and name of the parent model simulation files. The user may force the CMS with either water levels or water levels and velocities extracted from the parent simulation.

Table 2-57. CMS-Flow cards related to the single tidal current velocity boundary condition.

Input	Format	Notes
WSE Temporal Interpolation Order	[card=TIME_INTERP_ORDER, parent=**WSEblock**] [name=**WSEnti**, type=*int*, range=(1<**WSEnti**<3), default=2]	Order of Lagrangian polynomial used to interpolate time-series.
WSE Temporal Smoothing Iterations	[card=TIME_SMOOTH_ITER, parent=**WSEblock**] [name=**WSEnsi**, type=*int*, range=(0<**WSEnsi**<10), default=2]	Temporal smoothing iterations. Smoothing is done by a simple moving average.
WSE Temporal Smoothing Window Width	[card=TIME_SMOOTH_WIDTH, parent=**WSEblock**] [name=**WSEnsw**, type=*int*, range=(0<**WSEnsw**<10), default=0]	Temporal smoothing window width. Should be an odd number Smoothing is done by a simple moving average.
WSE Spatial Smoothing Iterations	[card=SPACE_SMOOTH_ITER, parent=**WSEblock**] [name=**WSEnsi**, type=*int*, range=(0<**WSEnsi**<10), default=2]	Spatial smoothing iterations. Smoothing is done by a simple moving average.
WSE Spatial Smoothing Window Width	[card=SPACE_SMOOTH_WIDTH, parent=**WSEblock**] [name=**WSEnsw**, type=*int*, range=(0<**WSEnsw**<10), default=0]	Spatial smoothing window width. Should be an odd number Smoothing is done by a simple moving average.
WSE Output File	[cards=(OUTPUT_FILE, WSE_OUTPUT_FILE), parent=**WSEblock**] [name=**WSEfile**, type=*char*, default=**none**]	Specifies the WSE output file. Must be a TSD file. Must include relative or absolute path if different from the CMS-Flow project files.

Note:

- Simulation results from other models may be used to force CMS by converting them into the ADCIRC format.

Table 2-58. CMS-Flow cards related to the wse boundary condition block.

Input	Format	Notes
Begins a WSE Parent Block	[begin=PARENT_BEGIN, name=**Parblock**]	Begins the block structure for specifying the parent simulation parameters
Ends a WSE Parent Block	[end=PARENT_END, name=**Parblock**]	Ends the block structure for specifying the parent simulation parameters
Parent Control File	[card=CONTROL_FILE, parent=**Parblock**] [name=**ParCTLfile**, type=*char*, default=none]	Name of parent simulation control file. Optional for parent ADCIRC simulations.
Parent Grid File	[card=GRID_FILE, parent=**Parblock**] [name=**ParCTLfile**, type=*char*, default=none]	Name of parent simulation grid file. Optional for parent CMS simulations.
Simulation Starting Date and Time	[card=STARTING_DATE_TIME] [name=**StartDate0**, type=*char*, default=none, example="2004-12-24"] [name=**StartTime0**, type=*char*, default=none, example="23:34:12"] [name=**StartTimeZone**, type=*char*, default=none, example='UTC' , otpional=*true*,]	Specifies the simulation starting calendar date and time. Double quotes are optional

Example: Nesting a CMS simulation within an ADCIRC simulation

In the example below, a child CMS simulation is nested within a larger ADCIRC simulation and forced with water surface elevations (WSE). The boundary information begins with user-specified name for the boundary and the cellstring information for the CMS (child) grid. The WSE block contains a WSE offset used in this case to convert between vertical coordinate systems. In addition a small amount of spatial smoothing is applied as a 3 element moving mean along the cellstring. The parent simulation block specifies all of the options of the ADCIRC simulation including the names of the grid, WSE solution files, starting date and time, and horizontal projection. The parent simulation files should include the absolute or relative path if not in the same directory as the CMS (child) simulation. The ADCIRC grid may be specified with the *.14 or *.grd extensions. The default name of the ADCIRC WSE solution file is fort.63 but the fort63.h5 name may also be used. Since the ADCIRC simulation files do not contain information on the starting date or horizontal projection, this information needs to be provided in the model input. Typically, the ADCIRC simulation will be run in a geographic coordinate system.

The simulation domain for both the ADCIRC and nested CMS computational grids is presented in the figure below. In a simulation that the one below, the larger ADCIRC simulation would be forced with tidal constituents while the nested grid is forced with extracted water levels from the ADCIRC simulation.

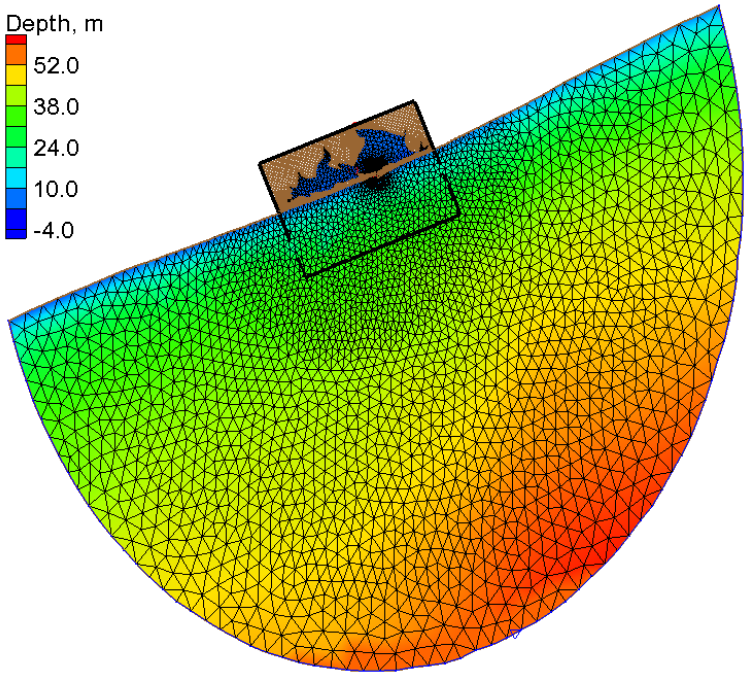


Figure 2-62. Nested CMS-Flow grid within a larger ADCIRC mesh. Note: The ADCIRC mesh was run in geographic coordinates, but was converted to the same horizontal projection as the CMS grid for display.

Note:

- ADCIRC is typically run for larger regional grids in a geographic co-ordinate system. If the horizontal projection is not specified then it is assumed to be NAD83, Geographic with units of degrees.
- When nesting a CMS simulation within an ADCIRC simulation, it is important that both simulations use the same horizontal datums. Otherwise, the model will not be able to internally convert between the datums.
- It is important to keep the model forcing for both the parent and child grids consistent. For example if the child grid is forced with winds, then the parent simulation should also be forced with the same winds.
- Because of differences in resolution, the grid coverages (extents) and bathymetries of the parent and child simulations will never be identical but they should be similar. If they are very different it will lead to boundary problems because the parent model solution will not be consistent with the child model solution.

Tips:

- It is recommended to display both the parent and child computational grids in SMS to check that the horizontal projections are correct and that the grid coverages and bathymetries are simular.

The input cards used to nest CMS within an ADIRC simulation are presented in the example below.

Example 2-51. Extracting water levels from parent ADCIRC simulation.

```
BOUNDARY_BEGIN
   NAME           "Nested Boundary"
   CELLSTRING     "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
   WSE_BEGIN !All settings for wse at the boundary set here
      OFFSET              0.1 m  !optional, positive is upwards
      TIME_INTERP_ORDER   2     !optional, temporal interpolation
      SPACE_SMOOTH_ITER   1     !optional, spatial smoothing iterations
      SPACE_SMOOTH_WIDTH  3     !optional, spatial smoothing window width
   WSE_END
   PARENT_BEGIN !All settings for parent simulation set here
      GRID_FILE           "..\ADCIRC_Regional\fort.14"
      WSE_SOL_FILE        "..\ADCIRC_Regional\fort.63" !Optional
      STARTING_DATE_TIME  2012-12-01 00:00:00 UTC
      HORIZ_PROJ_BEGIN !Optional, assumed if not specified
         DATUM        NAD83       !NAD27 | NAD83 | LOCAL
         SYSTEM       GEOGRAPHIC  !UTM | STATE_PLANE | GEOGRAPHIC | LOCAL
         UNITS        DEGREES      !METERS | FEET | DEGREES | etc.
      HORIZ_PROJ_END
   PARENT_END
BOUNDARY_END
```

An example comparison of the ADCIRC (parent) and CMS (child) water level solutions is presented in the figure below. The water levels for both simulations are quite close. The differences are mainly due differences in the computational grids.

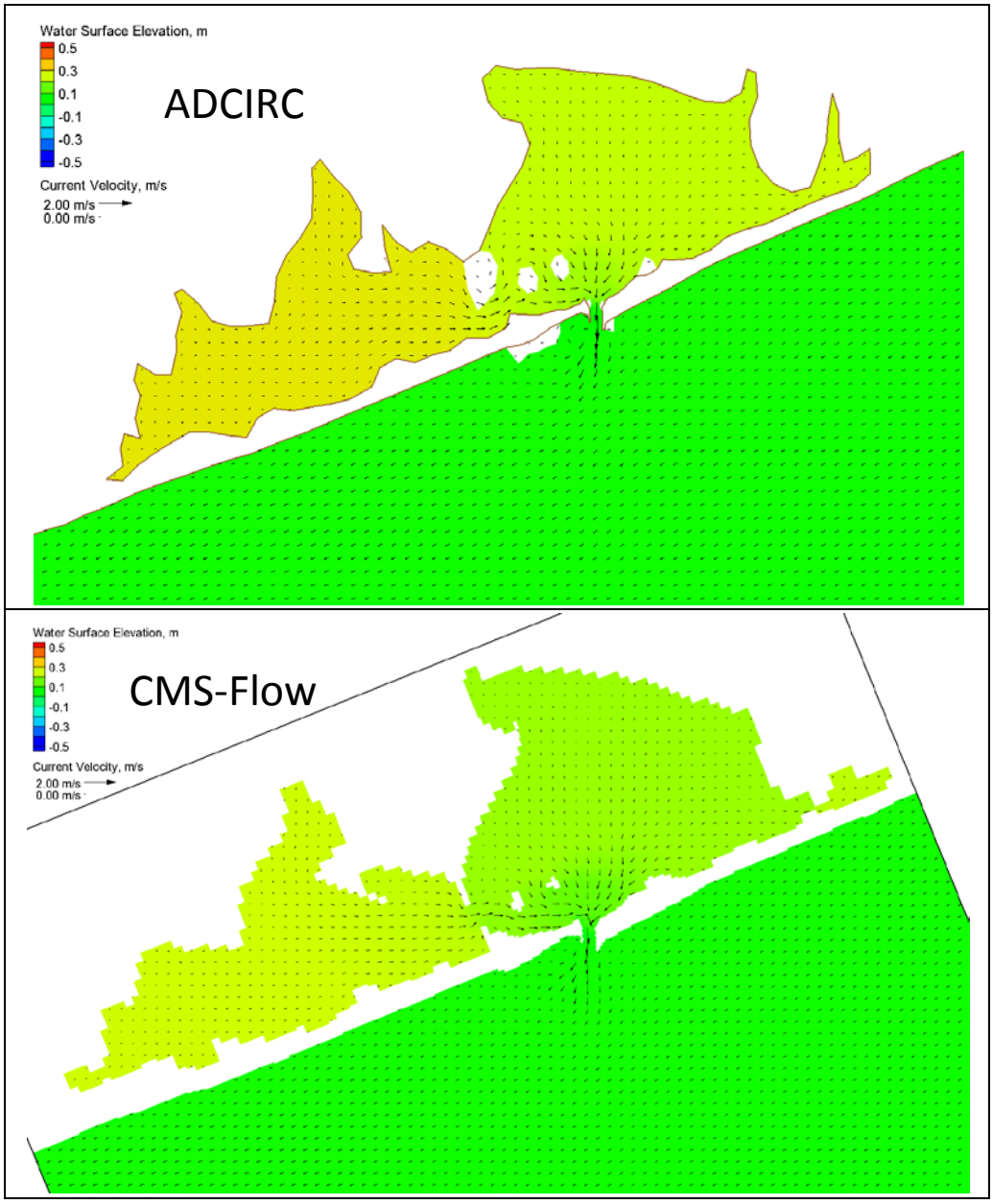


Figure 2-63. Example snapshots of water levels and current velocities for the ADCIRC (top) and CMS-Flow (bottom) simulations for the same time.

Example: Nesting a CMS simulation within a CMS simulation

If a parent CMS simulation is used, all of the necessary information including horizontal projection, grid file name, and water level solution file are extracted from the CMS Card File.

Example 2-52. Extracting water levels from parent CMS simulation.

```
BOUNDARY_BEGIN
   NAME              "Nested CMS Boundary"
   CELLSTRING        "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
   WSE_BEGIN
      WSE_OFFSET          0.1 m  !optional, positive is upwards
      TIME_INTERP_ORDER    2     !optional, temporal interpolation
      SPACE_SMOOTH_ITER    1     !optional, spatial smoothing iterations
      SPACE_SMOOTH_WIDTH   3     !optional, spatial smoothing window width
   WSE_END
   PARENT_BEGIN
      CONTROL_FILE     "..\CMS_Regional\Flow_Reg.cmcards"
      !All other settings are obtained from the control file
   PARENT_END
NESTED_WSE_BOUNDARY_END
```

Because the CMS-Flow Control (Card) File contains all of the information regarding the horizontal projection, simulation time period, and output files, nothing else needs to be specified except the parent CMS-Flow simulation Card File.

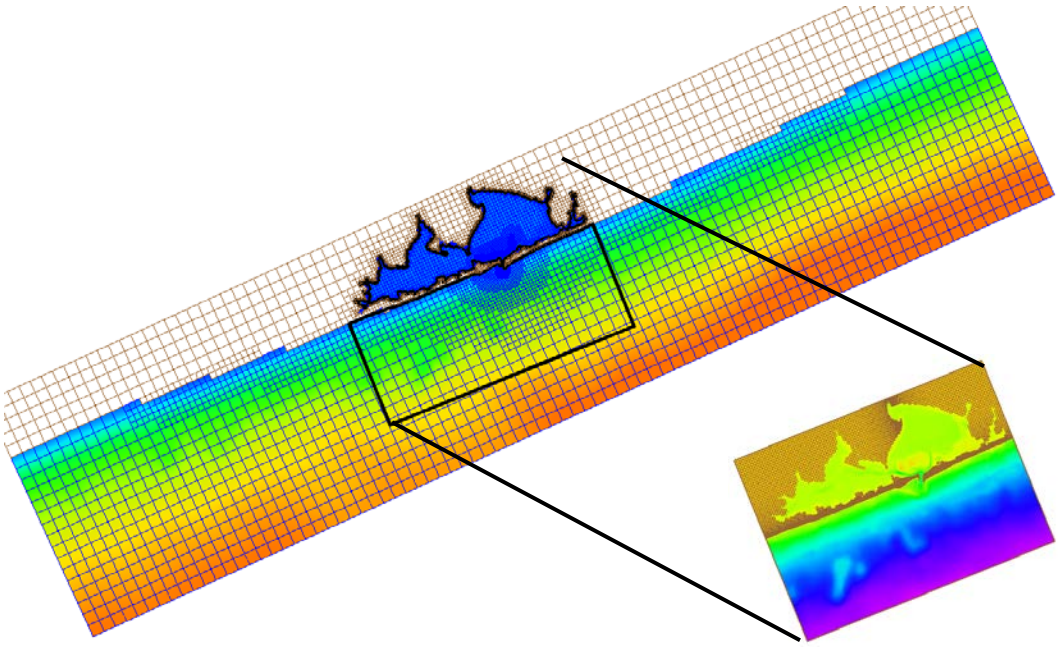


Figure 2-64. Computationa grids showing bathymetry for parent and child CMS-Flow grids.

Tidal Boundary

In the case of a tidal boundary condition, the external water level is calculated as

$$\overline{\eta}_E(t) = \sum f_i A_i \cos\left(\omega_i t + V_i^0 + \hat{u}_i - \kappa_i\right) \quad (2\text{-}18)$$

where

i = subscript indicating a tidal constituent

A_i = mean amplitude [m]

f_i = node (nodal) factor [-]

ω_i = frequency [deg/hr]

t = elapsed time from midnight of the starting year [hrs]

$V_i^0 + \hat{u}_i$ = equilibrium phase [deg]

κ_i = phase lag or epoch [deg]

The external water level is applied equally at all boundary cells unless an inflow

The tidal constituent information is specified using a Tidal Constituent Block and is described in the table below.

Table 2-59. CMS-Flow Cards related to the Tidal Boundary specification.

Input	Format	Notes
Begins a Tidal Constituents Boundary Block	[begin=TIDAL_CONSTITUENTS_BEGIN, name=**TCblock**]	Begins the block structure for a flux boundary condition
Tidal constituent card time series file	[card=TIDAL_CONSTITUENT_(**cname**), parent=**TCblock**] [name=**WSEfile**, type=*char*, default=**none**] [name=**WSEpath**, type=*char*, parent=**WSEblock**, default=**none**]	Specifies the WSE Data File Name (including path if different from Card File) and Dataset Path (for XMDF files only). Optional. Assumed to be the same as the cell-string if not specified. The file must contain a time-series for each boundary cell.
Ends Tidal Constituents Boundary Block	[end=TIDAL_CONSTITUENTS_END, name=**TCblock**]	Ends the block structure for a flux boundary condition

The first example for a tidal boundary condition applies two tidal constituents. The constituents are specified by their standard names as listed in Table 2-43, the amplitude in meters, and the phase in degrees. In the example below, the water level block in not included. Since a tidal forcing boundary condition is applied, it implies a water level boundary condition, and all of the default settings are applied to the water level boundary condition.

Example 2-53. Tidal boundary condition specification using the block structure.

```
BOUNDARY_BEGIN
   CELLSTRING        "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
   TIDAL_BEGIN
      CONSTITUENT     M2 0.75 211.0  !Name, Amplitude[m], Phase[deg]
      CONSTITUENT     S2 0.18 231.0  !Name, Amplitude[m], Phase[deg]
   TIDAL_END
BOUNDARY_END
```

In second example below, more advanced options are specified. The name is given to the boundary. A water level offset is specified of 0.1 m width respect to the still water level (SWL). The offset is useful for converting from the local vertical datum to the SWL. The direction of the forcing is also specified using the DIRECTION card and produces a phase lag along the tidal boundary condition so that the tidal wave propagates in the direction specified.

Example 2-54. Tidal boundary condition specification using the block structure

```
BOUNDARY_BEGIN
   NAME              "Offshore tidal boundary"
   CELLSTRING        "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
   WSE_BEGIN
      OFFSET          0.1 m     !Optional, default 0.0
   WSE_END
   TIDAL_BEGIN
      DIRECTION       245.33 deg  !Optional, default is normal to bnd
      CONSTITUENT     M2 0.75 211.0  !Name Amplitude Phase
      CONSTITUENT     S2 0.18 231.0  !Name Amplitude Phase
      CONSTITUENT     K1 0.15 185.0  !Name Amplitude Phase
      CONSTITUENT     O1 0.24 191.0  !Name Amplitude Phase
   TIDAL_END
BOUNDARY_END
```

The figure below shows an example of a tidal boundary specified for Shinnecock Inlet NY, with the direction specified parallel to the shoreline for illustration purposes.

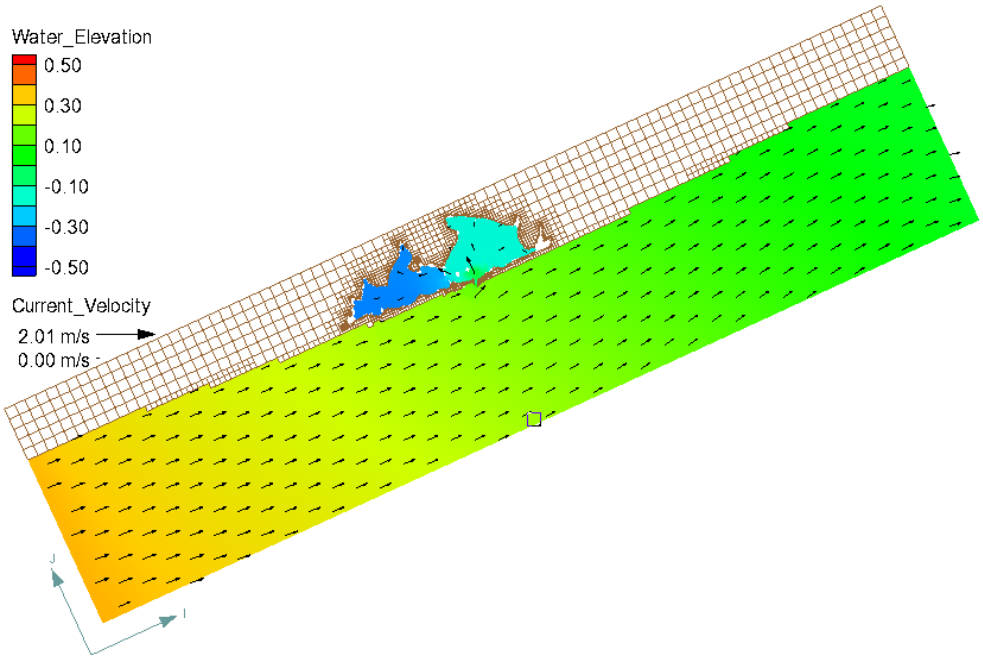


Figure 2-65. Example of a tidal boundary condition specified with an incident tidal wave direction parallel to the shore contours (not realistic).

Notes:

- The water level block may be omitted, since it is understood that a tidal forcing boundary condition applies a water level boundary condition.

Harmonic Boundary

The external water level at a harmonic boundary is calculated as

$$\bar{\eta}_E(t) = \sum A_i \cos\left(\omega_i t - \kappa_i\right) \tag{2-19}$$

where

i = subscript indicating a harmonic constituent

A_i = mean amplitude [m]

ω_i = frequency [deg/hr]

t = elapsed time from the start of the simulation [hrs]

κ_i = phase lag or epoch [deg]

The major difference between a tidal and a harmonic boundary condition is that the tidal boundary condition applies corrections to both the amplitudes and phases for each constituent which are a function of time while the harmonic boundary does not apply any corrections. Another difference is that in the case of the harmonic boundary, the speed (i.e. frequency) for each constituent must be specified rather than specifying the name of the constituent.

In the first harmonic boundary condition example, only two constituents are specified.

Example 2-55. Harmonic boundary condition specification for CMS V4.1 and later.

```
BOUNDARY_BEGIN
   CELLSTRING       "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
   HARMONIC_BEGIN
      !Constituent   Speed[deg/hr], Amplitude[m], Phase[deg]
      CONSTITUENT    28.9841042 0.75 211.0 !M2
      CONSTITUENT    13.9430356 0.24 191.0 !O1
   HARMONIC_END
BOUNDARY_END
```

Example 2-56. Harmonic boundary condition specification for CMS V4.1 and later.

```
BOUNDARY_BEGIN
   NAME             "Offshore harmonic boundary"
   CELLSTRING       "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
   WSE_BEGIN
      OFFSET         0.1 m     !Optional, default 0.0
   WSE_END
   HARMONIC_BEGIN
      DIRECTION      245.33 deg  !Optional, default is normal to bnd
      !Constituent   Speed[deg/hr], Amplitude[m], Phase[deg]
      CONSTITUENT    28.9841042 0.75 211.0 !M2
      CONSTITUENT    30.0000000 0.18 231.0 !S2
      CONSTITUENT    15.0410686 0.15 185.0 !K1
      CONSTITUENT    13.9430356 0.24 191.0 !01
   HARMONIC_END
BOUNDARY_END
```

Notes:

- The water level block may be omitted, since it is understood that a harmonic boundary condition applies a water level boundary condition.

Extracted Water Level Tidal Constituents from a Database

CMS supports extracting water level tidal constituent information directly from regional and world tidal databases. The constituent information is then used internally to generate boundary information along cellstrings according to

$$\bar{\eta}_E(\vec{x},t) = \sum f_i A_i \cos\left(\omega_i t + V_i^0 + \hat{u}_i - \kappa_i\right) \tag{2-20}$$

where

i = subscript indicating a tidal constituent

$\bar{\eta}_E(\vec{x},t)$ = external boundary water surface elevation

$A_i(\vec{x})$ = mean amplitude [m]

f_i = node (nodal) factor [-]

ω_i = frequency [deg/hr]

t = elapsed time from midnight of the starting year [hrs]

$V_i^0 + \hat{u}_i$ = equilibrium phase [deg]

$\kappa_i(\vec{x})$ = phase lag or epoch [deg]

The nodal factor is a time-varying correction to the mean amplitude. The equilibrium phase has a uniform component V_i^0 and a relatively smaller periodic component. The zero-superscript of V_i^0 indicates that the constituent phase is at time zero.

All tidal databases have constituent information for water levels and some also have constituent information for current velocities. The table below lists the tidal databases currently supported in SMS and some basic information.

Table 2-60. Tidal Databases supported by CMS-Flow.

Name	Description	Coverage and Grid	Water Level Constituents	Current Velocity Constituents
EC2001	US East Coast Tidal Database calculated with the Finite Element model ADCIRC (Mukai et al. 2002).	Western North Atlantic, Gulf of Mexico and Caribbean Sea. Unstructured triangular mesh with 254565 nodes and variable resolution ~1 – 25 km	STEADY, K1, O1, Q1, M2, S2, N2, K2, M4, M6	STEADY, K1, O1, Q1, M2, S2, N2, K2, M4, M6
ENPAC2003	US West Coast Tidal Database calculated with the Finite Element model ADCIRC (Spargo et al. 2004).	Eastern Pacific Ocean from Alaska to Central Mexico Coast. Unstructured triangular mesh with 272913 nodes and variable resolution of ~5 – 70 km	STEADY, K1, O1, P1, Q1, M2, S2, N2, K2, M4, M6	STEADY, K1, O1, P1, Q1, M2, S2, N2, K2, M4, M6
LEPROVOST	Finite Element Solution based global tidal database	Interpolated rectangular grid from an unstructured triangular mesh. Grid resolution 1/2 x 1/2 deg	S2, Q1, O1, N2, M2, MU2, K2, L2, T2, K1, 2N2	
FES952	Finite Element Solution based global tidal database based purely on hydrodynamic data (Le Provost et al. 1994)	Interpolated rectangular grid from an unstructured triangular mesh. Grid resolution 1/2 x 1/2 deg	M2, S2, N2, K2, 2N2, K1, Q1, O1, P1	
FES2004	Finite Element Solution based global tidal database. Includes assimilation of mareographic and satellite data (Lyard et al. 2006).	Interpolated rectangular grid from an unstructured triangular mesh. Grid resolution 1/8 x 1/8 deg	M2, S2, N2, K2, 2N2, K1, Q1, O1, P1	

The Le Provost tidal database can be downloaded from http://sms.aquaveo.com/leprovost.zip. The ADCIRC tidal databases can be downloaded from http://www.unc.edu/ims/ccats/tides/tides.htm. The FES2004 tidal database can be downloaded from http://www.legos.obs-mip.fr/en/share/soa/cgi/getarc/v0.0/index.pl.cgi?contexte=SOA&donnees=maree&produit=modele_fes.

Notes:

- The accuracies of the nonlinearly generated constituents STEADY, M4, and M6 in the ADCIRC tidal databases have not been verified and should therefore be used with caution.

The CMS-Flow cards used to specify the tidal database information are described in the table below.

Table 2-61. CMS-Flow cards related to the tidal database boundary block.

Input	Format	Notes
Begins a Tidal Database Boundary Block	[begin=TIDAL_DATABASE_BEGIN, name=**TCblock**]	Begins a tidal database block.
Tidal Database Name	[card=(DATABASE_NAME,NAME), parent=**TDBblock**] [name=**TDBname**, type=*char*, default=**none**]	Specifies the name of the tidal database.
Tidal Database Path	[card=(DATABASE_PATH,PATH), parent=**TDBblock**] [name=**TDBname**, type=*char*, default=**none**]	Ends the block structure for a flux boundary condition
Ends a Tidal Database Boundary Block	[begin=TIDAL_DATABASE_END, name=**TCblock**]	Ends a tidal database block.

The example below only 4 tidal constituents are selected from the tidal database. In addition an WSE offset is specified to correct for a tidal datum shift.

Example 2-57. Tidal database water level boundary specification

```
BOUNDARY_BEGIN
   NAME            "Tidal Database Boundary"
   CELLSTRING      "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
   WSE_BEGIN
      OFFSET       0.1 m    !Optional, default = 0.0 m, default units = m
   WSE_END
   TIDAL_DATABASE_BEGIN
      NAME         EC2001 !EC001 | ENPAC2003 | FES95 | FES2004 | LEPROVOST
      PATH         "E:\Tidal_Databases\ec2001\"
      CONSTITUENTS 4 M2 S2 K1 O1   ![num] [cname(1:num)], optional
   TIDAL_DATABASE_END
BOUNDARY_END
```

Example: CMS simulation of Southern California using Tidal Databases

The CMS model is setup for southern California coast. The computation grid is presented in the figure below.

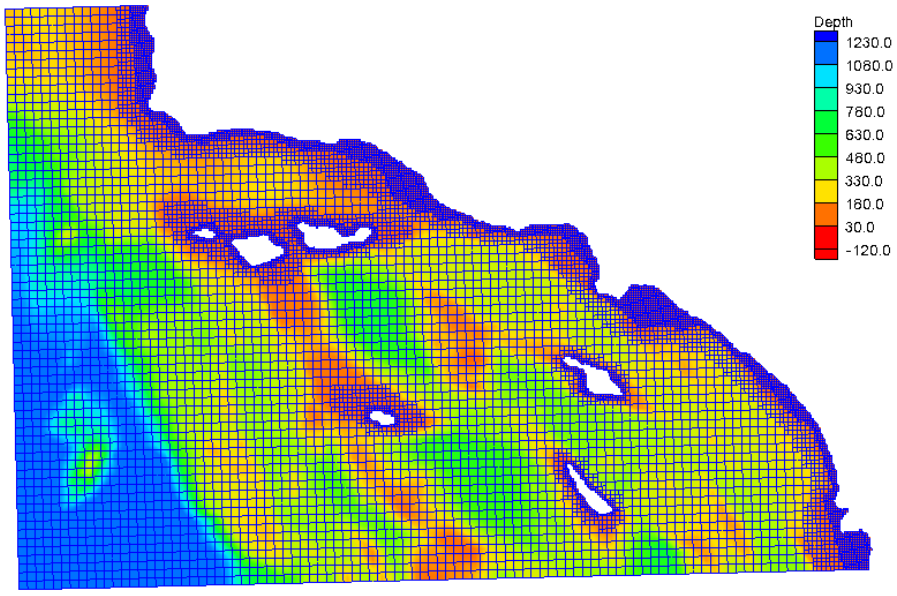


Figure 2-66. Example computational grid for southern California.

The offshore boundary is specified a tidal database boundary condition. The tidal dabase selected is the EC2001 database which is based on a regional ADCIRC simulation. In the case, the same regional database was used to generate the bathymetry for the CMS simulation. The CMS-Flow input cards used to specify the offshore boundary are described in the example below.

Example 2-58. Tidal database water level boundary specification.

```
BOUNDARY_BEGIN
   NAME           "Tidal Database Boundary"
   CELLSTRING     "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
   WSE_BEGIN
   WSE_END
   TIDAL_DATABASE_BEGIN
      NAME  ENPAC2003 !EC001 | ENPAC2003 | FES95 | FES2004 | LEPROVOST
      PATH  "E:\Tidal_Databases\ec2001\"
   TIDAL_DATABASE_END
BOUNDARY_END
```

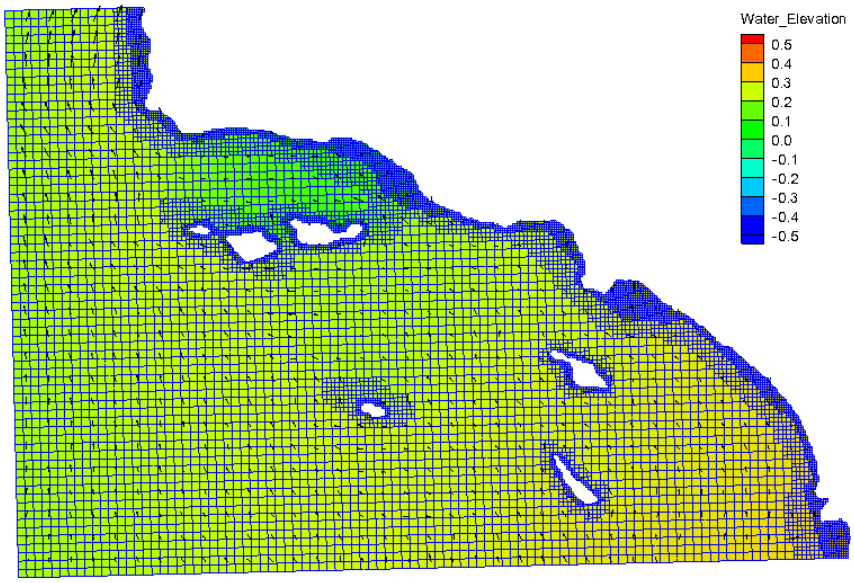


Figure 2-67. Sample water levels and current velocites for southern California forced with the water surface elevation constituents extracted from the ENPAC2003 tidal datbase.

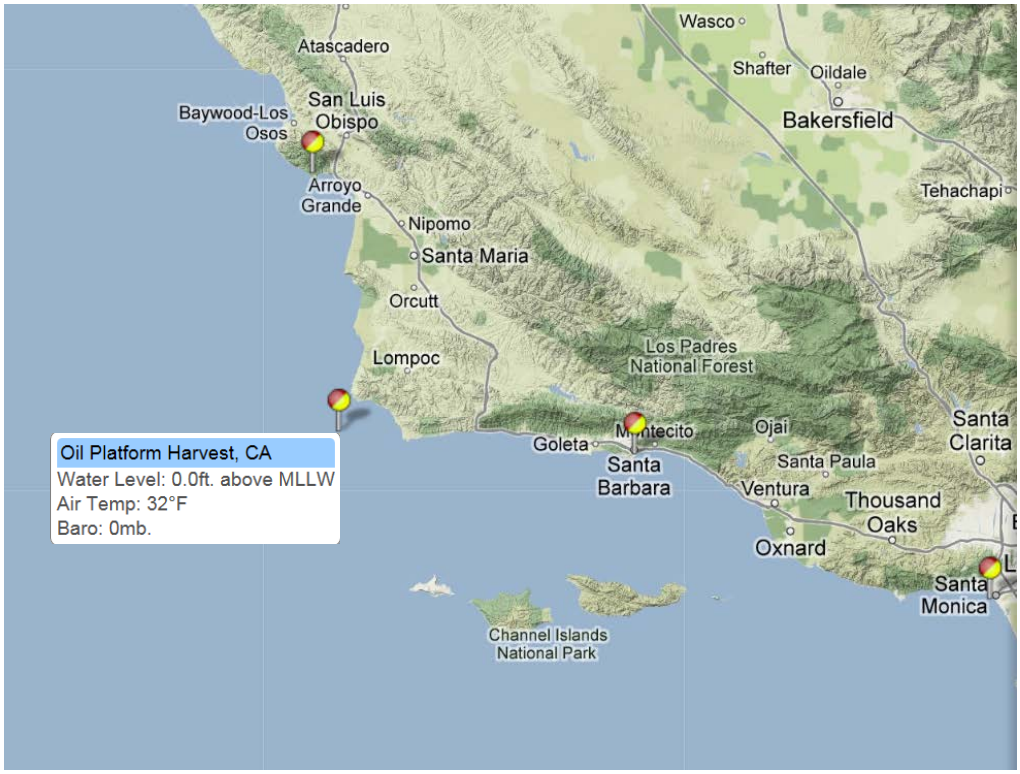
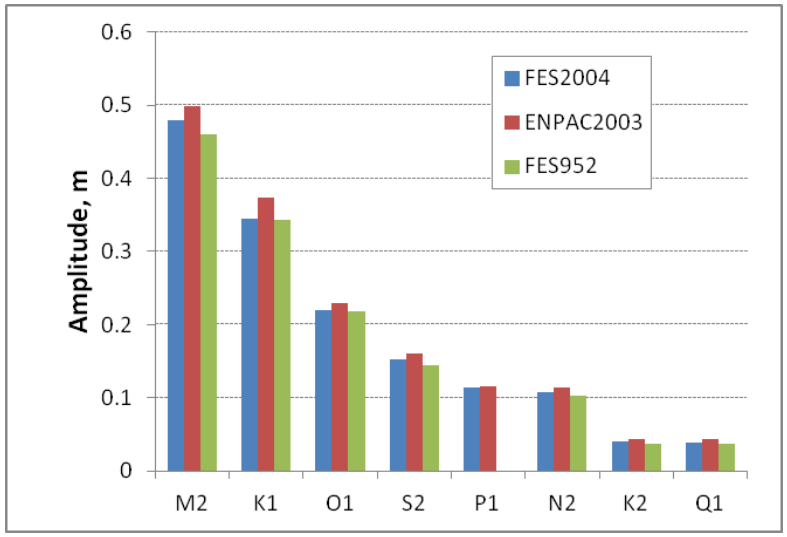
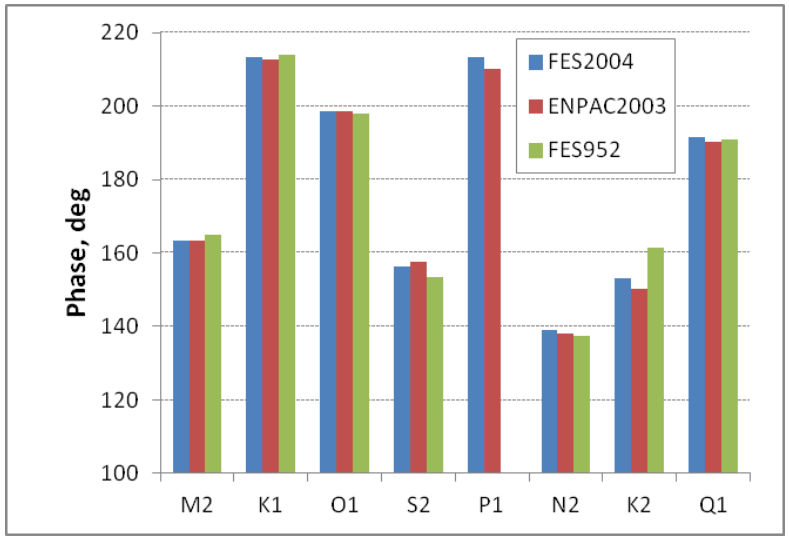


Figure 2-68. Location of Oil Platform Harvest, CA.

In order to show the variability in tidal constituents between different databases, the amplitudes and phases of three databases are presented in Figure 2-69 at the Oil Platform Harvest, CA. The ENPAC2003 database has the largest tidal amplitudes compared to the FES2004 and FES95.2 databases. It is noted that the FES95.2 database does not contain the P1 constituent. The phases are relatively similar with the largest differences occurring for the K2 constituent.



a.



b.

Figure 2-69. Comparison of tidal constituent amplitudes (a) and phases (b) at Oil Platform Harvest, CA, for three different tidal databases.

Mixed Water Level and Current Velocity Boundaries

There are three types of mixed water level and velocity boundary conditions supported using block structures

1. Multiple water level and velocity boundary condition
2. Nested water level and velocity boundary condition
3. Extracted water level and velocity using a tidal constituent database

Table 2-62. CMS-Flow cards related to the WSE and Currently Velocity boundary blocks.

Input	Format	Notes
Begins a WSE Boundary Block	[block=WSE_BEGIN, name=**WSEblock**]	Begins the block structure for a flux boundary condition
Ends a WSE Boundary Block	[block=WSE_END, name=**WSEblock**]	Ends the block structure for a flux boundary condition
Begins a Velocity Boundary Block	[block=VEL_BEGIN, name=**WSEblock**]	Begins the block structure for a flux boundary condition
Ends a Velocity Boundary Block	[block=VEL_END, name=**WSEblock**]	Ends the block structure for a flux boundary condition

Multiple Water Level and Current Velocity Time Series Boundary

The multiple water level and current velocity time series boundary condition (BC) is specified at each boundary cell. The BC is useful for nesting CMS within other models besides CMS and ADCIRC. It is noted that the water level and velocities are not strictly enforced at the boundary but rather specified as external values in the implicit temporal solution scheme. A first-order upwind is applied at boundary cells and therefore the external velocity is only used during inflow conditions. This has been found to work well for most field applications. Options are available for temporal interpolation and spatial smoothing. The BC is illustrated with the following two examples.

Example: SMS-generated Multiple Water Levels and Current Velocities

In the example below the SMS interface is used to create a multiple water level and current velocity time-series boundary and the CMS project files are saved. The CMS-Flow card file is then modified by commenting out the SMS generated cards MULTI_HDRIVER_CELLSTRING and MULTI_VDRIVER_CELLSTRING, and placing the card information in the user-specified boundary block. It is noted that both the cellstring information and the extracted water levels are contained within the same path of the CMS-Flow Model Parameters File (*_mp.h5). This is the reason why both the CELLSTRING and DATASET cards contain the same file name and path.

Example 2-59. Multiple water level and current velocity time-series boundary specification using a block structure.

```
#MULTI_HDRIVER_CELLSTRING "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
#MULTI_VDRIVER_CELLSTRING "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
BOUNDARY_BEGIN
   CELLSTRING    "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
   WSE_BEGIN
      DATASET    "Flow_mp.h5" "PROPERTIES/Model Params/Boundary" !Optional
      OFFSET     0.1 m    !Optional
      TIME_INTERP_ORDER    2 !Temporal interpolation
      TIME_SMOOTH_ITER     1  !Temporal smoothing iterations
      TIME_SMOOTH_WIDTH    3 !Temporal smoothing window width
      SPACE_SMOOTH_ITER    2 !Spatial smoothing iterations
      SPACE_SMOOTH_WIDTH   3 !Spatial smoothing window width
   WSE_END
   VEL_BEGIN
      DATASET    "Flow_mp.h5" "PROPERTIES/Model Params/Boundary" !Optional
      TIME_INTERP_ORDER    1 !Temporal interpolation
      TIME_SMOOTH_ITER     1  !Temporal smoothing iterations
      TIME_SMOOTH_WIDTH    3 !Temporal smoothing window width
      SPACE_SMOOTH_ITER    2 !Spatial smoothing iterations
      SPACE_SMOOTH_WIDTH   3 !Spatial smoothing window width
   VEL_END
BOUNDARY_END
```

Example: User-specified Multiple Water Levels and Current Velocities.

In the second example the cellstring is specified using the Boundary ID file (*.bid). This file is described in the *Cellstring* section of the present document. The water level and current velocity time series are specified at each boundary cell using the SMS Time-Series Data TSD file format (with extension *.tsd). Water levels and current velocities are specified in separate files. The time steps of the water levels and current velocities may be different. The temporal interpolation order and spatial smoothing may also be specified independently. However in the present example they are not modified from their default values. The CMS cards used to specify the BC are provided below.

Example 2-60. User-specified multiple water level and current velocity BC.

```
BOUNDARY_BEGIN
   NAME          "Mixed BC”
   CELLSTRING    "Boundaries.bid"  1 ![file name] [cellstring ID]
   WSE_BEGIN
      DATASET    "Flow_bnd_wse.tsd"
   WSE_END
   VEL_BEGIN
      DATASET    "Flow_bnd_vel.tsd"
   VEL_END
BOUNDARY_END
```

The SMS TSD files are convenient to use because they are ASCII files and include a time reference. Examples of water level and current velocity TSD files are provided below.

Example 2-61. Excerpt of a Time Series Data file (Flow_bnd_wse.tsd) used to specify water levels for a multiple water level and current velocity BC.

```
TIME_SERIES
"Boundary WSE"  "Unassigned"     7      100  "01/01/2011 00:00:00"
      0.000     0.0000    0.0000    0.0000    0.0000    0.0000    0.0000
   1800.000     0.0020    0.0021    0.0022    0.0023    0.0024    0.0025
   3600.000     0.0073    0.0074    0.0075    0.0076    0.0077    0.0078
   5400.000     0.0842    0.0843    0.0844    0.0845    0.0846    0.0847
   7200.000     0.1731    0.1732    0.1733    0.1734    0.1735    0.1736
   9000.000     0.2546    0.2547    0.2548    0.2549    0.2550    0.2551
   …
```

Example 2-62. Excerpt of a Time Series Data file (Flow_bnd_vel.tsd) used to specify current velocities for a multiple water level and current velocity BC.

```
TIME_SERIES
"Boundary Vel"  "Unassigned"     13      100  "01/01/2001 00:00:00"
        0.000      0.0000     0.0000     0.0000     0.0000    …
     1800.000     -0.0022     0.0005    -0.0022     0.0005    …
     3600.000      0.0038     0.0019     0.0038     0.0019    …
     5400.000      0.0132     0.0015     0.0132     0.0015    …
     7200.000      0.0242     0.0014     0.0242     0.0014    …
     9000.000      0.0354     0.0010     0.0354     0.0010    …
    10800.000      0.0461     0.0003     0.0461     0.0003    …
    12600.000      0.0554    -0.0004     0.0555    -0.0004    …
…
```

Notes:

- If the number of time steps is set to zero (forth entry in second line of TSD file), CMS will automatically determine the number from the input file. Therefore, the parameter may be considered optional when using as an input for CMS.
- The TSD files do not have to start at the same time as the CMS simulation but should in general cover the entire simulation period. If the values do not extend the entire simulation, then the nearest value is used.

Extracted Water Levels and Current Velocities from a Parent Simulation

As described previously, CMS supports automated one-way nesting within parent CMS and ADCIRC models. When using this feature, the CMS automatically extracts the boundary condition information from the parent model solution. The user only has to specify to point to the boundary cell-string and the location and name of the parent model simulation. The user may force the CMS with either water levels or water levels and velocities extracted from the parent simulation.

The table below describes the CMS cards used to specify a nested water level boundary condition in CMS.

Table 2-63. CMS-Flow cards related to the single tidal current velocity boundary condition.

Input	Format	Notes
Current Velocity solution	[card=VEL_SOLUTION, parent=**VELblock**] [arg=**VELsol**, type=*char*, options=PARENT]	Specifies the boundary as a nested current velocity boundary.
Current Velocity Temporal Interpolation Order	[card=TIME_INTERP_ORDER] [arg=**VELnti**, type=*int*, range=(1<**VELnti**<3), default=2]	Order of Lagrangian polynomial used to interpolate time-series.
Current Velocity Temporal Smoothing Iterations	[card=TIME_SMOOTH_ITER] [arg=**VELnsi**, type=*int*, range=(0<**VELnsi**<10), default=2]	Temporal smoothing iterations. Smoothing is done by a simple moving average.
Current Velocity Temporal Smoothing Window Width	[card=TIME_SMOOTH_WIDTH] [arg=**VELnsw**, type=*int*, range=(0<**VELnsw**<10), default=0]	Temporal smoothing window width. Should be an odd number Smoothing is done by a simple moving average.

Example: Nesting a CMS simulation within an ADCIRC simulation

If the ADCIRC simulation is 3D, it is important to export the depth-averaged current velocity file (fort.63 or *_.h5) so that this file can be used for current velocity extraction.

Example 2-63. Extracting water levels and velocities from parent ADCIRC simulation.

```
BOUNDARY_BEGIN
  NAME            "Nested ADCIRC wse and velocity boundary"
  CELLSTRING       "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
  WSE_BEGIN
    WSE_OFFSET    0.1 m
    TIME_INTERP_ORDER   2     !optional, temporal interpolation
    SPACE_SMOOTH_ITER   1     !optional, spatial smoothing iterations
    SPACE_SMOOTH_WIDTH  3     !optional, spatial smoothing window width
  WSE_END
  VEL_BEGIN
    TIME_INTERP_ORDER   2     !optional, temporal interpolation
    SPACE_SMOOTH_ITER   1     !optional, spatial smoothing iterations
    SPACE_SMOOTH_WIDTH  3     !optional, spatial smoothing window width
  VEL_END
  PARENT_BEGIN
    GRID_FILE      "..\ADCIRC_Regional\fort.14"
    WSE_FILE       "..\ADCIRC_Regional\fort.63"
    VEL_FILE       "..\ADCIRC_Regional\fort.64"
    STARTING_DATE_TIME  2012-12-01 00:00:00 UTC
    HORIZONTAL_PROJECTION_BEGIN !Optional, assumed if not specified
      DATUM       NAD83        !NAD27 | NAD83 | LOCAL
      SYSTEM      GEOGRAPHIC   !UTM | STATE_PLANE | GEOGRAPHIC | LOCAL
      UNITS       DEGREES      !METERS | FEET | DEGREES | etc.
    HORIZONTAL_PROJECTION_END
  PARENT_BEGIN
BOUNDARY_END
```

It is noted that since the ADCIRC simulation files do not contain information on the starting date or horizontal projection, this information should to be provided in the model input. If the horizontal projection is not specified, then it is assumed to be NAD83 Geographic in degrees.

In the example below, only the CMS Card file is specified for the parent simulation. The model automatically obtains the parent simulation starting time, horizontal projection, and output file names. The water level and velocity blocks are left empty so all of the default parameters are used.

Example 2-64. Interpolated water levels and velocities from parent CMS simulation.

```
BOUNDARY_BEGIN
  CELLSTRING              "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
  WSE_BEGIN
  WSE_END
  VEL_BEGIN
  VEL_END
  PARENT_BEGIN
    CONTROL_FILE          "..\CMS_Regional\Flow_Reg.cmcards"
  PARENT_END
BOUNDARY_END
```

Example: Nesting a CMS simulation within a CMS simulation

Example 2-65. Multiple water level and velocity boundary specification using a block structure.

```
BOUNDARY_BEGIN
  CELLSTRING    "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
  WSE_END
    WSE_FILE   "Flow_mp.h5" "PROPERTIES/Model Params/Boundary" !Optional
    WSE_OFFSET     0.1 m   !Optional
    WSE_GRADIENTS  1.E-5 0.0 !Optional
    TIME_INTERP_ORDER   2 !Temporal interpolation
    TIME_SMOOTH_ITER    1  !Temporal smoothing iterations
    TIME_SMOOTH_WIDTH   3 !Temporal smoothing window width
    SPACE_SMOOTH_ITER   1 !Spatial smoothing iterations
    SPACE_SMOOTH_WIDTH  3 !Spatial smoothing window width
  WSE_END
  VEL_BEGIN
    VEL_FILE   "Flow_mp.h5" "PROPERTIES/Model Params/Boundary" !Optional
    TIME_INTERP_ORDER   2 !Temporal interpolation
    TIME_SMOOTH_ITER    1  !Temporal smoothing iterations
    TIME_SMOOTH_WIDTH   3 !Temporal smoothing window width
    SPACE_SMOOTH_ITER   1 !Spatial smoothing iterations
    SPACE_SMOOTH_WIDTH  3 !Spatial smoothing window width
  VEL_END
  PARENT_BEGIN
    CONTROL_FILE         "..\CMS_Regional\Flow_Reg.cmcards"
  PARENT_END
BOUNDARY_END
```

Extracted Water Level and Current Velocity Tidal Constituents from a Database

TBC

Example 2-66. Tidal database water level boundary specification

```
BOUNDARY_BEGIN
   NAME           "Offshore boundary" !Optional
   CELLSTRING     "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
   WSE_BEGIN
   WSE_END
   VEL_BEGIN
   VEL_END
   TIDAL_DATABASE_BEGIN
      NAME  EC2001 !EC001 | ENPAC2003 | FES95 | FES2004 | LEPROVOST
      PATH  "E:\Tidal_Databases\ec2001\"
   TIDAL_DATABASE_END
BOUNDARY_END
```

Example 2-67. Tidal database water level boundary specification

```
BOUNDARY_BEGIN
   NAME           "Offshore boundary" !Optional
   CELLSTRING     "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
   WSE_BEGIN
      OFFSET      0.1 m   !Optional
   WSE_END
   VEL_BEGIN
   VEL_END
   TIDAL_DATABASE_BEGIN
      NAME        EC2001 !EC001 | ENPAC2003 | FES95 | FES2004 | LEPROVOST
      PATH        "E:\Tidal_Databases\ec2001\"
      CONSTITUENTS 4 M2 S2 K1 O1   !cnum cname(1:num), optional
   TIDAL_DATABASE_END
BOUNDARY_END
```

Step-by-Step Instructions for Setting-up Advanced Boundary Conditions

Advanced boundary conditions using boundary blocks are not yet supported in SMS and are therefore entered manually by editing the CMS-Flow *Control File*.

Option 1: Reassigning an existing boundary condition

This is the simplest option and requires the least amount of work from the user. The option is basically creating a "fake" boundary condition in SMS, saving the project files, so that SMS generates the important cellstring identification (ID) cells to the *CMS-Flow Model Parameters File* and then manually reassigning the boundary a different boundary condition in the *CMS-Flow Control File*.

1. Create and select a cellstring in SMS. Open the *CMS-Flow Boundary Conditions* window (see section Cellstrings for more information).
2. Assign any boundary condition to the boundary such as a tidal or river flux boundary condition. It is recommended to use a boundary condition type that is different from all of the others if possible so that it can be easily identified in the *CMS-Flow Control File* later. Note: The boundary information does not need to be specified in order for SMS to write out the cellstring information.
3. Save the *CMS-Flow P*roject Files by going to *File* | *CMS-FLOW*.
4. Open the *CMS-Flow Control File* in a text editor.
5. Create an advanced boundary condition block.
6. Copy-paste the cellstring file and path from the corresponding "fake" boundary condition which is being reassigned to the cellstring card within the new boundary block (see example below).
7. Specify any addition cards for the advanced boundary condition.
8. Either comment out or delete the corresponding "fake" boundary condition which is being reassigned (see example below).

Example 2-68. Converting the CMS flux boundary condition input from a single card format to a block structure format.

```
#QDRIVER_CELLSTRING   "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
BOUNDARY_BEGIN
   CELLSTRING    "Flow_mp.h5" "PROPERTIES/Model Params/Boundary"
   FLUX_BEGIN
      VALUE        5.0 'm^3/s' !Constant water flux
   FLUX_END
BOUNDARY_END
```

Option 2: Copying cellstring ID's from the CMS-Flow Model Parameters File into a Boundary ID File

1. Create and select a cellstring in SMS.
2. Assign any boundary condition to the boundary such as a tidal or flux boundary condition.
3. Save the *CMS-Flow* Project Files by going to *File | CMS-FLOW*.
4. Open the *CMS-Flow Model Parameters File* using HDFView.
5. Find the cellstring for the corresponding boundary condition and copy-paste the cell ID's to an ASCII Boundary ID File (*.bid) (see Section Cellstrings for more information on the Boundary ID File).
6. Open the *CMS-Flow Control (Card) File* in a text editor, and enter the advanced boundary condition using a block structure making sure to reference the Boundary ID File and cellstring ID (see example below).

Example 2-69. Specifying a flux boundary condition using all ASCII input files.

```
BOUNDARY_BEGIN
  NAME             "River"
  CELLSTRING       "Boundaries.bid"  1 ![file name] [cellstring ID]
  FLUX_BEGIN
    CURVE          "Flux.xys"  !Contains flux data
    UNITS          'm^3/s'     !Optional, default is m^3/s/cell
    DIRECTION      110.0 'deg' !Optional, -360<real<360 in degrees
  FLUX_END
BOUNDARY_END
```

Preprocessing CMS-Flow Input Time Series

Typically, time series of water levels, winds, river fluxes, salinity, and other types are specified as boundary conditions and it is extremely important that this data be prepared properly for use in the model. Although measured data is often regarded as the "truth", no instrument is free of errors, failures, or immune to human misuse and mistakes. In numerical models "If junk goes in, junk goes out". Time series analysis is a topic much too large to be covered in this manual, and only key points are described. For an in-depth review of time series analysis the reader is referred to Emery and Thompson (1998). The general steps for preprocessing time series for CMS are:

1. Data Assembly and Conversion
2. Quality Control (QC)
3. Interpolation
4. Filtering

Data Assembly and Conversion

This step includes the download, format conversion (e.g. binary to ASCII), and conversion of units, reference frames, datum's, etc. Often time series must be compiled from several raw data files.

Quality Control

Before any manipulation of the data is made, a QC must be performed to remove inaccurate or unreliable data. The definition of inaccurate data depends on the measurement type and its use in a model. Bad data should be removed from the time series before continuing and manipulating the data any further.

Interpolation

Gaps or "holes in data are very common in geophysical measurements. In some cases the model input may require that it be spaced at regular intervals. Therefore, any gaps in the data must be interpolated. It is more convenient to have the data at uniformly spaced intervals.

Note:

- CMS input time series can have irregular intervals.
- It is important that the input time series cover the extent of the simulation to avoid extrapolation issues.
- CMS uses a piece-wise quadratic interpolation for input boundary condition forcing by default. This may be changed but requires using advanced boundary blocks (see Section Advanced Boundary Specification).

Filtering

Measured data often noise or physical processes at frequencies other than the frequencies that want to be used as a mode input. Filtering is the process of removing the signals at any frequencies other than the desired frequencies.

Tip:

- A Matlab Guided User Interface (GUI) for preprocessing model time series is provided on the CIRP wiki (http://cirp.usace.army.mil/wiki/Utilities#Time_Series_Analysis) called *filter1d.m*. A hands-on tutorial on how to use this utility is provided in Appendix G.

Notes:

- The wave grid for most inlet and coastal cases will be not much longer alongshore than the flow model. If the wave model does not extend far enough, there are model parameters to smoothly interpolate the wave data to some distance alongshore in the flow model.
- Because the grid boundary is often the location of a wave buoy, the cross-shore domain tends to extend further.
- In some cases the best IJ location for the half-plane model may not be apparent, especially for open coasts. Often the degree of energy in the open ocean is dominantly from one direction or the other, and the i-direction should be aligned to that direction.

Salinity Transport

Overview

Salinity refers to the salt content of water. Its value runs typically from 0 for fresh water to 31-35 ppt (parts per thousand) for ocean water. In water bodies with poor mixing and limited water exchange, or experiencing high evaporation, salinity can be higher and lead to formation of brine (see Table 4-21) taken from the Wikipedia, presents typical values and nomenclature for describing degree of saline water:

Table 2-64. Water salinity classification.

Fresh water	Brackish water	Saline water	Brine
< 0.05 %	0.05 – 3 %	3 – 5 %	> 5 %
< 0.5 ppt	0.5 – 30 ppt	30 – 50 ppt	> 50 ppt

In coastal zones and estuaries, both temporal and spatial variations in salinity are controlled by changes in circulation, waves, tides, precipitation, evaporation, and freshwater inflows. These changes in salinity can have major effects on water density and water stratification, changing circulation patterns. Dynamic behavior of suspended sediment can be controlled by the salinity-driven flow and mixing. Any sustained changes to salinity can directly change the aggregation and consolidation of cohesive sediment as well (Nicholson and O'Connor 1986). Salinity can also alter the water chemistry that is closely related to marine organisms. Distribution and abundance of marine life will change water turbidity and define water quality in coastal and estuarine systems. Modifications of coastal inlets, such as channel deepening and widening and rehabilitation or extension of jetties may alter the salinity distribution within the estuary.

Setup

The salinity transport is activated within the *CMS-Flow Model Control* window (see figure below).

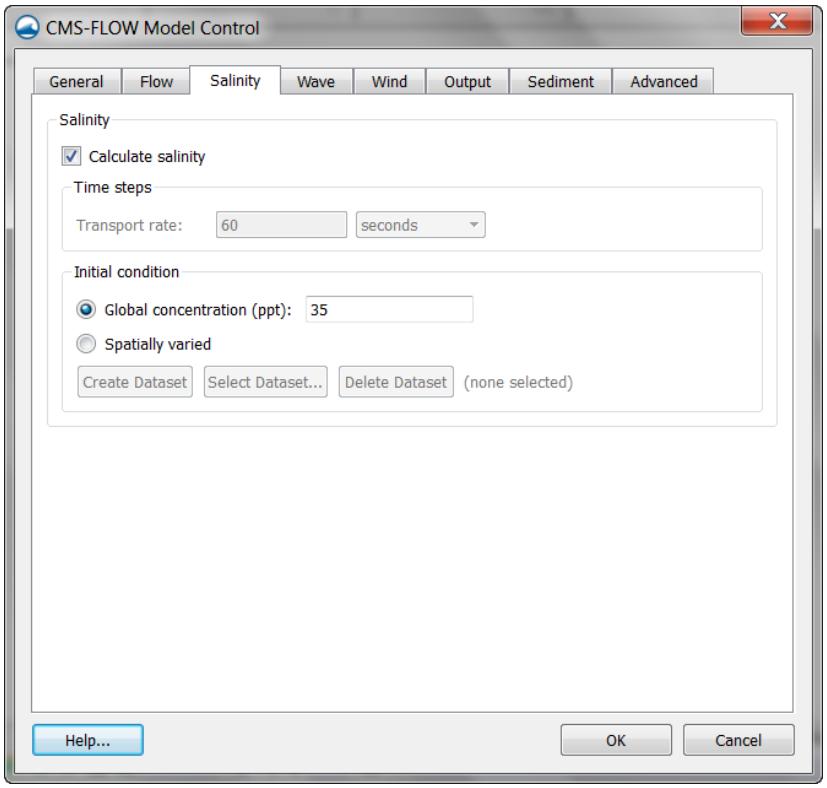


Figure 2-70. Salinity tab in the *CMS-Flow Model Control* window in SMS 11.1.

The CMS card used to turn on or off the salinity transport is described in Table 2-68.

Table 2-65. CMS card used to turn *On* or *Off* the salinity transport

Input	Format	Notes
Salinity Transport Activation	[card=CALC_SALINITY_TRANSPORT] [name=**SalTrans**, type=*char*, options=(ON,OFF), default=OFF]	Turns on or off the salinity transport calculation.
Salinity Transport Time Step	[card=SALINITY_CALC_INTERVAL] [name=**SalTransdt**, type=*float*, default=1.0, dependency=] [name=**SalTransdtUnits**, type=*char*, options=**TimeUnits**), default='s']	Specifies the salinity transport time step (interval). Only used for explicit temporal scheme.

Initial Condition

The salinity initial condition is specified from within the Salinity tab of the *CMS-Flow Model Control* window (see Figure 2-74. Two options are available for the initial condition: (1) a constant value, or (2) a user-specified dataset.

The CMS card used to specify the salinity initial condition is described in Table 2-68.

Table 2-66. CMS card used to specify the salinity initial condition

Input	Format	Notes
Salinity Initial Condition Constant	[card=SALINITY_IC_CONSTANT] [name=**SalIC**, type=*char*, options=(ON,OFF), default=OFF]	Specifies the salinity initiation condition as a constant value for the whole solution domain.
Salinity Initial Condition Dataset	[card=SALINITY_IC_DATASE] [name=**SalICfile**, type=*char*, default=**none**] [name=**SalICpath**, type=*char*, default=**none**]	Specifies the salinity initiation condition dataset file and path.

Note:

- For some applications, using a constant value for the initial condition may be lead to a long spin-up time for salinity. The spin-up time is the time it takes for the salinity to reach a quasi equilibrium or in other words the time it takes for the effects of the initial condition to disappear. A better approach is to use a spatially-varying user-specified dataset. The dataset may be obtained from a previous simulation or manually created based on user judgment.

Boundary Conditions

The salinity boundary conditions are setup in the *CMS-Flow Boundary Conditions* window (see figure below). Currently, the CMS only supports a single salinity time series boundary condition. In this boundary condition, a constant salinity is assigned along the whole boundary. The value is interpolated in time from a user specified time series. The salinity must be specified at ALL open boundary conditions. The user specified salinity is only applied for inflow conditions. When the flow is directed outward of the computational domain, a zero-gradient boundary condition is applied.

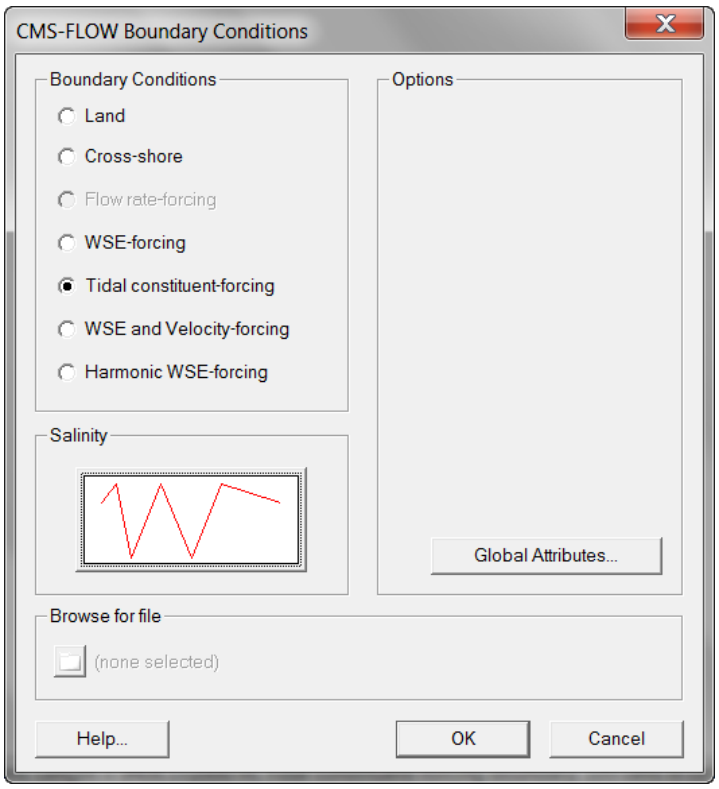


Figure 2-71. Specification of boundary salinity time-series in the *CMS-Flow Boundary Conditions* window.

The CMS card used to turn on or off the salinity transport is described in Table 2-68.

Table 2-67. CMS card used to turn *On* or *Off* the sediment transport

Input	Format	Notes
Saliniity Transport Activation	[card=CALC_SALINITY_TRANSPORT] [name=**SalTrans**, type=*char*, options=(ON,OFF), default=OFF]	Turns on or off the salinity transport calculation.

Notes:

- The salinity must be specified at all open boundary conditions. In addition the input time series at each boundary must be at least as long as the simulation duration.
- The salinity is NOT used to update the water density or viscosity during the simulation. The CMS uses a constant water density and viscosity. Therefore density gradients are ignored in the momentum equations.
- Turning on the salinity transport increases the computational time because an additional transport equation needs to be solved.

Advanced Settings

To Be Completed

Step-By-Step Salinity Transport Application for Matagorda Bay, TX

Matagorda Bay is the largest estuarine bay on the coast of Texas and is connected to the Gulf of Mexico and the Gulf Intracoastal Waterway (GIWW) through Matagorda Ship Channel (MSC), a federally-maintained inlet, and Pass Cavallo, a natural inlet just downdrift from the MSC (Figure 2-72).

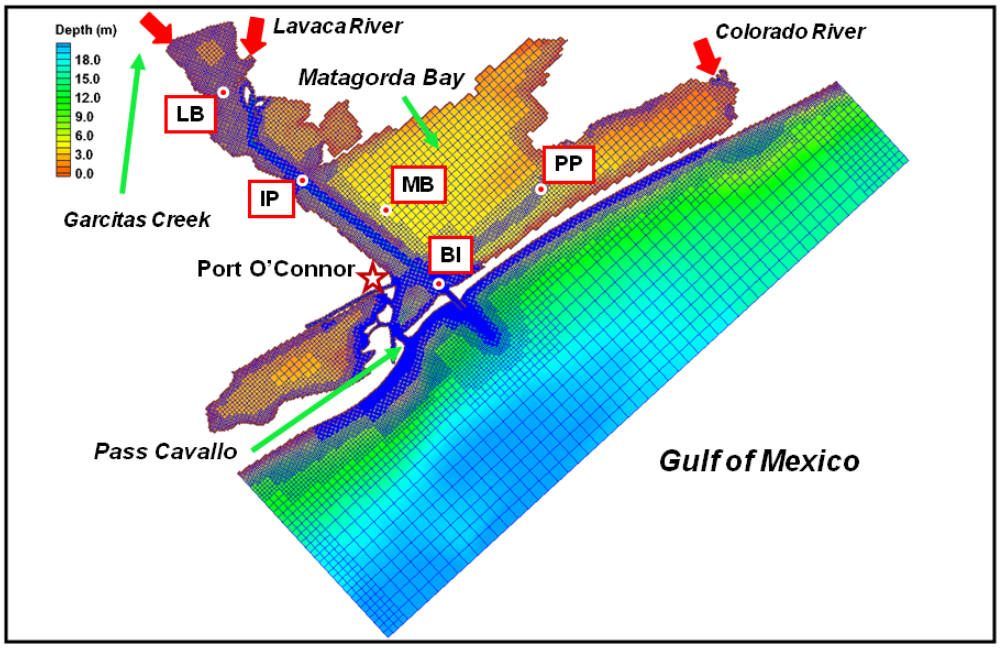


Figure 2-72. CMS domain, quadtree grid, and bathymetry of Matagorda Bay, TX. Red dots are the survey stations and red arrows indicate freshwater inflow locations.

In application of the CMS to Matagorda Bay, a telescoping grid was developed to discretized the bay and the offshore. The computational domain extends approximately 80 km alongshore and 20 km offshore, and the seaward boundary of the domain reaches to the 25-m isobath. Figure 2-72shows the telescoping grid with 70,000 ocean cells, bathymetric features of Matagorda Bay, and the adjoining nearshore area. The CMS grid permits fine resolution in areas of high interest such as jetties and channels. The implicit solver of the CMS, with a large time step of 15 minutes, was employed for the simulation.

Freshwater discharges into the bay come from a number of streams along the coast. The Colorado and the Lavaca Rivers provide most of the inflows. However, "the freshwater discharge is typically less than 10 percent of the daily tidal exchange" in the bay (Kraus et al. 2006). The bay entrance is protected by dual jetties from ocean waves. Momentum transfer, diffusive process and spatial distributions of salinity in the system are mostly controlled by wind, tide, and freshwater inflows.

CMS-Flow is driven by time-dependent water surface elevation at the offshore open boundary, wind forcing over the air-sea interface, and freshwater inflows from rivers and tributaries. Time varying salinity values at BI are also specified along the open boundaries with the water surface elevation and the river boundaries with the freshwater inflows. The initial salinity field is specified to the entire CMS domain as well.

5. **CMS-Flow setup:** The CMS hydrodynamic input files for Matagorda Bay are required and prepared by the SMS shown in Figure 2-73.

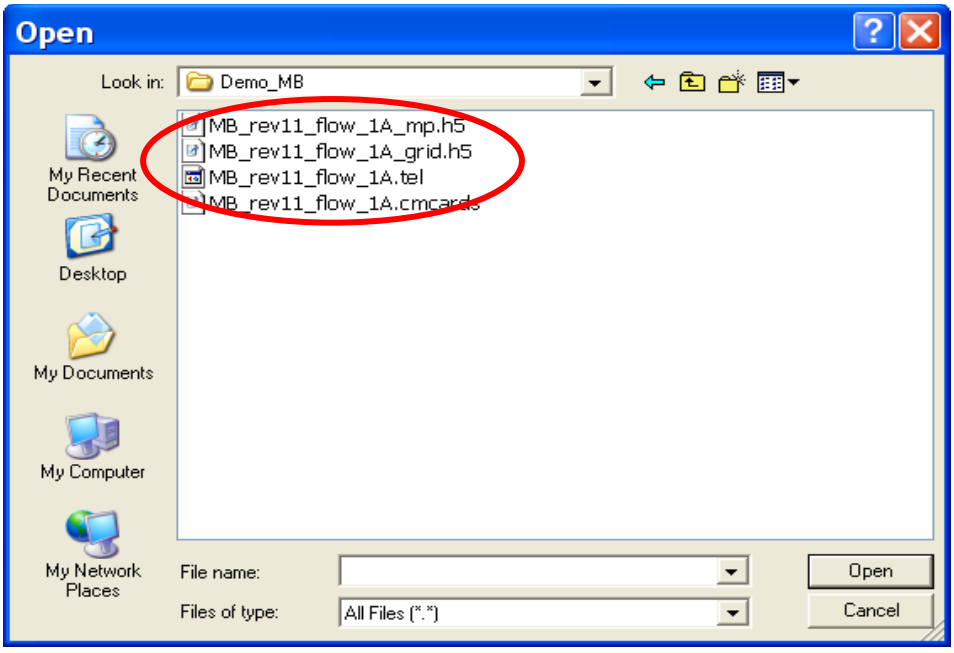


Figure 2-73. Files for the CMS-Flow salinity simulation.

After opening "MB_rev11_flow_1A.cmcards" in the SMS, choose CMS-Flow | Model Control, click on Salinity, and select Calculate salinity (Figure 2-74). A default time step equal to the hydrodynamic time step has been specified. In this case, 900 sec is used for the salinity calculation.

6. **Salinity initial condition:** Because of the large spatial variability of salinity in a coastal system, it usually requires long spin-up periods for a salinity simulation to reach to the present salinity distribution, which could range from a few days to weeks. To shorten the spin-up time, an accurate initial condition for the salinity field should be specified. There are two options to assign the initial salinity condition in CMS-Flow:

7. **A global initial salinity:** Specify a constant initial value for the entire model domain. The salinity value can be specified by checking the *Global concentration (ppt)* under the *Initial condition* (Figure 2-74). If this option is applied, it is best to define an average representative salinity for the entire domain.
8. **Spatially varying initial salinity:** Generate a spatially varying initial salinity field by choosing the *Spatially varied* toggle under the *Initial condition* (Figure 2-74). Clicking the *Create Dataset* and assigning a value under the *Default concentration (ppt)* in the pop-up window will generate a new dataset with a constant initial salinity value. Clicking *OK* to close this window and then clicking *OK*, to close the *CMS-FLOW Model Control* window, will cause the dataset, *Salinity Initial Concentration*, to appear in the CMS-Flow data tree, as shown in Figure 2-75a. Highlight the dataset to specify different salinity values in the CMS domain in the same way to modify other datasets such as *D50* or *Hard Bottom*.

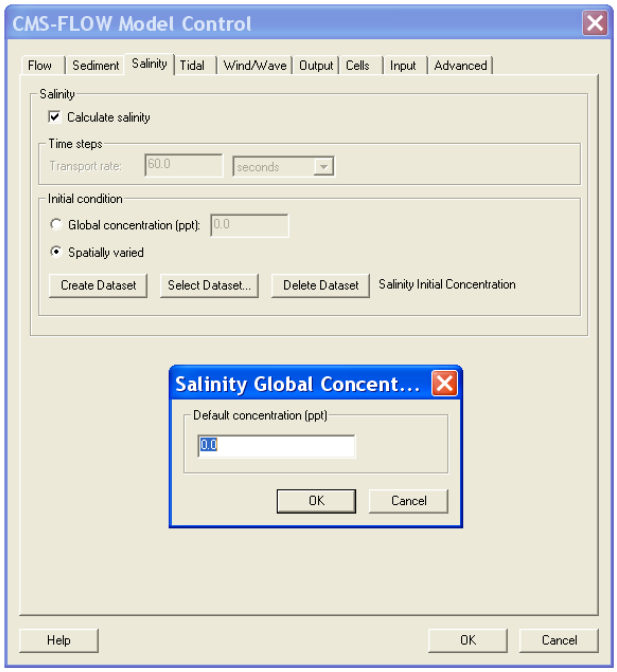
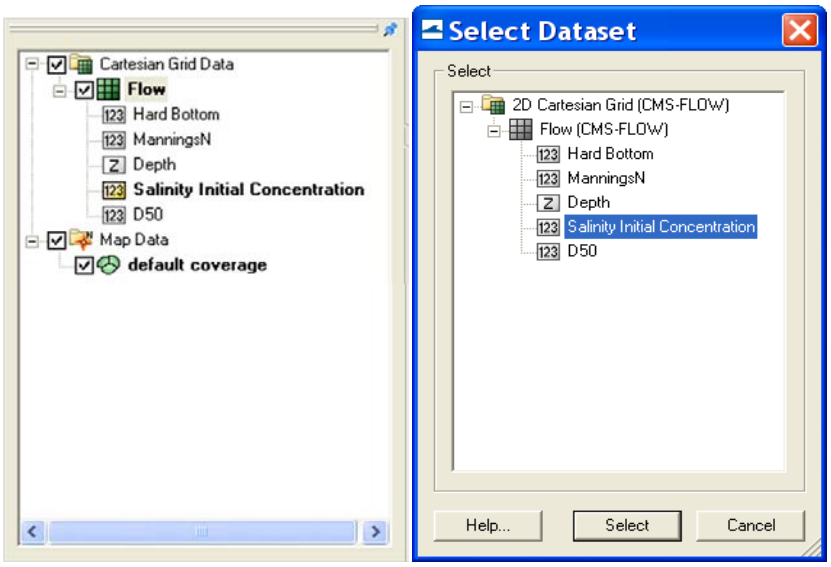


Figure 2-74. Setting up the salinity calculation and specifying spatially varied initial salinity.



(a) (b)

Figure 2-75. CMS-Flow data tree.

The dataset for a spatially varying initial salinity can also be generated by using the *Data Calculator* tool in the *Data* menu (Demirbilek et al. 2007). For an existing dataset, click the *Select Dataset* under the *Spatially varied* toggle and then select the dataset for the initial salinity that already exists (Figure 2-75b).

Based on the historical survey data, initial salinity is assigned in the dataset, *Salinity Initial Concentration*, for the Matagorda Bay system. The salinity varies from 21.0 ppt near the mouth of the Lavaca River to 33.0 ppt at the offshore open boundary (Figure 2-76).

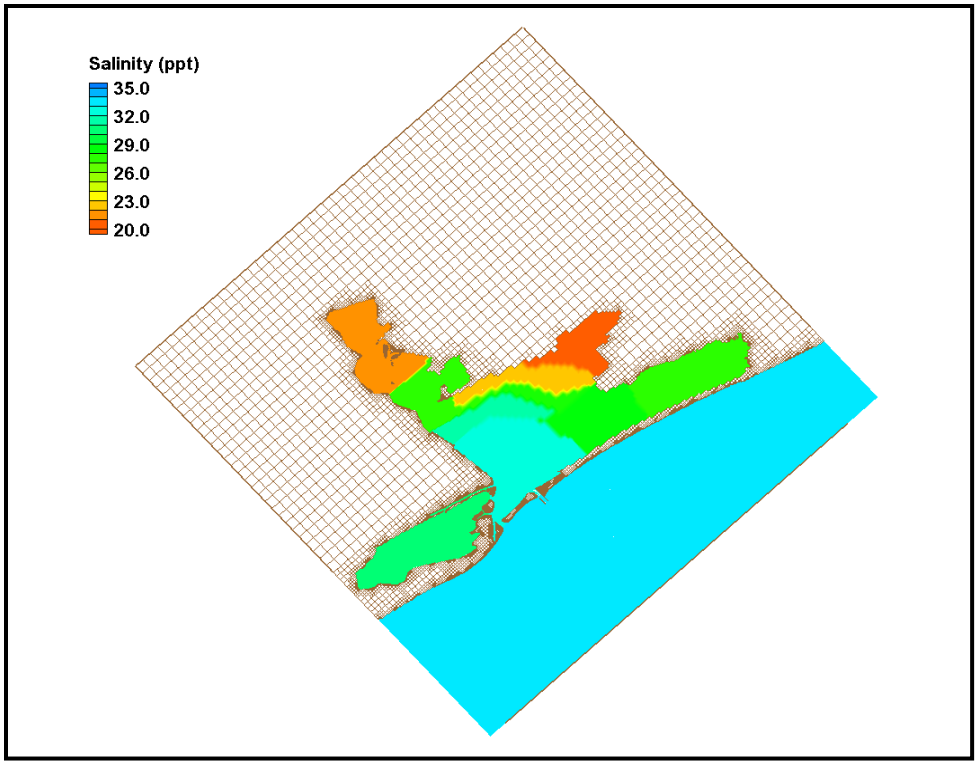


Figure 2-76. Initial salinity distribution.

9. **Salinity boundary conditions:** To calculate salinity transport, salinity values at CMS-Flow boundaries need to be specified. Salinity may be specified at two boundary types in the CMS: water surface elevation (WSE) boundary (*WSE-forcing* boundary) (Figure 2-77a) and freshwater inflow boundary (*Flow rate-forcing* boundary) (Figure 2-77b).

10. **WSE-forcing boundary:** Using the *Select Cellstring* tool and clicking/ highlighting, the cellstring of water surface elevation boundary can be specified as shown in Figure 2-77a. Selecting *CMS-Flow | Assign BC* will open the *CMS-Flow Boundary Conditions* window (Figure 2-77). A time series of salinity can be assigned along the *WSE-Forcing* boundary by clicking the *Curve undefined* under *Salinity* on the left hand side of the dialog.
11. The time series is specified either by clicking the *Import* button to read a salinity boundary input file in xys format (Figure 2-78) (Aquaveo 2010), or by entering time and salinity values manually in two separate data columns, or by importing salinity data from an opened *Excel* file by using *Copy/Paste*.

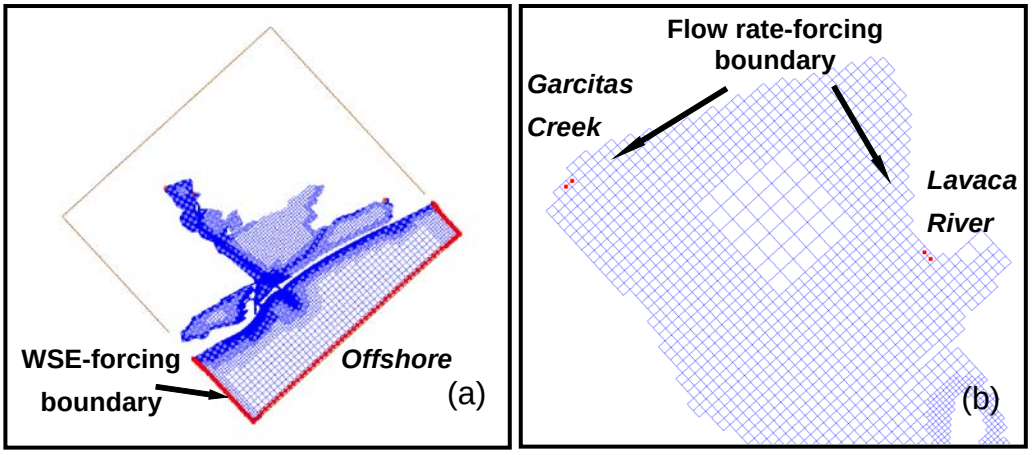


Figure 2-77. Salinity boundary types in the CMS.

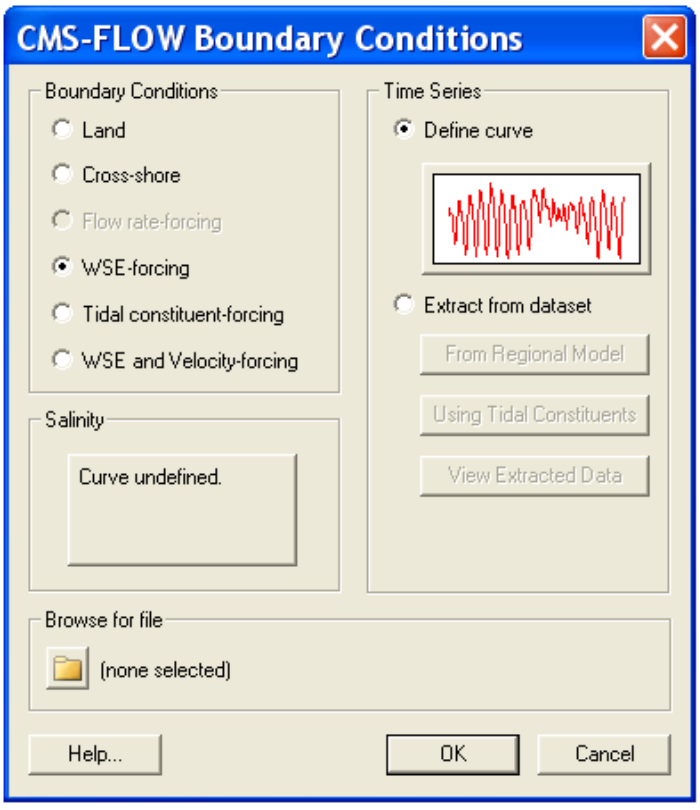


Figure 2-78. Salinity specifications along the *WSE-forcing* boundary.

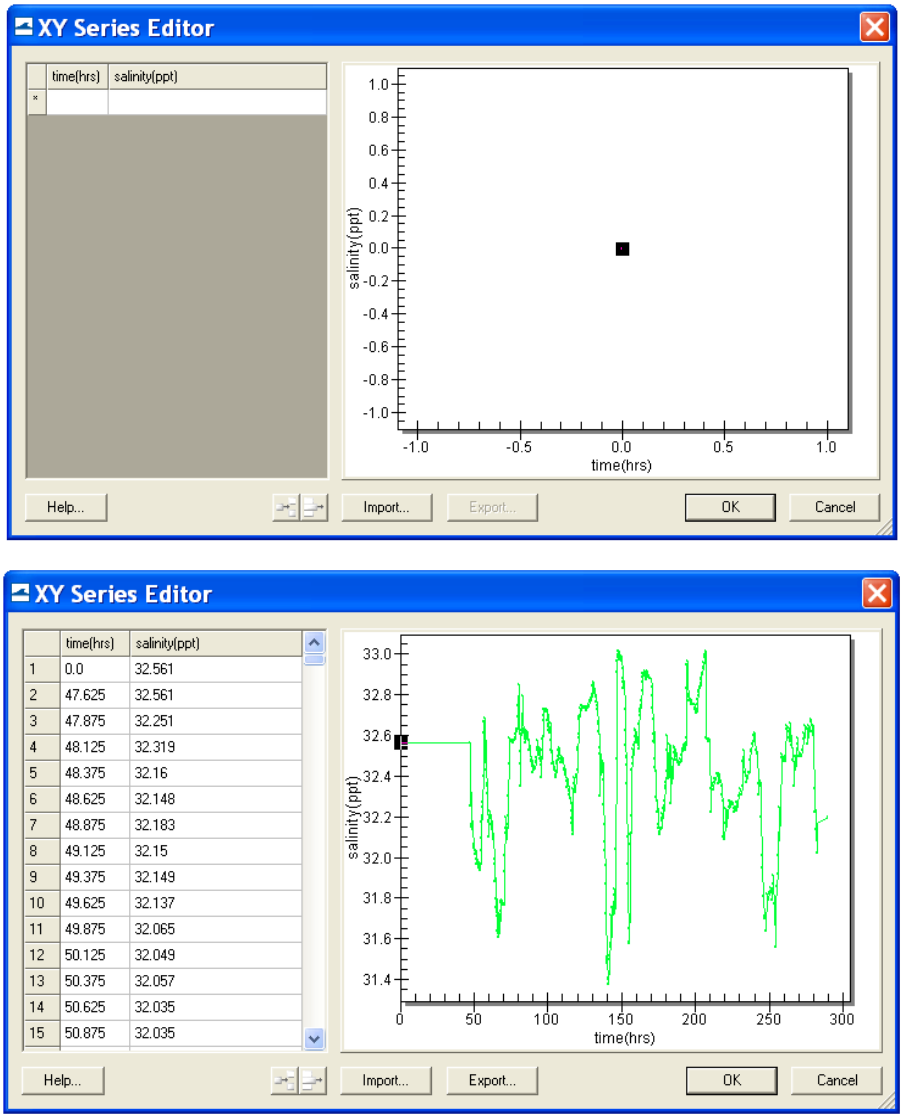


Figure 2-79. Salinity boundary input from a xys file.

The salinity and water surface elevation measurements at Pass Cavallo and the NOAA Corpus Christi Gage were assigned to the offshore boundary. The 12-day time series of salinity data (November-December 2005) is shown for the *WSE-forcing* boundary in Figure 2-79. Salinity at this location varies between 31.5 and 33.0 ppt and shows apparent influence of the ocean during the period.

12. **ii) *Flow rate-forcing* boundary:** Following the same steps as specifying *WSE-forcing* boundary, salinity values at freshwater inflow boundaries can be assigned together with flow specifications.

13. The Colorado River, the Lavaca River, and the Garcitas Creek are fresh water sources that flow into Matagorda Bay and flow measurements are available at three USGS gages. A zero salinity value is assigned at the *Flow rate-forcing* boundaries.

The CMS' capability in conducting the depth-averaged salinity calculation in Matagorda Bay was demonstrated. The CMS simulations represent the salinity transport in Matagorda Bay to a level useful for comparison between engineering alternatives, and to understand the temporal variation and spatial distribution of salinity and the interaction between tides, freshwater inflows and meteorological conditions in the bay.

Sediment Transport

The sediment transport controls are located in the *Sediment* tab of the *CMS-Flow Model Control* window. The sediment tab for the SMS 11.0 and SMS 11.1 interfaces are shown in Figure 2-80 and Figure 2-81, respectively. The sediment transport is activated by going to the *Transport* section of the *CMS-Flow Model Control* and checking the box labels *Calculate sediment transport*. When the checkbox is unchecked, all of the sediment transport options become dimmed in SMS 11.0 or disappear in SMS 11.1. When the checkbox is checked the options become active or appear.

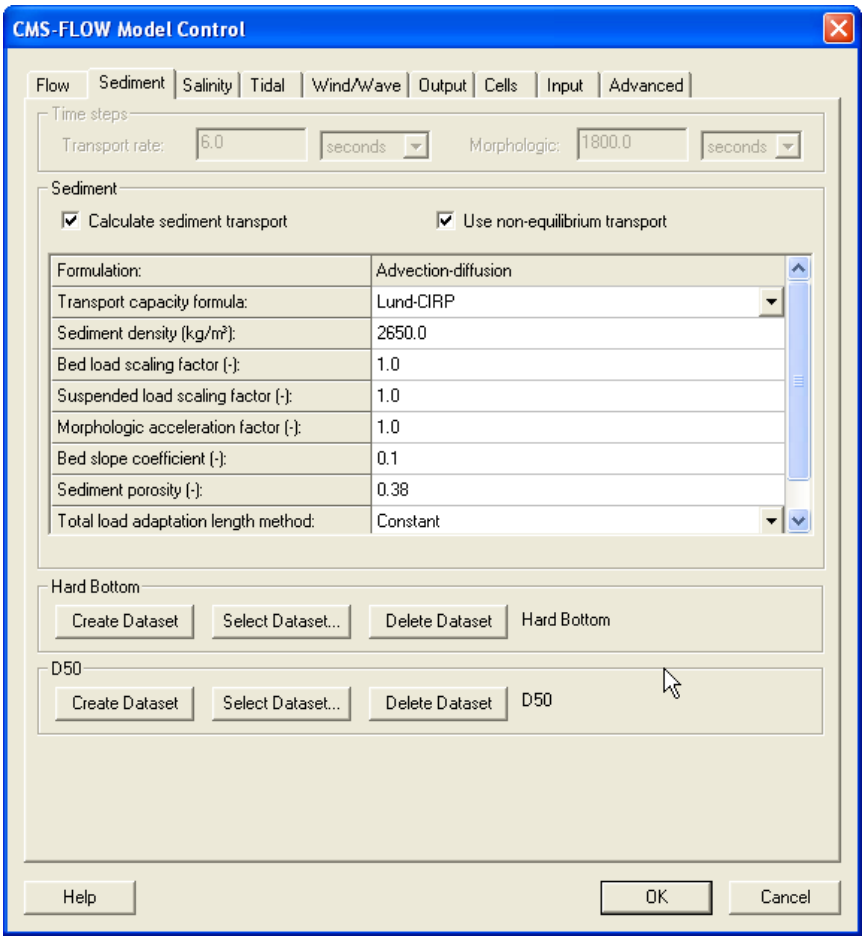


Figure 2-80. SMS 11.0 interfaces for sediment transport.

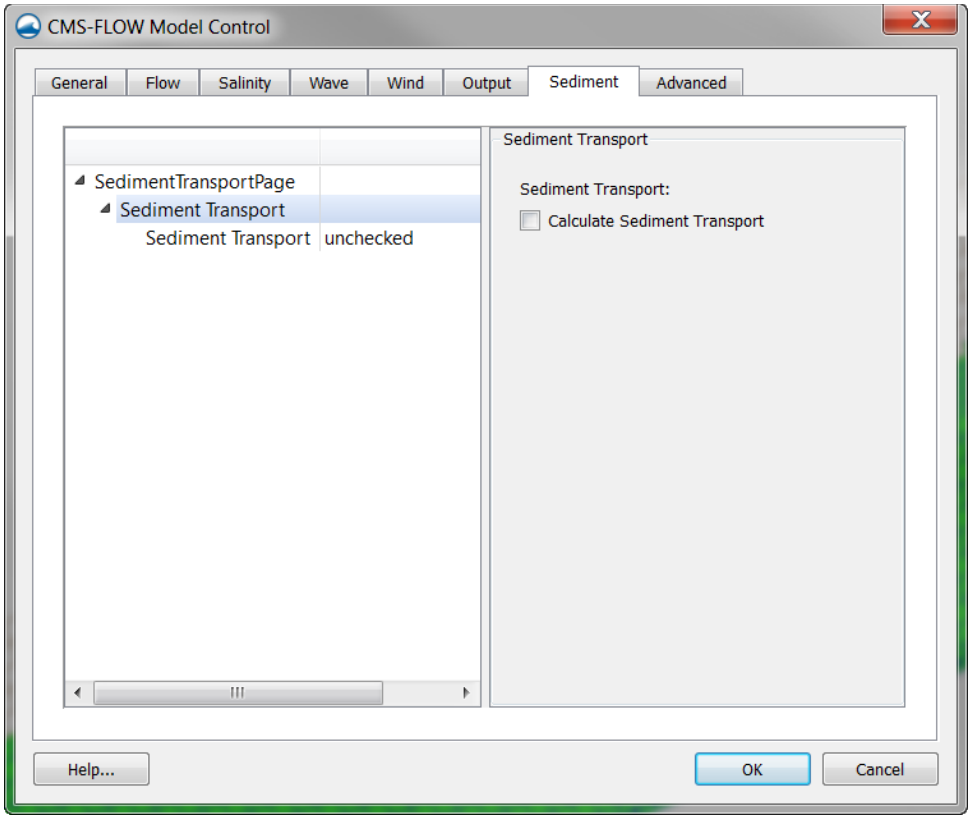


Figure 2-81. SMS 11.1 interfaces for sediment transport (sediment transport deactivated).

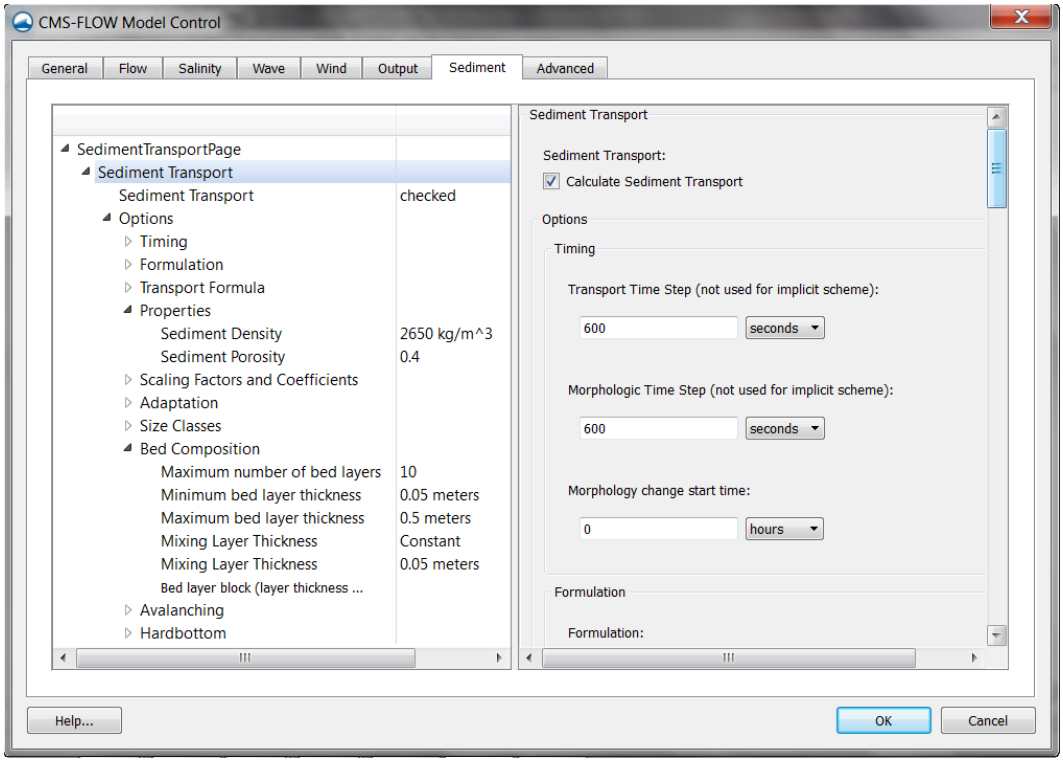


Figure 2-82. SMS 11.1 interfaces for sediment transport (sediment transport activated).

The CMS card used to turn on or off the sediment transport is described in Table 2-68.

Table 2-68. CMS card used to turn *On* or *Off* the sediment transport

Input	Format	Notes
Sediment Transport Activation	[card=CALC_SEDIMENT_TRANSPORT] [name=**SedTrans**, type=*char*, options=(ON,OFF), default=OFF]	Turns on or off the sediment transport calculation.

Timing

The sediment transport and morphologic time steps are the time steps at which the transport and bed change equations are calculated. For the explicit model, these time steps can be several times that of the hydrodynamic model. Table 2-69 gives a description, default value, and range for the sediment transport and morphologic time steps. The option is available to not calculate the morphology change during the ramp period. The best practice is the start the model simulation so that the time when the ramp period ends corresponds to the time of the measured bathymetry. This avoids the initial bed erosion (although slight) of the bed. This also facilitates calculating simulation statistics such as transport rates and residual currents.

Table 2-69. CMS card used to set the sediment transport and morphologic time steps.

Input	Format	Notes
Sediment Transport Time Step	[card=SED_TRAN_CALC_INTERVAL] [name=**SedTransdt**, type=*float*, default=1.0, dependency=] [name=**SedTransdtUnits**, type=*char*, options=**TimeUnits**), default='s']	Time step used for transport equation. See note 2 below.
Morphologic Time Step	[card=MORPH_UPDATE_INTERVAL] [name=**SedMorphdt**, type=*float*, default=10.0, range=(>=0)] [name= **SedTransdtUnits**, type=*char*, options=**TimeUnits**), default='s']	Time step used for updating bed elevation. See note 2 below.
Turns on the morphology change during the ramp period	[card=CALC_MORPH_DURING_RAMP] [name=**CalcMorphRamp**, type=*bool*, default=15.0]	Turns On or Off the morphology change calculation during the ramp period
Time to start the morphology change	[card=MORPH_START_TIME] [name=**MorphStartTime**, type=*float*, default=0.0, range=(>=0)] [name= **MorphStartTimeUnits**, type=*char*, options=**TimeUnits**), default='s']	Sets the starting time for the morphology change calculation.
Time to start the bed composition calculation	[card=BED_COMP_START_TIME] [name=**MorphStartTime**, type=*float*, default=0.0, range=(>=0)] [name= **MorphStartTimeUnits**, type=*char*, options=**TimeUnits**), default='s']	Sets the starting time for the bed composition calculation.

Example 2-70. Specifying the sediment transport timing for the explicit temporal scheme.

```
SED_TRAN_CALC_INTERVAL     10.0 'sec' !ONLY for explicit scheme
MORPH_UPDATE_INTERVAL      10.0 'sec' !ONLY for explicit scheme
MORPH_START_TIME           120.0 'hrs !Recommend: longer than ramp
```

Example 2-71. Specifying the sediment transport timing for the implicit temporal scheme.

```
MORPH_START_TIME          120.0 'hrs  !Recommend: longer than ramp
BED_COMP_START_TIME       24.0 'hrs'  !ONLY available for implicit scheme
```

Notes:

- For the explicit solver, the sediment transport and morphologic time steps should always be equal or a multiple of the hydrodynamic time step.
- For the implicit solver, the sediment transport and morphologic time steps are always set to the hydrodynamic time. In SMS 11.0 the user is not allowed to change them for the implicit temporal scheme. However, in SMS 11.1 the user is allowed to set the sediment transport time steps. It is important to remember that these time steps are not used when selecting the implicit temporal scheme.
- When comparing measured and computed morphology change, it is recommended to start the simulation before the time of the initial measured bed change so that the end of the spin-up time matches the start time of the morphology change and the time of the initial bathymetry. The spin-up time is the time it takes for the hydrodynamics to reach a dynamic equilibrium and varies depending on the site. At a minimum the spin-up time should be equal to the ramp period.

Sediment Properties

The sediment properties are set in the "Sediment" tab of the "CMS-Flow Model Control" window. The sediment properties used in CMS are the density, porosity, fall velocity, and Corey shape factor. The sediment density and porosity are constant and thus referred to as global sediment properties. The fall velocity is must be either specified or calculated for each size class and are therefore referred to as size class properties.

Global Sediment Properties

The global sediment properties are the sediment properties which are constant both spatially and temporally. Currently in CMS these properties are the sediment density and porosity. The sediment density depends on the mineralogy of the sediment. Most natural coastal sediments are composed of mostly quartz (SiO_2) due to its abundance and it's resistance to physical and chemical weathering. The density of quartz is approximately 2,650 kg/m^3. This is why it is often assumed to be the density of natural sands. However natural beaches often contain other minerals such as lighter feldspar and heavier miners including hornblende, magnetite, and garnet. In tropical regions coastal sediments may contain large fractions of bioclastic carbonate material from corals and coralline algae. In addition, some beaches contain large fractions of shell fragments which have a lower density of about 2,400 kg/m^3.

The sediment porosity is a function of the sorting and roundness of the bed material. Approximate porosity values can be obtained from Table 2-70. A sorting classification is described in Table 2-91.

Table 2-70. Porosity of natural sand beds (modified from Soulsby 1997).

Packing	Well-sorted	Average	Poorly-sorted
Loosely packed	0.46	0.43	0.38
Average	0.42	0.4	0.33
Densely Packed	0.4	0.37	0.3

Table 2-71 gives a description, default value, and range for the sediment properties.

Table 2-71. CMS-Flow card used for specifying the global sediment properties.

Input	Format	Notes
Sediment Porosity	[card=SEDIMENT_POROSITY] [name=**SedPoro**, type=*float*, default=0.4, range=(0.3>SedPoro<0.46)]	Sets the sediment porosity
Sediment Density	[card=SEDIMENT_DENSITY] [name=**MorphStartTime**, type=*float*, default=2650.0, range=(**MorphStartTime**>=0)] [name=**SedDens**, type=*char*, options=**DensUnits**), default='kg/m^3']	Sets the sediment density in kg/m^3

Example 2-72. Specifying the global sediment properties.

```
SEDIMENT_DENSITY            2650.0 'kg/m^3'    !Units optional
SEDIMENT_POROSITY           0.4                !Dimensionless
```

Notes:

- The sediment density and porosity should NOT be used as calibration parameters.
- The sediment porosity and density are assumed constant for the whole domain and all grain size classes. For most coastal applications these assumptions are reasonable but need to be taken into consideration.

Sediment Size Class Properties

The transported sediment material is discretized into different groups each representing the sediments within a specific size range. These groups are referred to as sediment size classes and have the following properties.

1. Characteristic diameter
2. Fall velocity
3. Corey shape factor
4. Critical shear stress

In SMS 11.1 the sediment size class properties are specified with the *Size Class* section within the *Sediment* tab of the CMS-Flow Model Control window (see figure below). Sediment size class classes can be added or removed using the and buttons, respectively. At least one sediment size class should be specified.

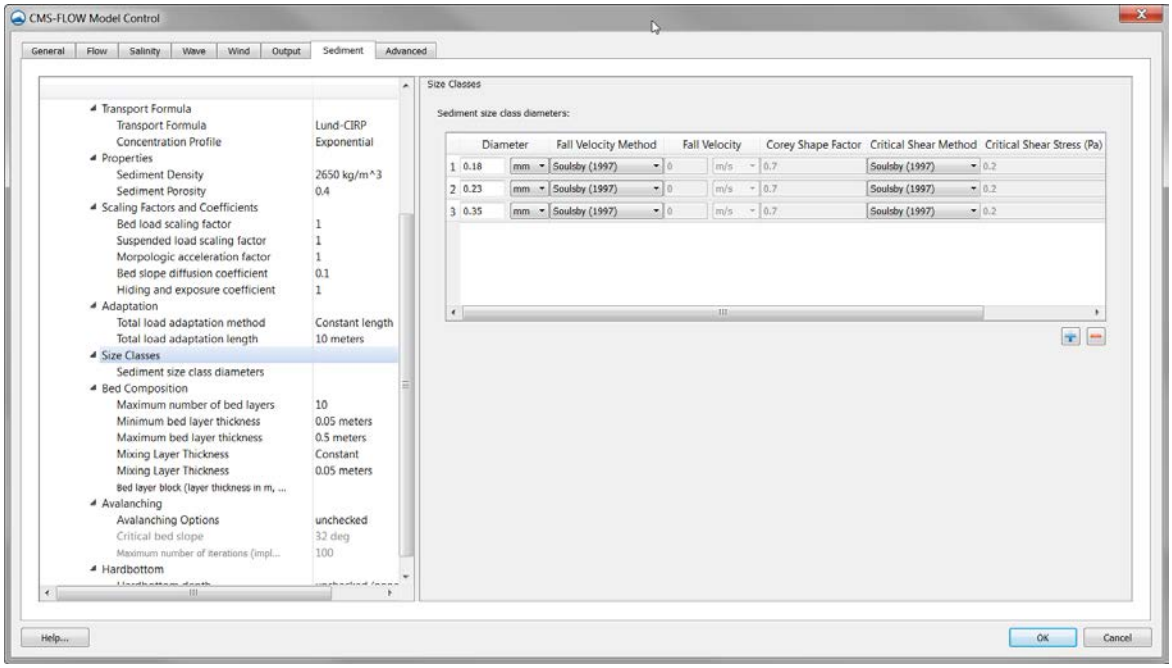


Figure 2-83. Sediment *Size Class* section within the *Sediment* tab of the *CMS-Flow Model Control* window in SMS 11.1.

Table 2-72. CMS-Flow card used for specifying the sediment size class block.

Card	Format	Notes
Begins a Sediment Size Class Block	[begin=SEDIMENT_SIZE_CLASS_BEGIN, name=**SedClass**, children=**SedClass**, CMS_version=(>4.10.00), SMS_version=(>11.1)]	Begins the sediment size block
Sediment Size Class Name	[card=NAME, parent=**SedClass**, SMS_version=none] [name=**SedClassName**, type=*char*]	Specifies a sediment size class name. Optional.
Ends a Sediment Size Class Block	[end=SEDIMENT_SIZE_CLASS_END, name=**SedClass**]	Ends the sediment size class block

Diameter

In the multiple-sized sediment transport model, the continuous grain sizes are divided into discrete bins or size classes. Each size class has a characteristic diameter, and lower and upper limits. The characteristic diameter is the diameter used in the sediment transport formula and to estimate other sediment size class properties if they are not specified. It is always required for all sediment size classes. The characteristic diameters d_k is related to the bin lower $d_{k-1/2}$ and upper $d_{k+1/2}$ bin limits by

$$d_k = \sqrt{d_{k-1/2} d_{k+1/2}} \tag{2-21}$$

The CMS-Flow card used to specify the size class characteristic diameter is described in the table below.

Table 2-73. CMS-Flow card used for specifying the sediment size class diameter.

Input	Format	Notes
Sediment Size Class Characteristic Diameter	[card=DIAMETER, parent=**SedClass**] [name=**diam**, type=*float*, default=none, range=(**diam**>0.06 'mm'),required=true] [name=**diamUnits**, type=*char*, options=**DistUnits**), default='mm']	Sediment size class characteristic diameter.

Notes:

- The size classes are constant for the whole domain.
- The fraction of each sediment size class in the bed describes the bed composition.
- Increasing the number of sediment size classes increases the computational time because each size class requires its own governing equations.
- If not size classes are specified, then a single transport grain size is used based on the mean of the median grain size diameter for the surface bed layer.

Recommendations:

- It is NOT recommended to use more 8-9 sediment size classes, because of the increased computational time. For most cases, 3-5 sediment size classes are sufficient.
- A good and simple way of estimating the grain size classed based only on the size limits of the distribution is by distributing the diameters logarithmically:

$$d_k = \exp\left[\ln d_1 + \ln(d_N / d_1)\frac{k-1}{N-1}\right] \qquad (2\text{-}22)$$

where N is the number of sediment size classes, k indicates the sediment size class, and the subscript indicates the size class number. This leads diameters more closely spaced in the finer grain sizes and more loosely spaced in the coarser grain sizes. Since the sediment transport is larger for the finer grain sizes, it makes sense to give more resolution near the finer grain sizes.

- The size class diameter should be chosen carefully so that encompass the whole range of sediment sizes found in the bed.

A Matlab example how to select the sediment size classes based on a median grain size and geometric standard deviation is given in Appendix E.

Fall Velocity

The user may select one of several sediment size class fall velocity formulas to calculate the fall velocity. It is noted that the fall velocity should not be used as a calibration parameter. There are three options for the sediment fall velocity:

1. User-specified
2. Soulsby (1997)
3. Wu and Wang (2006)

The options for the sediment fall velocity can be selected with within the *Size Class* section within the *Sediment* tab of the *CMS-Flow Model Control* window (see figure below).

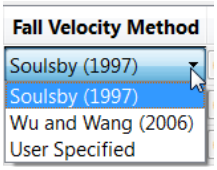


Figure 2-84. Sediment size class *Fall Velocity Method* options in the *Size Class* section within the *Sediment* tab of the *CMS-Flow Model Control* window in SMS 11.1.

The cards used to specify the sediment fall velocity are described in the table below.

Table 2-74. CMS-Flow cards used for specifying the sediment size class fall velocity.

Input	Format	Notes
Fall Velocity	[card=FALL_VELOCITY, parent=**SedClass**, exclusion=**wsform**] [name=**wsfall**, type=*float*, default=none, range=(**wsfall**>0.0 'm/s')] [name=**wsfallUnits**, type=*char*, , options=**VelUnits** default='m/s']	Sediment size class fall velocity.
Fall Velocity Formula	[card=FALL_VELOCITY_FORMULA, parent=**SedClass**] [name=**wsform**, type=*char*, options=(SOULSBY,WU_WANG), default=none]	Sediment size class fall velocity formula.
Corey Shape Factor	[card=COREY_SHAPE_FACTOR, parent=**SedClass**] [name=**Corey**, type=*float*, default=0.7, range=(0.2<**corey**<1.0), dependence=(**wsform**=WU_WANG)]	Sediment size class Corey Shape Factor. Used in the Wu and Wang (2006) sediment fall velocity formula.

The Soulsby (1997) formula is given by

$$\omega_s = \frac{\nu}{d}\left[\left(10.36^2 + 1.049 d_*^3\right)^{1/2} - 10.36\right] \quad (2\text{-}23)$$

where d_* is the dimensionless grain size

$$d_* = d_{50}\left[\frac{(s-1)g}{\nu^2}\right]^{1/3} \quad (2\text{-}24)$$

where d_{50} is the median grain size, s is the sediment specific gravity, g is the gravitational constant, ν is the kinematic viscosity.

The Wu and Wang (2006) formula includes the effect of the sediment shape through and is given by

$$\omega_s = \frac{M\nu}{Nd}\left[\sqrt{\frac{1}{4} + \left(\frac{4N}{3M^2} d_*^3\right)^{1/n}} - \frac{1}{2}\right]^n \quad (2\text{-}25)$$

where $M = 53.5\exp(-0.65F_S)$, $N = 5.65\exp(-2.5F_S)$, $n = 0.7 + 0.9F_S$. Here F_S is the Corey shape factor defined as $F_S = d_S / \sqrt{d_I d_L}$ in which d_S, d_I, and d_L are the diameters of the short, intermediate and long mutually perpendicular axes. Naturally worn quartz sands have a typical Corey shape factor of 0.7 and calcareous sand of about 0.55.

Notes:

- For noncohesive sediments, it is NOT recommended to use the sediment fall velocity, or shape factor as calibration parameters. These parameters should be estimated using field or literature data. If not measurements are available then the default formula should be used.
- For high sediment concentrations, the sediment fall velocity can be reduced. However, because the effects are important for high concentrations, the effect can usually be ignored.
- Decreasing the shape factor decreases the sediment fall velocity.

Critical Shear Stress

When using the Lund-CIRP or Watanabe transport formula, the option is given to modify the critical shear stress for incipient motion. In the case of the Soulsby-van Rijn, and van Rijn transport formula, the depth-averaged critical velocities are used and cannot be modified by the user. Therefore this section is only applicable to the Lund-CIRP and Watanabe transport formula. There are three options for the critical shear stress for incipient motion:

4. User-Specified
5. Soulsby (1997)
6. Wu and Wang (1999)

The options for the sediment size class critical shear stress can be selected with within the *Size Class* section within the *Sediment* tab of the *CMS-Flow Model Control* window (see figure below).

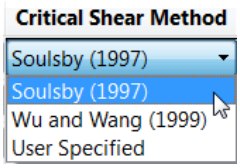


Figure 2-85. Sediment size class *Critical Shear Method* options in the *Size Class* section within the *Sediment* tab of the *CMS-Flow Model Control* window in SMS 11.1.

The cards used to specify the sediment size class critical shear stress are described in the table below.

Table 2-75. CMS-Flow cards used for specifying the critical shear stress.

Input	Format	Notes
Critical Shields Parameter	[card=SHIELDS_PARAMETER, parent=**SedClass**] [name=**thetacr**, type=*float*, default=none, range=(**thetacr**>0.0)]	Sediment size class characteristic diameter.
Critical Shear Stress	[card=(CRITICAL_SHEAR,CRITICAL_STRESS, parent=**SedClass**)] [name=**taucr**, type=*float*, default=none, range=(**taucr**>0.0)] [name=**taucrUnits**, type=*char*, options=**PresUnits**, default='Pa']	Sediment size class characteristic critical shear stress.
Critical Shear Stress Formula	[card=(CRITICAL_SHEAR_FORMULA, CRITICAL_STRES_FORMULA)]] [name=**wsform**, type=*char*, options=(SOULSBY,WU_WANG), default=SOULSBY]	Sediment size class characteristic critical shear stress formula.

The critical shear stress, τ_{cr}, is related to the Shields parameter, Θ_{cr}, by

$$\frac{\tau_{cr}}{g(\rho_s-\rho)d}=\Theta_{cr} \tag{2-26}$$

Soulsby (1997) proposed the following formula for the Shields parameter

$$\Theta_{cr}=\frac{0.3}{1+1.2d_*}+0.055\left[1-\exp\left(-0.02d_*\right)\right] \tag{2-27}$$

The formula by Wu and Wang (1999) is given by

$$\Theta_{cr}=\begin{cases}0.126d_*^{-0.44} & \text{for } d_*<1.5\\ 0.131d_*^{-0.55} & \text{for } 1.5\le d_*<10\\ 0.0685d_*^{-0.27} & \text{for } 1.5\le d_*<20\\ 0.0173d_*^{0.19} & \text{for } 1.5\le d_*<40\\ 0.0115d_*^{0.30} & \text{for } 1.5\le d_*<150\\ 0.052 & \text{for } 150\le d_*\end{cases} \tag{2-28}$$

Notes:

- For noncohesive sediments, it is NOT recommended to use the sediment fall velocity, or shape factor as calibration parameters. These parameters should be estimated using field or literature data. If not measurements are available then the default formula should be used.

Examples

In the example below a single sediment size is specified. Since there is only one sediment size class, the bed sorting and gradation is not considered.

Example 2-73. Single-size sediment transport size class specification.

```
SEDIMENT_SIZE_CLASS_BEGIN
   DIAMETER       0.2 'mm'        !Positive is upwards
SEDIMENT_SIZE_CLASS_END
```

In the following example three sediment size classes are considered which allows the model to track the bed composition. It is noted that not the same input parameters have to be specified for all sediment size classes. Any parameters which are not specified are set the default value.

Example 2-74. Multiple-sized sediment transport size class specification.

```
SEDIMENT_SIZE_CLASS_BEGIN
   DIAMETER       0.2 'mm'        !Positive is upwards
   !All other values left as default
SEDIMENT_SIZE_CLASS_END
SEDIMENT_SIZE_CLASS_BEGIN
   DIAMETER                0.3 'mm'        !Positive is upwards
   FALL_VELOCITY           0.043 'm/s'   !measured
   CRITICAL_SHEAR          0.25 'Pa'
SEDIMENT_SIZE_CLASS_END
SEDIMENT_SIZE_CLASS_BEGIN
   NAME                    "Sand 3" !Optional
   DIAMETER                0.4 'mm'  !Units option, card required
   FALL_VELOCITY_FORMULA   SOULSBY  !{SOULSBY} | WU_WANG, card optional
   CRITICAL_SHEAR_FORMULA  SOULSBY  !{SOULSBY} | WU_WANG, card optional
SEDIMENT_SIZE_CLASS_END
```

Transport Model

There are currently three sediment transport models available in CMS: (1) Equilibrium total load, (2) Equilibrium bed load plus advection-diffusion for suspended load, and (3) Non-equilibrium total load. The first two models are selected by unchecking the checkbox which says "Use non-equilibrium transport" and selecting either "Total load" for the first model, or "Advection-diffusion" for the second next to input item named "Formulation". The third model is selected by checking the box "Use non-equilibrium transport".

Table 2-76gives a description, default value, and range for the sediment transport models.

Table 2-76. CMS card used to specify the sediment transport model.

Card	Format	Notes
Sediment Transport Model	[card=SED_TRANS_FORMULATION] [name=**SedModel**, type=*char*, options=(EXNER,A-D,NET), default=NET]	Specifies the sediment transport model

Example 2-75. Specifying the sediment transport model.

```
SEDIMENT_TRANS_FORMULATION     NET       !EXNER | A-D | NET
```

Notes:

- When selecting the equilibrium total load model, the SED_TRAN_FORMULATION card is set to either WATANABE or LUND_CIRP depending on the transport formula chosen. When selecting the equilibrium A-D model, the transport formula is specified through the concentration profile formula (described below).
- All three sediment transport models are available with the explicit solver, while only the NET is available only with the implicit time stepping scheme.
- Only the Nonequilibrium Total Load model is available in CMS versions 4.0 and greater at the moment. Plans are under way to include the Equilibrium Bed plus Advection-Diffusion Suspended Load model as an option, but the Equilibrium Total Load model will likely be discontinued.

A description of each sediment transport model is described in the sections below.

Equilibrium Total load

In this model, both the bed load and suspended load are assumed to be in equilibrium. The bed change is solved using a simple mass balance equation known as the Exner equation.

Equilibrium Bed load plus Advection-Diffusion Suspended Load

Calculations of suspended load and bed load are conducted separately. The bed load is assumed to be in equilibrium and is included in the bed change equation while the suspended load is solved through the solution of an advection-diffusion equation. Actually the advection diffusion equation is a non-equilibrium formulation, but because the bed load is assumed to be in equilibrium, this model is referred to the "Equilibrium A-D" model.

Nonequilibrium Total Load

In this approach, neither the bed nor suspended loads are assumed to be in equilibrium. The suspended- and bed-load transport equations are combined into a single equation and thus there is one less empirical parameter to estimate (adaptation length).

Sediment Transport Formula

The near bed sediment concentration or concentration capacity are calculated with one of the following transport formula:

- Lund-CIRP (2006)
- Van Rijn (1998)
- Watanabe (1987)
- Soulsby-van Rijn (1997) (>=V4.0)

Table 2-77 gives a description, default value, and range for the sediment transport formulae.

Table 2-77. CMS-Flow cards used to specify the sediment transport formula and related parameters and options.

Card	Arguments	Notes
Sediment Transport Formula	[cards=(TRANSPORT_FORMULA, NET_TRANSPORT_CAPACITY), versions=(<4.00.00)]] [name=**TransForm**, type=*char*, default=*none*, options=(LUND-CIRP, VAN_RIJN,SOULSBY,WATANABE)]	Selects the transport formula. Note that SOULSBY is only available in v>=4.0
Concentration Profile	[cards=CONCENTRATION_PROFILE] [name=**ConcPro**, type=*char*, options=(LUND-CIRP,VAN_RIJN, EXPONENTIAL,ROUSE), default=EXPONENTIAL]	Selects the concentration profile to be used either in the equilibrium A-D or NET models. In the A-D model, it is used to estimate the near bed concentration, whereas in the NET, it is used in the total-load correction factor.
Watanabe coefficient	[cards=A_COEFFICIENT_WATANABE] [name=**SedWatA**, type=*float*, default=0.1, range=(0.05-<**SedWatA**<0.5)] [name= **SedTransDtUnits**, type=*char*, optional=true, options=**TimeUnits**), default='s']	Empirical coefficient which goes into the Watanabe transport formula.

Example 2-76. Specifying the sediment transport formula, concentration profile, and Watanabe coefficient.

```
TRANSPORT_FORMULATION    WATANABE       !LUND-CIRP | VAN_RIJN | etc.
CONCENTRATION_PROFILE    EXPONENTIAL    !ROUSE | EXPONENTIAL
A_WATANABE_COEFFICIENTE  0.2
```

Notes:

- Different transport formula may produce very different results in morphology change.
- The Lund-CIRP does well in predicting the surf zone sediment transport but tends to overestimate the transport rates near the wetting and drying limit and in deep water (>10 m).
- The van Rijn transport formula tends to underestimate the transport for conditions near the critical shear stress of motion. The formula also tends to underestimate the transport close to the shoreline.
- The Watanabe formula tends to underestimate the transport in deep water (>10 m).

Recommendations:

- For every sediment transport project, it is recommended to test several sediment transport formula to observe the sensitivity of the results to the sediment transport formula. The sediment transport formula is perhaps the most important parameter of any sediment transport model and the largest source error.

Scaling Factors

Transport Scaling Factors

The bed- and suspended-load transport scaling factors multiply directly by the transport capacity or near-bed sediment concentration calculated from the transport formula. The table below gives a description, default value, and range for the bed- and suspended-load scaling factors.

Table 2-78. CMS-Flow card used for specifying the global sediment properties.

Input	Format	Notes
Bed Load Scaling Factor	[card=BED_LOAD_SCALE_FACTOR] [name=**SedBedScale**, type=*float*, default=1.0, typical=(0.5><**SedBedScale**<2.0)]	Calibration factor for bed load transport capacity formula
Suspended Load Scaling Factor	[card=BED_LOAD_SCALE_FACTOR] [name=**SedSuspScale**, type=*float*, default=1.0, typical=(0.5><**SedSuspScale**<2.0)]	Calibration factor for suspended load transport capacity formula

Example 2-77. Specifying the sediment transport scaling factors.

```
BED_LOAD_SCALE_FACTOR     0.7  !default = 1.0
SUSP_LOAD_SCALE_FACTOR    0.8  !default = 1.0
```

Note:

- The transport scaling factors should be one of the first parameters used in calibrating a sediment transport model. The scaling factors can be calibrated using measured sediment transport rates or morphology change. In general, larger scaling factors will produce larger bed change.
- Due to the large uncertainty in the transport formula, it is generally common to use scaling factors in the range of 0.5-2.0. However values between 0.2 and 5.0 are not unreasonable given the uncertainty in the equilibrium sediment transport formula.
- For simplicity, the same transport scaling factors are applied for the whole solution domain and for all sediment sizes.

Morphologic Acceleration Factor

The morphologic scaling factor is directly multiplied by the calculated bed change at every time step and is intended as a means of speeding up the computational time. It is only recommended for periodic boundary conditions or conditions that do not change rapidly over time. Table 2-79 gives a description, default value, and range for the morphologic acceleration factor.

Table 2-79. CMS card used to specify the morphology acceleration factor.

Input	Format	Notes
Morphologic Acceleration Factor	[card=MORPH_ACCEL_FACTOR] [name=**MorphFac**, type=*float*, default=1.0, range=(1.0><**MorphFac**<100)]	Morphologic acceleration factor. Directly multiplies by calculated bed change.

Example 2-78. Specifying the morphologic acceleration factor.

```
MORPH_ACCEL_FACTOR     2.0   !default is 1.0
```

Note:

- The morphologic scaling factor is NOT a calibration parameter. It should only be used in cases with periodic forcing and boundary conditions and even then it should be used with caution. It is NOT recommended to use larger values than 20-30.
- The acceleration factor may be used to turn off the morphology change by setting it to zero.

Schmidt Number

The sediment mixing coefficient is calculated as the eddy viscosity divided by the Schmidt number. For simplicity the Schmidt number is assumed to be constant and the default value is 1.0. Table 2-80 gives a description, default value, and range for the Schmidt number.

Table 2-80. CMS card used to specify the Schmidt number

Input	Format	Notes
Schmidt Number	[card=SCHMIDT_NUMBER] [name=**Schmidt**, type=*float*, default=1.0, typical=(0.3>**Schmidt**<2.0)]	Controls the sediment mixing strength

The example below demonstrates how the Schmidth number is specified in the CMS-Flow Model Control file.

Example 2-79. Specifying the Schmidt number.

```
SCHMIDT_NUMBER     1.5   !default is 1.0, OPTIONAL
```

Note:

- The Schmidt number should NOT be used as a calibration number and should only be changed in sensitivity analysis or model testing.

Adaptation Coefficient

The adaptation coefficient is an important parameter to consider in setting up the CMS sediment transport model. The sensitivity of results to the adaptation coefficient depends on the spatial and temporal scales of the problem. For example, if a high resolution grid is used to model short-term dynamics of a nearshore disposal site or a small inlet, then the adaptation coefficient is likely to be important. However, if a relatively coarse grid is used to study sediment pathways in a large estuary entrance, then the adaptation coefficient will not be important.

Total Load

There are four methods for calculating the total load adaptation coefficient in CMS:

1. Constant length
2. Constant time
3. Maximum of bed and suspended load lengths
4. Weighted average of bed and suspended load lengths

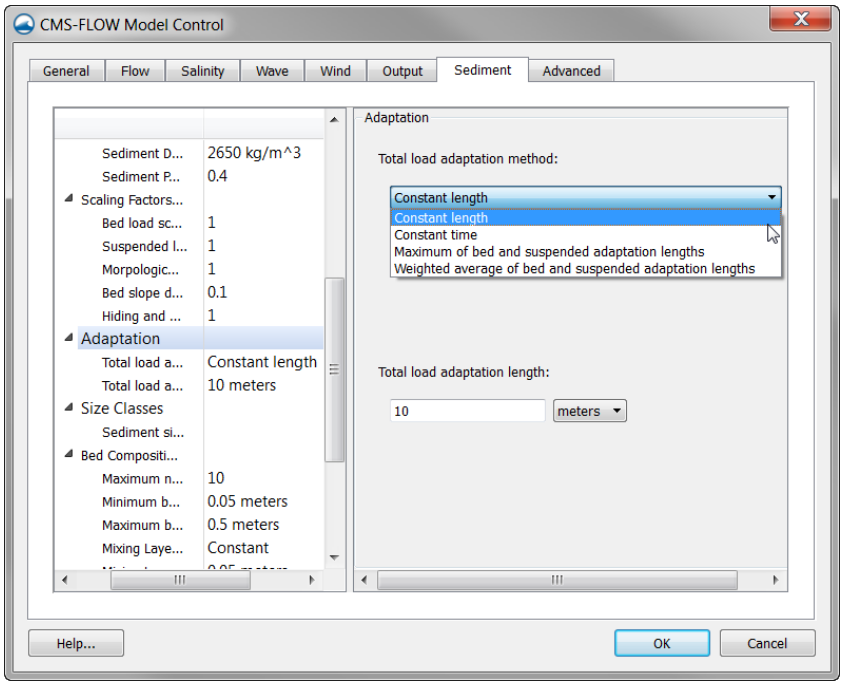


Figure 2-86. *Adaptation* section within the *Sediment* tab of the *CMS-Flow Model Control* window in SMS 11.1.

Constant Length

A temporally and spatially total load adaptation length is used for the whole domain. The total load adaptation coefficient is calculated as

$$\alpha_t = Uh / (L_t \omega_s) \tag{2-29}$$

where ω_s is the sediment fall velocity of the transport grain size, for single size sediment transport, or the median grain size, in the case of multiple-sized sediment transport.

Constant Time

A temporally and spatially constant total-load adaptation time is used for the whole domain. The total load adaptation coefficient is calculated as

$$\alpha_t = h / (T_t \omega_s) \tag{2-30}$$

where ω_s is the sediment fall velocity of the transport grain size for single size sediment transport, or the median grain size, in the case of multiple-sized sediment transport.

Maximum of Bed and Suspended Load Lengths

A temporally and spatially constant total-load adaptation time is used for

$$L_t = \max(L_s, L_b) \tag{2-31}$$

where L_s is the suspended load adaptation length and L_b is the bed load adaptation length. The methods for determining L_b and L_s are described in subsequent sections.

Weighted Average of Bed and Suspended Load Lengths

A temporally and spatially constant total-load adaptation time is used for

$$L_t = r_s L_s + (1 - r_s) L_b \tag{2-32}$$

where L_s is the suspended load adaptation length, L_b is the bed load adaptation length, and r_s is the fraction of suspended load of the total load.

The methods for determining L_b and L_s are described in subsequent sections.

The CMS cards related to the total load adaptation coefficient are described in the table below.

Table 2-81. CMS card used to specify the total-load adaptation coefficient.

Input	Format	Notes
Total-load Adaptation Method	[cards=ADAPTATION_METHOD_TOTAL, versions=(<4.00.00)]] [name=**SedAdaptMethTotal**, type=*char*, default=*none*, options=(CONSTANT_LENGTH, CONSTANT_TIME,WGHT_AVG_BED_SUSP_LENGTH, MAX_BED_SUSP_LENGTH)]	Specifies the total-load adaptation coefficient method
Total-load Adaptation Length	[cards=ADAPTATION_LENGTH_TOTAL] [name=**SedAdaptLenTotal**, type=*real*, default=10.0, range=(0.05-<**SedAdaptLenTotal**<0.5)] [name=**SedAdaptLenTotalUnits**, type=*char*, optional=true, options=**DistUnits**), default='m', dependencies=(**SedAdaptMethTotal=** (CONSTANT_LENGTH, WGHT_AVG_BED_SUSP_LENGTH, MAX_BED_SUSP_LENGTH))]	Specifies the total-load adaptation length
Total-load Adaptation Time	[cards=ADAPTATION_TIME_TOTAL] [name=**SedAdaptLenTotal**, type=*real*, default=10.0, range=(0.05-<**SedAdaptLenTotal**<0.5)] [name=**SedAdaptLenTotalUnits**, type=*char*, optional=true, options=**DistUnits**), default='s', dependencies= (**SedAdaptMethTotal=**CONSTANT_TIME)]	Specifies the total-load adaptation time

Note:

- If the total-load adaptation length is set to -1.0, then the maximum of the bed- and suspended-load adaptation lengths is used. This is the input format used by SMS.

Recommendations:

- When first setting up a sediment transport model, it is recommended to use a spatially constant total load adaptation length for simplicity.

- If the user is not sure of the sensitivity of the results to the adaptation length for the specific project, then it is recommended to test different total load adaptation lengths; for example 5, 10, and 50 meters. This will provide an insight into the sensitivity of the results to the adaptation length. In many cases, the results are not found to be sensitive and the default value of 10 m can be used. If the results are found to be more tests are necessary in determining to optimal method and parameters for calculating the adaptation length.
- WGHT_AVG_BED_SUSP_LENGTH is the most physically accurate method for determining the total load adaptation length. However, it can lead to relatively small adaptation lengths which cause instabilities, especially for large computational time steps. If this occurs then it is recommended to use MAX_BED_SUSP_LENGTH.

Bed Load

The bed load adaptation parameters only need to be specified if the total-load adaption coefficient is set as a function of the bed load adaptation length. The CMS cards related to the bed load adaptation length are described in the table below.

Table 2-82. CMS card used to specify the bed-load adaptation coefficient.

Input	Format	Notes
Bed-load Adaptation Method	[cards=ADAPTATION_METHOD_BED, versions=(<4.00.00)]] [name=**SedAdaptMethBed**, type=*char*, default=*none*, options=(CONSTANT_LENGTH, CONSTANT_TIME,DEPTH_DEPENDANT), dependencies=(**SedAdaptMethTotal**= (WGHT_AVG_BED_SUSP_LENGTH, MAX_BED_SUSP_LENGTH)]	Specifies the bed-load adaptation coefficient method.
Bed-load Adaptation Length	[cards=ADAPTATION_LENGTH_BED] [name=**SedAdaptLenBed**, type=*real*, default=10.0] [name=**SedAdaptLenBedUnits**, type=*char*, optional=true, options=**DistUnits**), default='m', dependencies=(**SedAdaptMethBed=** CONSTANT_LENGTH)]	Specifies the bed-load adaptation length.
Bed-load Adaptation Time	[cards=ADAPTATION_TIME_TOTAL] [name=**SedAdaptTimeBed**, type=*real*, default=10.0] [name=**SedAdaptTimeBedUnits**, type=*char*, optional=true, options=**DistUnits**), default='s', dependencies= (**SedAdaptMethBed=**CONSTANT_TIME)]]	Specifies the bed-load adaptation time
Bed-load Adaptation Length Factor	[cards=ADAPTATION_DEPTH_FACTOR_BED] [name=**SedAdaptLenFacBed**, type=*real*, default=10.0] [name=**SedAdaptLenFacBedUnits**, type=*char*, optional=true, options=**DistUnits**), default='s', dependencies=(**SedAdaptMethTotal**= (WGHT_AVG_BED_SUSP_LENGTH, MAX_BED_SUSP_LENGTH)]]	Specifies the bed-load adaptation length factor which multiplies by the local total water depth.

Note:

- If the total-load adaptation length is set to a negative number, then the value is interpreted as being the bed-load adaptation length depth factor. The bed-load adaptation length is therefore calculated as the depth times the positive depth factor.

Recommendations:

- Based on experience, it is recommended to use a constant bed load length of at least one to three times the grid resolution.

Suspended Load

The suspended load adaptation parameters only need to be specified if the total load adaption coefficient is set as a function of the suspended-load adaptation length. The CMS cards related to the suspended-load adaptation coefficient are described in the table below.

Table 2-83. CMS card used to specify the suspended-load adaptation coefficient.

Input	Format	Notes
Suspended-load Adaptation Method	[cards=ADAPTATION_METHOD_SUSPENDED, versions=(<4.00.00)]] [name=**SedAdaptMethSusp**, type=*char*, default=*none*, options=(CONSTANT_LENGTH, CONSTANT_TIME,CONSTANT_ALPHA, ARMANINI_DISILVIO,LIN,GALLAPPATTI), dependencies=(**SedAdaptMethTotal**= (WGHT_AVG_BED_SUSP_LENGTH, MAX_BED_SUSP_LENGTH))]	Specifies the suspended-load adaptation coefficient method.
Suspended-load Adaptation Length	[cards=ADAPTATION_LENGTH_SUSPENDED] [name=**SedAdaptLenSusp**, type=*real*, default=10.0] [name=**SedAdaptLenSuspUnits**, type=*char*, optional=true, options=**DistUnits**), default='m', dependencies=(**SedAdaptMethSusp=** CONSTANT_LENGTH)]	Specifies the suspended-load adaptation length.
Suspended-load Adaptation Time	[cards=ADAPTATION_TIME_SUSPENDED] [name=**SedAdaptTimeSusp**, type=*real*, default=10.0] [name=**SedAdaptTimeSuspUnits**, type=*char*, optional=true, options=**DistUnits**), default='s', dependencies=(**SedAdaptMethSusp=** CONSTANT_TIME)]]	Specifies the suspended-load adaptation time
Suspended-load Adaptation Coefficient	[cards=ADAPTATION_COEFFICIENT_SUSPENDED] [name=**SedAdaptCoeffSusp**, type=*real*, default=10.0] [name=**SedAdaptCoeffSuspUnits**, type=*char*, optional=true, options=**DistUnits**), default='s', dependencies=(**SedAdaptMethSusp=** CONSTANT_ALPHA)]]	Specifies the suspended-load adaptation Coefficient

Notes:

- By default, SMS writes the CMS-Flow card ADAPTATION_COEFF_SUSPENDED. However, this card is outdated and can be confused with the card ADAPTATION_COEFFICIENT_SUSPENDED (see table above). Instead, it is recommended for the user to use the ADAPTATION_METHOD_SUSPENDED to specify the method for the suspended-load adaptation coefficient.

Examples

The first example shown below demonstrates the simplest method for specifying the total load adaptation length which is with a constant value. It is noted that if the card ADAPTATION_LENGTH_TOTAL is specified, then the total load adaptation method is automatically set to a constant length. The way this option is specified in the SMS 11.1 interface is shown in Figure 2-86.

Example 2-80. Constant Total-load Adaptation Length.

```
ADAPTATION_METHOD_TOTAL     CONSTANT_LENGTH  !optional
ADAPTATION_LENGTH_TOTAL     10.0 'm'
```

The following example uses a more complicated formulation for calculating the total load adaptation method. The total load adaptation length is calculated as the weighted average of the bed and suspended loads. The average weights for the bed and suspended loads are given by the fraction of each load relative to the total load. The bed load adaptation length is calculated as a function of the water depth ($L_b = 7h$) while the suspended load adaptation length is calculated using the formulation of Armanini and di Silvio (1986). The figure below shows how to setup the adaptation parameters in the SMS 11.1 interface for this example.

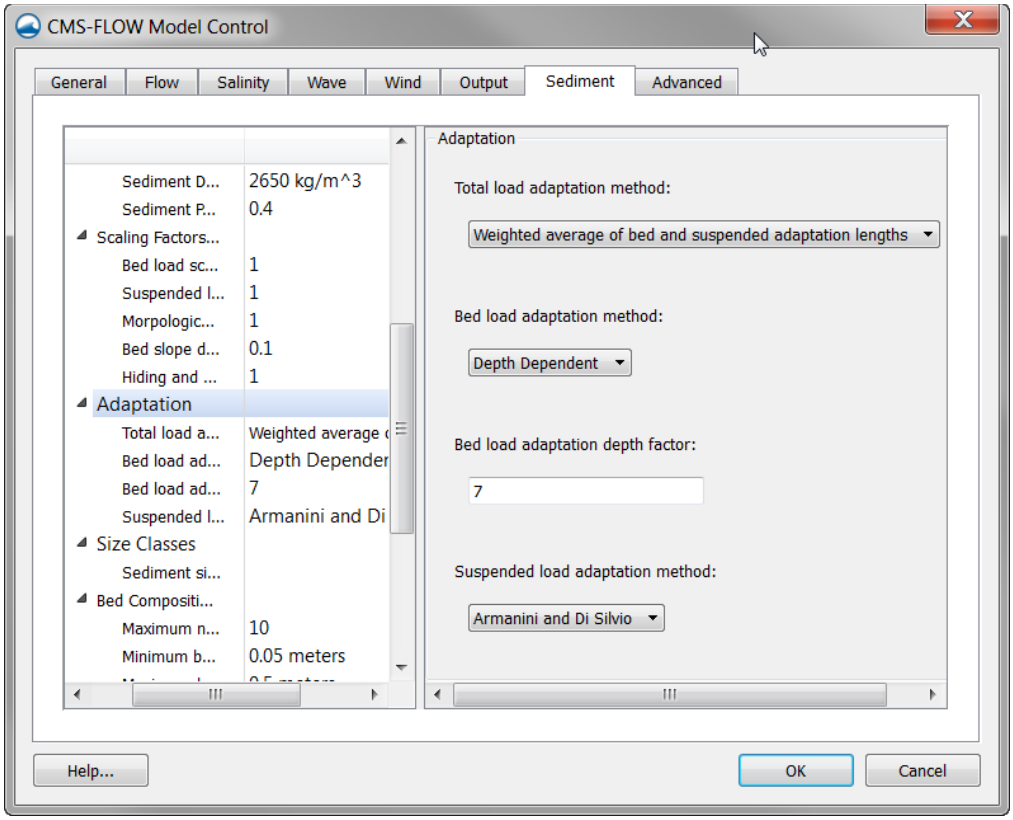


Figure 2-87. Example of the total load a*daptation* length specification within the *Adaptation* section within the *Sediment* tab of the *CMS-Flow Model Control* window in SMS 11.1.

Example 2-81. Total load adaptation length as the weighted averaged of the bed and suspended load lengths.

```
ADAPTATION_METHOD_TOTAL          WGHT_AVG_BED_SUSP_LENGTH
ADAPTATION_METHOD_BED            DEPTH_DEPENDENT
ADAPTATION_DEPTH_FACTOR_BED      7.0 !dimensionless
ADAPTATION_METHOD_SUSPENDED      ARMANINI_DISILVIO
```

Hard Bottom Specification

Although the hard bottom dataset can be edited (when it's the active dataset) by selecting a cell (or group of cells) and changing the scalar (S) value in the Edit Window, an user-friendly window exists which provides specification options. With the *Select Grid Cell* tool active, make a selection, right click to bring up the tool menu and choose the *Specify Hard Bottom...* option. This will open the CMS-Flow Hard Bottom Specification window.

The following options are provided in the Hard Bottom Specification window:

- **Use bathymetric cell depth** - Sets the cell hard bottom depth to be the cell geometry value thereby creating an exposed non-erodible condition. If multiple cells were selected, then each cell will use its respective bathymetric depth.
- **Specified distance below bathymetric cell depth** - Sets the cell hard bottom depth to be the cell geometry value plus the specified distance thereby creating a sediment-covered non-erodible condition. The distance is limited to positive values to ensure the hard bottom depth is greater than the geometry value. The cell can provide sediment for transportation, however, the amount of erosion is limited. If multiple cells were selected, then each cell will use its respective bathymetric depth.
- **Specified depth** - Sets the cell hard bottom depth to the specified depth thereby creating a sediment-covered non-erodible condition similar to specified distance. The depth is limited to greater than the geometry value. If multiple cells were selected, then the depth is limited to greater than the largest geometry value and all cells will have the same value.
- **Unspecified** - Resets to an inactive hard bottom condition. The cell hard bottom depth is set to the CMS-Flow null value. If multiple cells were selected, then all cells will be reset.

If no cells are selected when opening the Hard Bottom Specification window, then all computational (ocean) cells will be used. If a selection of only non-computational cells, then specification cannot occur. If a selection contains computational and non-computational cells, then the specification will only apply to the computational cells.

If multiple computational cells with differing specifications are selected, the window will not display a selected specification type and the *OK* button will be disabled. This is to protect the previous specifications from being

overwritten by mistake. The *OK* button will be enabled when an option is selected. The minimum hard bottom depth of the multiple computational cells selected will be displayed in the *Depth* edit field and the minimum hard bottom depth minus the maximum geometry depth of the multiple computational cells selected will be displayed in the *Distance* edit field.

Display Options

The hard bottom dataset (when it's the active dataset) will only display the cells with hard bottom specified if the *Ocean cell* display option is turned on. Inactive hard bottom cells are not displayed.

CMS-Flow includes hard bottom symbols to differentiate specifications. On the *Cartesian Grid* page of the Display Options window (when CMS-Flow is the active model), the *Hard bottom symbols* check box controls the display of symbols that will appear in hard bottom cells (even if the hard bottom dataset is not active). If this is turned on, then the user must be aware of the individual symbol settings accessed by clicking on the *Options...* button. The *Options...* button displays the CMS-Flow Hard Bottom Symbols window.

Hard bottom symbols can be selected for three hard bottom specification types:

- **Non-erodible** - Displayed in exposed hard bottom cells (cell hard bottom depth is equal to cell bathymetric depth).
- **Erodible to specified depth** - Displayed in sediment-covered hard bottom cells (cell hard bottom depth is greater than cell bathymetric depth).
- **Invalid specification** - Displayed in hard bottom cells where the hard bottom depth is less than cell bathymetric depth (the geometry is below the erosion limit).

If the *Hard bottom symbols* check box is turned off, no symbols will be displayed and the individual settings cannot be accessed, however, the individual settings will not be changed.

Bed Slope Term

The bed slope term accounts for the effect of gravity on sloped beds. The larger the bed slope coefficient, the more sediment tends to move downslope, thus smoothing the solution. The CMS-Flow used to specify the slope coefficient is described in the table below. The bed slope coefficient is set in the *Sediment* tab of the *CMS-Flow Model Control* window in SMS 11.0.

Table 2-84. CMS card used to *specify the sediment bed slope diffusion coefficient*

Input	Format	Notes
Sediment Bed Slope Diffusion Coefficient	[card=SLOPE_COEFFICIENT] [name=**SedBedSlopeCoeff**, type=*float*, default=0.1, range=(0.0>=**SedBedSlopeCoeff**<5.0), typical=(0.1>**SedBedSlopeCoeff**<1.0)]	Sets the bed slope coefficient which controls enters a diffusion term which moves sediment down slope

Example 2-82. Specifying the sediment bed slope coefficient.

```
SLOPE_COEFFICIENT        0.1   !Dimensionless
```

Note:

- A constant bed slope coefficient is used spatially, temporally and for all sediment size classes.
- Increasing the bed slope coefficient will increase the smoothing of the bed.
- The measured morphology change can be used to calibrate this parameter. However, its effect is less significant than the capacity transport formula and scaling factors.

Total-Load Correction Factor

The total-load correction factor accounts for the nonuniform vertical profile of sediment concentration and current velocity and produces temporal lag between the flow and sediment transport. The factor is used in the nonequilibrium total load sediment transport formula.

Table 2-85. CMS-Flow cards related to the total load correction factor.

Input	Format	Notes
Total load Correction Factor	[card=TOTAL_LOAD_CORR_FACTOR_CONSTANT, SMS_ver=none, CMS_ver=(>4.00.00)] [name=**SedTotalCorrFac**, type=*float*, default=none, optional=true, typical=(0.3><**SedTotalCorrFac**<1.0)]	Sets the total load correction factor to a constant.
Concentration Profile	[cards=CONCENTRATION_PROFILE] [name=**ConcPro**, type=*char*, options=(LUND-CIRP,VAN_RIJN, EXPONENTIAL,ROUSE), default=EXPONENTIAL, exlusion=**SedTotalCorrFac**]	Selects the concentration profile to be used either in the equilibrium A-D or NET models. In the A-D model, it is used to estimate the near bed concentration, whereas in the NET, it is used in the total-load correction factor.

Example 2-83. Specifying a constant total-load correction factor.

```
TOTAL_LOAD_CORR_FACTOR_CONSTANT        0.7   !Dimensionless
```

Note:

- The total load correction factor is only used for the total-load nonequilibrium transport model (NET).
- Computing the total-load correction factor based on the sediment concentration and current velocity profiles increases slightly the computational time but is almost negligible.

Recommendations:

- It is NOT recommended to use the total-load correction factor as a calibration factor.

Bed Composition

Single-sized Sediment Transport

In the case of single-sized sediment transport the bed is usually assumed to be uniform and variations in the bed composition are ignored. In CMS, the option is provided to use a unique approach which the bed composition is allowed to be different from the single transported grain size and the bed composition is considered in hiding and exposure corrections to the sediment transport. The approach works well for cases where the bed is uniform and well sorted with local patches of coarse material such as shell hash. In this approach the bed composition is specified as a median grain size and optionally the 90th percentile diameter. Since only one sediment size class is transported, the bed composition cannot change and the input bed composition is assumed to be constant over time. Therefore, the approach does not consider armoring or downstream fining. In addition, since only the surface bed composition is used, the there is no need to specify the bed layering. The advantage of the single-sized sediment transport mode is that it is much less computationally and data intensive compared to the multiple-sized sediment transport model.

Table 2-86. CMS-Flow cards used to specify the composition for single-sized sediment transport.

Input	Format	Notes
Begins a sediment size class block	`[begin=BED_LAYER_BEGIN, block=BedLay, children=BedLayNum]`	Begins a bed layer block.

Bed layer number	`[card=(LAYER,LAYER_NUMBER)]`
	`[name=BedLayNum, type=float, default=none, range=(diam >0.06 'mm')]`
	`[name=SedDens, type=char, options=DensUnits), default='mm']`
		Specifies the bed layer number from the surface. If not specified then set the sequential bed layer block number.

50th percentile diameter dataset	`[cards=D50_DATASET, parent=BedLay]`
	`[name=d50file, type=char, example= "Flow_grid.h5"]`
	`[name=d50path, type=char, example= "Flow/Datasets/D50 2"]`	

90th percentile diameter dataset	`[cards=D90_DATASET, parent=BedLay]`
	`[name=d90file, type=char, example= "Flow_grid.h5"]`
	`[name=d90path, type=char, example= "Flow/Datasets/D90 2"]`	

Ends a sediment size class block	`[end=BED_LAYER_END, block=BedLay]`	Ends a bed layer block

Example 2-84. Bed composition specification for a single-sized sediment transport simulation with D50 dataset only.

```
!Notes: The bed composition is OPTIONAL for single-sized sediment transport
!If D90 is not specified it is set to 2.5*D50
D50_DATASET           "Flow_grid.h5" "FlowGrid/Datasets/D50"
```

Example 2-85. Bed composition specification for a single-sized sediment transport simulation with D50 and D90 datasets.

```
!Note: The bed composition is OPTIONAL for single-sized sediment transport
D50_DATASET           "Flow_grid.h5" "FlowGrid/Datasets/D50"
D90_DATASET           "Flow_grid.h5" "FlowGrid/Datasets/D90"
```

Example 2-86. Bed composition specification for a single-sized sediment transport simulation with a bed layer block and D50 dataset only.

```
!Notes: Only one layer can to be specified and is optional
!If D90 is not specified it is set to 2.5*D50
BED_LAYER_BEGIN !Block optional since there is only one layer
  D50_DATASET             "Flow_grid.h5" "FlowGrid/Datasets/D50 1"
BED_LAYER_END
```

Example 2-87. Bed composition specification for a single-sized sediment transport simulation with a bed layer block and D50 and D90 datasets.

```
!Note: Only one layer can to be specified and is optional
BED_LAYER_BEGIN !Block optional since there is only one layer
  D50_DATASET             "Flow_grid.h5" "FlowGrid/Datasets/D50 1"
  D90_DATASET             "Flow_grid.h5" "FlowGrid/Datasets/D90 1"
BED_LAYER_END
```

Note:

- In the case of the single-sized sediment transport model, the bed composition (specified using the d50 and d90 datasets) is optional. The input bed composition is only used to correct the sediment transport for hiding and exposure and does NOT change during simulation.

Multiple-sized Sediments

In the case of where multiple sediment size classes are specified, the bed material is discretized vertically into a number of layers and horizontally using the 2DH computational grid. The bed composition refers to the grain size distribution in each layer and cell.

The basic bed layer settings are specified in the *Bed Composition* section of the *Sediment* tab in the *CMS-Flow Model Control* window (see figure below). The basic parameters for the bed layers are the maximum number of bed layers, and the minimum and maximum bed layer thicknesses.

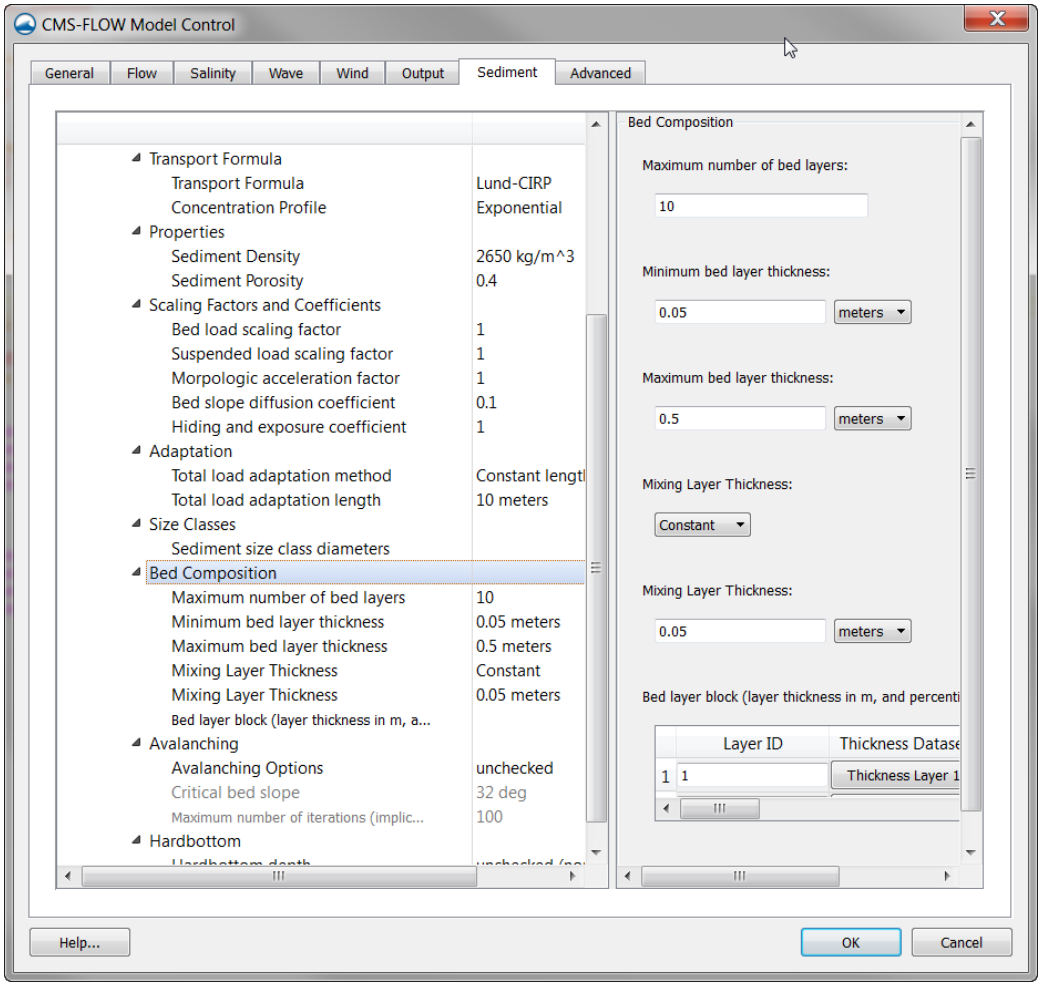


Figure 2-88. Bed Composition section within the *Sediment* tab of the CMS-Flow Model Control window.

The CMS-Flow cards used to specify the maximum number of bed layers, and the minimum and maximum bed layer thickness are described in the table below.

Table 2-87. CMS card used to specify the total-load adaptation coefficient.

Input	Format	Notes
Maximum Number of Bed Layers	[cards=BED_LAYERS_MAX_NUMBER] [name=**BedLayMaxNum**, type=*int*, default=10, range=(5<**BedLayMaxNum**<20)]	The input number of bed layers must be less or equal to the maximum number of bed layers
Minimum Bed Layer Thickness	[cards=BED_LAYERS_MIN_THICKNESS] [name=**BedLayMinThick**, type=*real*, default=0.05, required=false, typical=(0.001<**BedLayMinThick**<0.2)] [name=**BedLayMinThickUnits**, type=*char*, optional=true, options=**DistUnits**, default='m']	Specifies the minimum bed layer thickness for all layers during the simulation.
Maximum Bed Layer Thickness	[cards=BED_LAYERS_MAX_THICKNESS] [name=**BedLayMaxThick**, type=*real*, default=0.5, required=false, typical=(0.1<**BedLayMaxThick**<1.0)] [name=**BedLayMaxThicknits**, type=*char*, optional=true, options=**DistUnits**, default='m']	Specifies the minimum bed layer thickness for all layers during the simulation.

Mixing Layer Thickness

The mixing layer is the first layer from the surface which is allowed to exchange (or mix) sediments with the sediment transport. The mixing layer thickness is calculated based on the median grain size and bed form size. The option is also available to set the mixing layer to a constant. The CMS-Flow cards used for setting the mixing layer thickness to a constant is described in the table below.

Table 2-88. CMS-Flow used for setting a constant mixing layer thickness.

Input	Format	Notes
Mixing Layer Formulation	[cards=MIXING_LAYER_FORMULATION] [name=**MixLayForm**, type=*char*, default=CONSTANT options=(CONSTANT,AUTOMATIC)]	The input number of bed layers must be less or equal to the maximum number of bed layers
Mixing Layer Constant Thickness	[cards=MIXING_LAYER_THICKNESS_CONSTANT] [name=**MixLayConst**, type=*real*, default=0.05, required=false, typical=(0.001<**MixLayConst**<0.2)] [name=**MixLayConstUnits**, type=*char*, optional=true, options=**DistUnits**, default='m']	Sets the mixing layer thickness to a constant in meters.

Notes:

- Using a larger mixing layer will slow down the temporal change in bed material composition and enhance model stability, but will reduce the accuracy of the bed composition estimates.
- Erosional cases tend to have smaller mixing layer thicknesses and depositional cases tend to have larger mixing layer thicknesses.
- The mixing layer thickness should NOT be used as a calibration parameter.

Bed Layer Thickness and Composition

The initial bed layer thickness and composition are specified at least one layer for the whole grid. The Layer ID corresponds to the layer number from the surface downwards. If the Layer ID is not specified in the card file, then it is assumed to be equal to the bed layer block sequential number. In order to avoid having to specify all of the bed layers, the bed layer ID's are allowed to have gaps in which case the missing bed layers or copied from the first specified bed layer above. For example if bed layers 1 and 3 are specified and there is a maximum of 10 layers, then bed layer 2 is copied from bed layer 1, and bed layers 4 through 10 are copied from bed layer 3.

The bed composition is specified using percentile diameters which indicate the percentage of diameters smaller than a specific diameter. The percentile diameters are denoted by d_{per} where the subscript per indicates the percentage of diameters smaller than d.

Notes:

- There are several advantages to using percentile diameter datasets instead of the fraction of each sediment size class. The percentile diameters are independent of the sediment size classes and therefore do not need to be changed if the size classes are changed. In addition the percentile diameters are easier to manipulate spatially than sediment fractions because the sum of the later must equal to one.

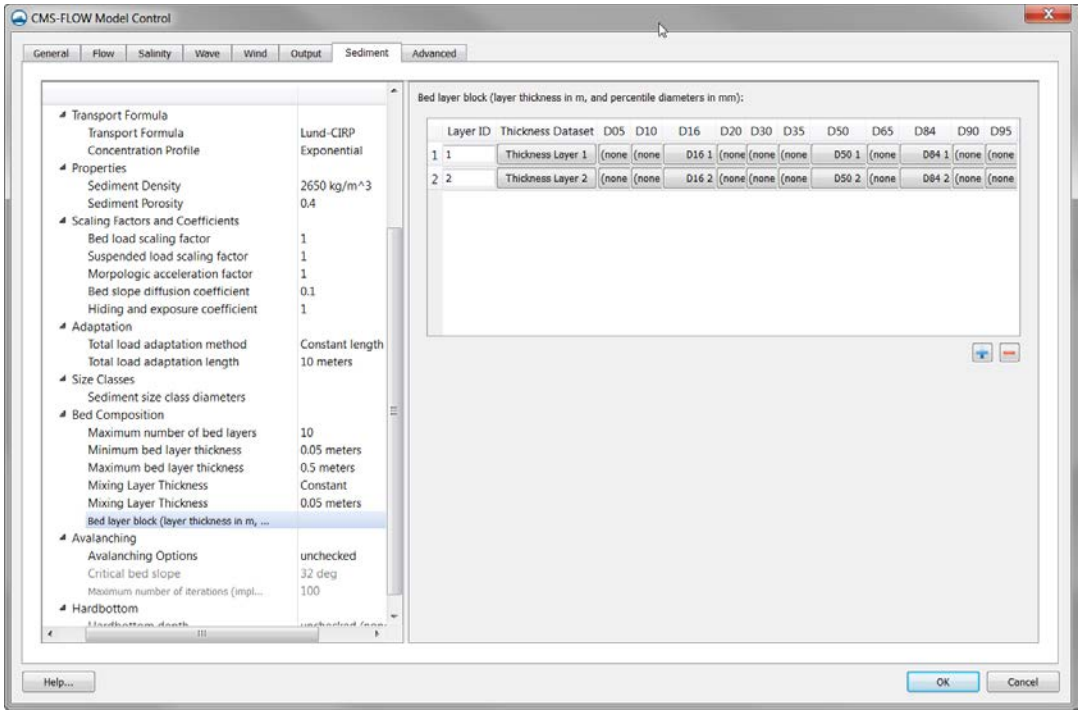


Figure 2-89. Bed layer block section within the *Sediment* tab of the *CMS-Flow Model Control* window in SMS 11.1.

A bed layer thickness and composition is defined with a bed layer block. The cards used to define the beginning and end of a bed layer block are described in the table below.

Table 2-89. CMS-Flow cards used to specify a *Bed Layer* block.

Input	Format	Notes
Begins a sediment size class block	`[begin=BED_LAYER_BEGIN, block=BedLay, children=BedLayNum]`	Begins a bed layer block.

Bed layer number	`[card=(LAYER,LAYER_NUMBER)]`
	`[name=BedLayNum, type=float, default=none, range=(diam >0.06 'mm')]`
	`[name=SedDens, type=char, options=DensUnits), default='mm']`
		Specifies the bed layer number from the surface. If not specified then set the sequential bed layer block number.

Ends a sediment size class block	`[end=BED_LAYER_END, block=BedLay]`	Ends a bed layer block

There are two methods for specifying each bed layer thickness:

1. Constant value
2. Spatially variable (dataset)

The CMS-Flow cards related to the bed layer thickness are described in the table below.

Table 2-90. CMS card used to specify the bed layer thickness.

Input	Format	Notes
Bed layer thickness dataset	[cards=THICKNESS_DATASET, SMS_ver=(>11.1), CMS_ver=(>4.10.00)] [name=**BedLayThickDatFile**, type=*char*, example= "Flow_grid.h5"] [name=**BedLayThickDatPath**, type=*char*, example= "Flow/Datasets/D50 2"]	Specifies the file name and dataset path (within the file) for the bed layer thickness dataset.
Bed layer thickness value	[cards=THICKNESS_VALUE] [name=**BedLayThickVal**, type=*real*] [name=**BedLayThickValUnits**, type=*char*, optional=true, options=**DistUnits**, default='m']	Specifies the bed layer thickness as a constant value.

There are four methods for specifying the bed layer composition:

1. D16, D50, and D84 Datasets
2. D35, D50, D90 Datasets
3. Cumulative Grain Size Distribution

The first two options assume a log-normal grain size distribution

$$p_k = \frac{\Delta d_k}{d_k \ln \sigma_g \sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{\ln d_k - \ln d_g}{\ln \sigma_g} \right)^2 \right] \tag{2-33}$$

where p_k is the fraction, $\Delta d_k = d_{uk} - d_{lk}$ is the size class bin width, d_k is the size class diameter, and d_{50} is the median grain size. The geometric standard deviation σ_g can be defined using the method of moments

$$\sigma_g = \exp\sqrt{\sum p_k \left(\ln d_k - \ln d_g \right)^2} \tag{2-34}$$

It is noted that the geometric standard deviation is dimensionless.

The geometric mean d_g is in given by

$$d_g = \exp\left(\sum p_k \ln d_k\right) \tag{2-35}$$

An alternate approach to the above equation for obtaining geometric standard deviation is by using the graphical measures (cumulative percentile values) (Folk and Ward 1957)

$$\sigma_g \approx \sigma_G = \exp\left(\frac{\ln d_{16} - \ln d_{84}}{4} + \frac{\ln d_5 - \ln d_{95}}{6.6}\right) \tag{2-36}$$

where σ_G is a graphical measure of the geometric standard deviation.

Sediments with a geometric standard deviation σ_g close to one are classified as well sorted and poorly graded (see table below).

Table 2-91. Sediment sorting classification based on the geometric standard deviation (Folk and Ward, 1957).

Geometric Standard Deviation	Sorting Classification
<1.27	Very well sorted
1.27-1.41	Well sorted
1.41-1.62	Moderately well sorted
1.62-2.00	Moderately sorted
2.00-4.00	Poorly sorted
4.00-16.00	Very poorly sorted
>16.00	Extremely poorly sorted

Recommendations:

- Areas with sediments with σ_g greater then approximate 1.41 should be simulated with multiple grain sizes.

D16, D50, and D84 Datasets

In this option, 2DH spatial datasets for the diameters corresponding to the 16th, 50th, and 84th percentiles are specified for the whole computational domain. The bed composition is calculated using the log-normal distribution (Equation 2.28) with $d_g \approx d_{50}$ and

$$\sigma_g \approx \left(\frac{d_{84}}{d_{16}} \right)^{1/2} \tag{2-37}$$

where the subscript indicates the percentile of the grain size diameter.

D30, D50, and D90 Datasets

In this option, 2DH spatial datasets for the diameters corresponding to the 30th, 50th, and 90th percentiles are specified for the whole computational domain. The bed composition is calculated using the log-normal distribution (Equation 2.28)with $d_g \approx d_{50}$ and

$$\sigma_g \approx \left(\frac{d_{90}}{d_{35}} \right)^{0.61} \tag{2-38}$$

where the subscript indicates the percentile of the grain size diameter.

Cumulative Grain Size Distribution

This option should be used in cases where the initial bed grain size distribution does not follow a log-normal distribution such as the case of a bimodal distribution. In this option more than 3 percentile diameter datasets need to be specified with at least one below and above the 5oth percentile. Currently the percentile diameters which are allowed in SMS 11.1 are: 5, 10, 16, 20, 30, 50, 64, 84, 90, and 95. The cumulative grain size distribution is linearly interpolated from the percentile diameters at the size class bounds and then converted to sediment fractions.

The CMS-Flow cards used to specify the bed composition are described in the table below

Table 2-92. CMS card used to specify the bed layer percentile diameters.

Input	Format
5th percentile diameter dataset	[cards=D05_DATASET, parent=**BedLay**] [name=**d05file**, type=*char*, example="Flow_grid.h5"] [name=**d05path**, type=*char*, example="Flow/Datasets/D05 1"]
10th percentile diameter dataset	[cards=D10_DATASET, parent=**BedLay**] [name=**d10file**, type=*char*, example="Flow_grid.h5"] [name=**d10path**, type=*char*, example="Flow/Datasets/D10 2"]
16th percentile diameter dataset	[cards=D16_DATASET, parent=**BedLay**] [name=**d16file**, type=*char*, example="Flow_grid.h5"] [name=**d16path**, type=*char*, example="Flow/Datasets/D16 3"]
20th percentile diameter dataset	[cards=D20_DATASET, parent=**BedLay**] [name=**d20file**, type=*char*, example="Flow_grid.h5"] [name=**d20path**, type=*char*, example="Flow/Datasets/D20 1"]
30th percentile diameter dataset	[cards=D30_DATASET, parent=**BedLay**] [name=**d30file**, type=*char*, example="Flow_grid.h5"] [name=**d30path**, type=*char*, example="Flow/Datasets/D30 1"]
50th percentile diameter dataset	[cards=D50_DATASET, parent=**BedLay**] [name=**d50file**, type=*char*, example="Flow_grid.h5"] [name=**d50path**, type=*char*, example="Flow/Datasets/D50 2"]
65th percentile diameter dataset	[cards=D64_DATASET, parent=**BedLay**] [name=**d64file**, type=*char*, example="Flow_grid.h5"] [name=**d64path**, type=*char*, example="Flow/Datasets/D65 2"]
84th percentile diameter datset	[cards=D84_DATASET, parent=**BedLay**] [name=**d30file**, type=*char*, example="Flow_grid.h5"] [name=**d84path**, type=*char*, example="Flow/Datasets/D84 3"]
90th percentile diameter dataset	[cards=D90_DATASET, parent=**BedLay**] [name=**d90file**, type=*char*, example="Flow_grid.h5"] [name=**d90path**, type=*char*, example="Flow/Datasets/D90 2"]
95th percentile diameter dataset	[cards=D95_DATASET, parent=**BedLay**] [name=**d95file**, type=*char*, example="Flow_grid.h5"] [name=**d95path**, type=*char*, example="Flow/Datasets/D95 1"]

Notes:

- When creating a percentile diameter dataset in SMS 11.1, the block number is appended to the end of the dataset name as for example "D50 2". The number is used to distinguish datasets from different bed layer blocks and is not necessarily the same as the bed layer ID number.
- If the percentile diameters are specified outside of a bed layer block, then they are assumed to belong to the first and only layer.

Advanced Bed Layer Composition

Two additional options are available for specifying the bed composition:

1. D50 dataset and constant geometric standard deviation
2. Size class fractions

A description of the CMS-flow card used to select each bed material composition option is described in the table below.

Table 2-93. CMS-Flow card used to specify the option for specifying the bed material composition.

Input	Format
Bed Layer Geometric Standard Deviation σ_g	[cards=(SIGMA,GEO_STD_DEV), parent=**BedLay**, SMS_ver=none, CMS_ver=(>4.10.00)] [name=**BedLaySig**, type=*real*, range=(**BedLaySig**>=1.0), default=1.5]
Bed Layer Size Class Fractions p_k	[cards=FRACTIONS, parent=**BedLay**, SMS_ver=none, CMS_ver=(>4.10.00)] [name=**SedClassNum**, type=*int*] for(k=1:**SedClassNum**, [name=**BedLayFrac**(k), type=*real*, range=(0.0>BedLayFrac(k)<1.0)])

Notes:

- The above two options are not available in SMS and are therefore considered advanced.
- The geometric standard deviation must always be equal or larger than 1.0.
- When the user specifies the sediment fractions, the sum of the fractions must equal 1.0. If they do not equal 1.0, the fractions are divided by the sum so that they equal 1.0.

Examples

In the case of single-sized sediment transport, the option is available to specify a median grain size dataset which is used to calculate a hiding and exposure correction to the single transport grain size.

Example 2-88. Bed composition specification for a single-sized sediment transport simulation.

```
!Note: Only one layer can to be specified and is optional
BED_LAYER_BEGIN !Block optional since there is only one layer
   D50_DATASET       "Flow_grid.h5" "SimpleCMS-FLOW/Datasets/D50 1"
BED_LAYER_END
```

Example 2-89. Bed layer thickness and composition specification for a multiple-sized sediment transport simulation.

```
BED_LAYERS_MAX_NUMBER            10       !> # of bed layer blocks
BED_LAYERS_MIN_THICKNESS         0.05 'm' !
BED_LAYERS_MAX_THICKNESS         0.5 'm' !
MIXING_LAYER_FORMULATION         CONSTANT !CONSTANT | AUTOMATIC
MIXING_LAYER_THICKNESS_CONSTANT  0.05 'm' !for all cells
BED_LAYER_BEGIN
   LAYER                          1 !bed layer # from surface downwards
   THICKNESS_DATASET           "Flow_grid.h5" "FLOW/Datasets/Thickness Layer 1"
   D16_DATASET                   "Flow_grid.h5" "FLOW/Datasets/D16 1"
   D30_DATASET                   "Flow_grid.h5" "FLOW/Datasets/D30 1"
   D50_DATASET                   "Flow_grid.h5" "FLOW/Datasets/D50 1"
   D84_DATASET                   "Flow_grid.h5" "FLOW/Datasets/D84 1"
   D90_DATASET                   "Flow_grid.h5" "FLOW/Datasets/D90 1"
BED_LAYER_END
!Note: bed layer 2 not specified, so it is copied from the first
BED_LAYER_BEGIN
   LAYER                          3 !bed layer # from surface downwards
   THICKNESS_VALUE                0.10 'm' !for
   D16_DATASET                   "Flow_grid.h5" "FLOW/Datasets/D16 2"
   D50_DATASET                   "Flow_grid.h5" "FLOW/Datasets/D50 2"
   D84_DATASET                   "Flow_grid.h5" "FLOW/Datasets/D84 2"
BED_LAYER_END
BED_LAYER_BEGIN
   LAYER                          4 !bed layer # from surface downwards
   THICKNESS_VALUE                0.20 'm' !for
   D30_DATASET                   "Flow_grid.h5" "FLOW/Datasets/D30 3"
   D50_DATASET                   "Flow_grid.h5" "FLOW/Datasets/D50 3"
   D90_DATASET                   "Flow_grid.h5" "FLOW/Datasets/D90 3"
BED_LAYER_END
```

Hiding and Exposure

The hiding and exposure is an important parameter for sediment transport. Currently, there are no well established coastal sediment transport formulas for nonuniform sediments. The most common approach is to adapt single-size transport formulas by applying correction factors. The most common of these factors is the hiding and exposure correction. The implementation of this factor varies depending on the transport formula and usually involves a coefficient (exponent) which needs to be calibrated. Because all of the transport formulas used in CMS were originally developed for uniform sediments, there is still a large uncertainty in the value of the hiding and exposure coefficient. It is expected that the hiding and exposure coefficient will be different for different transport formula especially since the implementation of the hiding and exposure correction may be different for different formula.

Single-sized Sediment Transport

For single-sized sediment transport, the correction function for hiding and exposure is calculated following Parker et al. (1982) as

$$\xi_k = \left(\frac{d_{50}}{d_k} \right)^m \tag{2-39}$$

where m is the hiding and exposure coefficient. The approach has been successfully applied to Shinnecock Inlet, NY to simulate morphology change at a coastal inlet (Sánchez and Wu 2011a). If the sediment is not well sorted, then a multiple sediment fractions need to be simulated (see section below)

Multiple-sized Sediment Transport

The hiding and exposure of the each sediment size class is considered by modifying the critical shields parameter Θ_{ck} for each sediment size class based on Wu et al. (2000)

$$\xi_k = \left(\frac{P_{ek}}{P_{hk}} \right)^{-m} \tag{2-40}$$

where m is an empirical coefficient that varies for each transport formula, approximately equal to 0.6-1.0. P_{ek} and P_{hk} are the total hiding and exposure probabilities and are calculated as

$$P_{hk} = \sum_{j=1}^{N} p_{1j} \frac{d_j}{d_k + d_j} \qquad P_{ek} = \sum_{j=1}^{N} p_{1j} \frac{d_k}{d_k + d_j} \tag{2-41}$$

where N is the number of grain size classes.

The hiding and exposure coefficient used for both the single- and multiple-sized sediment transport is specified with the same CMS-Flow card (see table below).

Table 2-94. CMS-Flow card used for setting the hiding and exposure coefficient.

Input	Format	Notes
Hiding and Exposure Coefficient *m*	[cards=HIDING_EXPOSURE_COEFFICIENT] [name=**SedHidExpCoeff**, type=*float*, default=1.0, optional=true, typical=(0.1><**SedHidExpCoeff**<1.0)]	Sets the hiding and exposure coefficient.

Recommendations:

- Due to the uncertainty in the hiding and exposure correction coefficient. It is recommended to always calibrate the hiding and exposure coefficient. If no or limited data is available for calibration then a sensitivity analysis should be done

Note:

- The hiding an exposure coefficient may have a different “best” value for different transport formula.

Avalanching

Avalanching is the process of sediment sliding when the critical angle of repose is reached. In CMS, avalanching is simulated using a mass conservative relaxation method which limits the bed slope to the critical angle of repose. For most coastal applications, the critical angle of repose is never reached, so it is not needed. The CMS-Flow cards used for specifying avalanching, and its options, are described in table below.

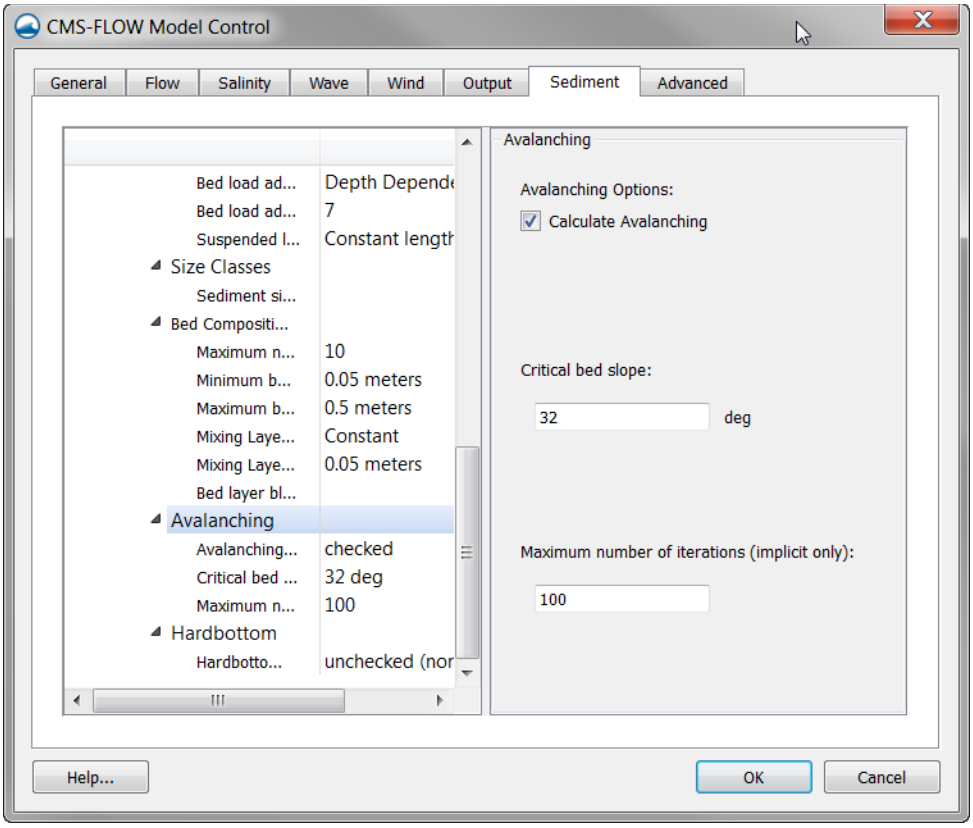


Figure 2-90. *Avalanching* section within the *Sediment* tab of the *CMS-Flow Boundary Conditions* window.

The table below describes the CMS-Flow cards used to specify the avalanching options.

Table 2-95. CMS-Flow cards related to avalanching.

Input	Format	Notes
Avalanching Activation	[card=USE_AVALANCHING] [name=**Aval**, type=*char*, options=(ON,OFF), default=OFF]	Turns On or Off the avalanching.
Repose Angle	[card=REPOSE_ANGLE, dependence=(**Aval**==ON)] [name=**RepAng**, type=*real*, default=32.0 'deg'] [name=**RepAngUnits**, type=*char*, optional=true, options=**AngUnits**, default='deg']	Specifies the sediment repose angle. Avalanching is activated when the bed slope exceeds the repose angle.
Avalanching Maximum Iterations	[card=AVALANCHE_MAX_ITERATIONS, dependence=(**Aval**==ON), dependence=(**TimeSolSch**=IMPLICIT)] [name=**AvalIterMax**, type=*int*, default=200]	Maximum number of iterations for implicit solution scheme. When using the explicit scheme, one iteration is performed every time step.
Avalanching Relaxation Coefficient	[card=(AVALANCHE_RELAX,AVALANCHE_RELAX_COEF), dependence=(**Aval**==ON)] [name=**AvalRelax**, type=*real*, default=0.1]	Relaxation coefficient for the avalanching algorithm.

Example 2-90. Specifying the avalanching settings.

```
USE_AVALANCHING             ON          !{OFF} | ON
REPOSE_ANGLE                32.0 'deg'  !Units optional
AVALANCHE_MAX_ITERATIONS    200         !Note: only for implicit scheme
```

Notes:

- Turning on avalanching increases slightly the computational time.
- For most coastal morphodynamic applications, the avalanching can be turned off since the angle of repose is almost never reached. If a simulation is run without avalanching and steep angles are found in the calculated morphology change, then the simulation should be rerun with avalanching turned on.

Hard Bottom

Hard Bottom is a morphologic constraint that provides the capability to simulate mixed bottom types within a single simulation. This cell-specific feature limits the erodability of the constrained cells down to a specified depth below the water surface. During sediment transport calculations, exposed hard bottom cells may become covered through deposition. By default, CMS-Flow cells are fully-erodible cells with no specified hard bottom depth (inactive cells; denoted by the CMS-Flow null value of -999.0). Hard bottom only needs to be specified only for computational (ocean) cells.

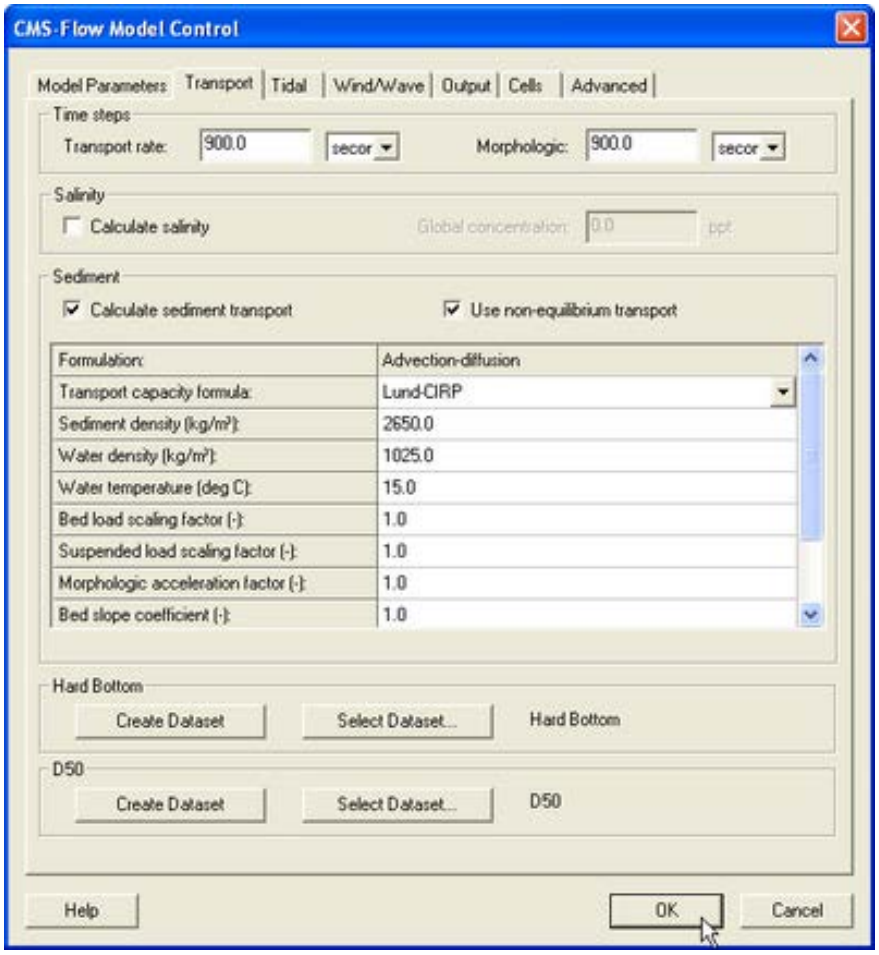


Figure 2-91. CMS-Flow Model Control window showing the location where the hard bottom dataset is specified.

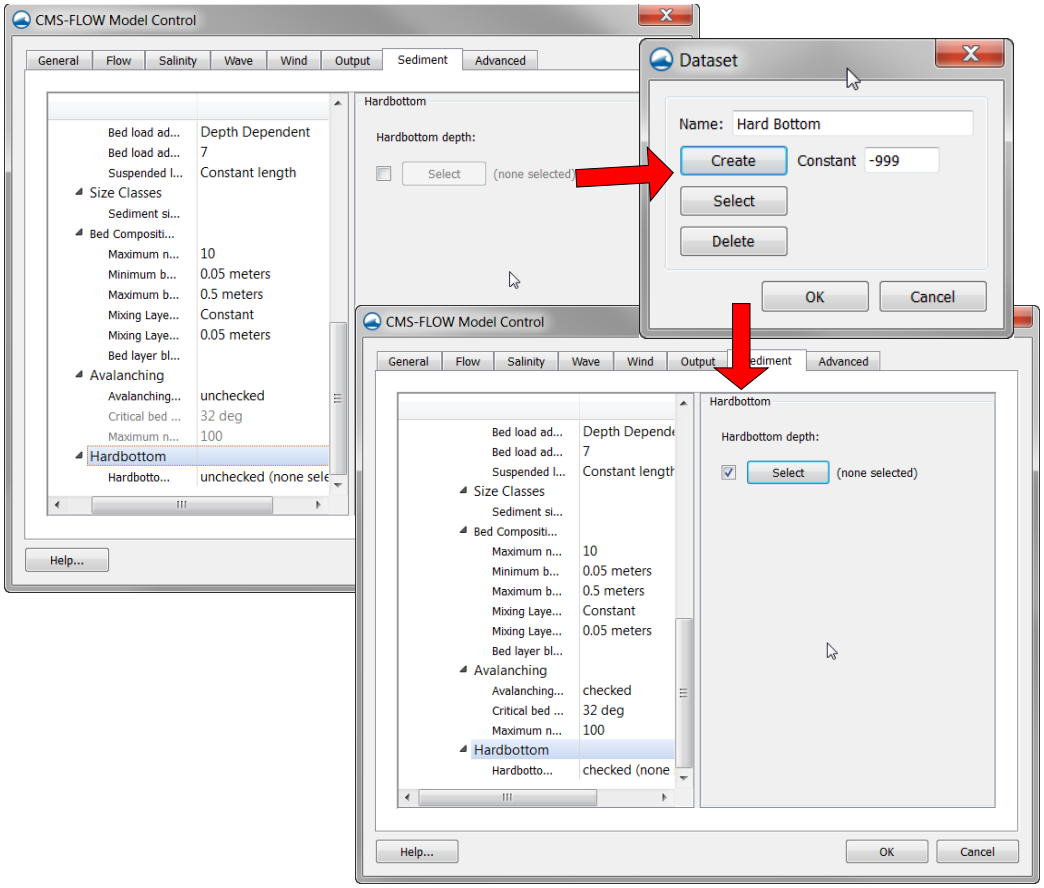


Figure 2-92. Specification of the Hardbottom Dataset in the *Sediment* tab of the *CMS-Flow Model Control* window in SMS 11.1.

Within the *CMS-Flow Model Control* window, the hard bottom dataset can be created from the *Sediment* tab. If the dataset does not exist, it can be created using the *Create Dataset* button. If a dataset exists (created using the Data Calculator) which represents the intended hard bottom specifications, the *Select Dataset* button can be used to select such dataset and copy the values to the hard bottom dataset.

When specified, cell hard bottom depths will appear in the Project Explorer as a scalar dataset beneath the CMS-Flow grid. This dataset cannot be deleted, though it can be edited like any other dataset. A CMS-Flow simulation must contain the hard bottom dataset (even if it is not specified) so SMS will create a defaulted (inactive cells) dataset if it does not already exist when saving the simulation. The hard bottom dataset can created, edited, viewed and verified using the following SMS interface features.

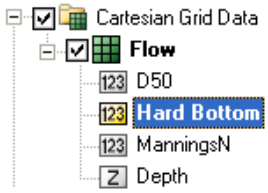


Figure 2-93. SMS Project Explorer showing Hard bottom dataset

Example 2-91. Specifying the hard bottom dataset.

```
HARDBOTTOM_DATASET        "Flow_grid.h5" "FLOW/Datasets/Hard Bottom"
```

Boundary and Initial Conditions

In the case of the Equilibrium Total Load sediment transport model, all boundaries are set to the equilibrium transport rate. For the Equilibrium Bed Load plus Advection Diffusion model, the suspended load is specified as the equilibrium concentration at inflow cells and a zero gradient at outflow cells. For the Total load nonequilibrium sediment transport model, the sediment concentration is set to the equilibrium concentration at inflow cells and a zero gradient boundary condition is applied at outflow cells.

In the case an initial conditions file is NOT specified both the hydrodynamics and sediment concentrations are initialized as zero. If an initial conditions file is specified, than the initial sediment concentrations are read in. If an initial conditions file is specified but without the sediment concentration, than the initial sediment concentration is set to the equilibrium concentration.

Table 2-96. CMS Flow cards related to the boundary conditions.

Card	Arguments	Default	Range	Description
NET_LOADING_FACTOR	real	1.0	0.5-2.0	Used to specify under- or over-loading at sediment inflow boundaries. Only for NET.
SEDIMENT_INFLOW_LOADING_FACTOR	Real	1.0	0.5-2.0	Used to specify under- or over-loading at sediment inflow boundaries.
CALC_MORPH_DURING_RAMP	character	ON	ON OFF	Determines whether to calculate the morphology change during the ramp period

Step-by-Step Sediment Transport Application for Shark River Inlet, NJ

TBC

Numerical Methods

> *The most overlooked advantage of owning a computer is that if they foul up there is no law against whacking them around a little.*
> *- Joe Martin*

Temporal Solution Scheme

This refers to the temporal discretization of the hydrodynamic, sediment and salinity transport equations. There are two options in CMS: Implicit and Explicit. The implicit scheme uses a time step on the order of 5-20 minutes and is designed for tidal flow, and mid-term morphology change. The explicit scheme uses a time step on the order of 0.5-1.0 seconds and is appropriate for cases that vary quickly in time such as flooding or barrier island breaching.

Table 2-97. CMS-Flow cards related to the temporal solution scheme.

Input	Format	Notes
Hydrodynamic time step	[cards=HYDRO_TIMESTEP, optional=*false*] [name=**HydroDt**, type=*float*, default=**none**] [name=**HydroDtUnits**, type=*char*, options=**timeUnits**, default='s']	Sets the time step for the hydrodynamics.
Temporal solution scheme	[cards=SOLUTION_SCHEME, versions=(>4.0)] [name=**TimeSolSch**, type=float, options=(EXPLICIT,IMPLICIT), default=IMPLICIT]	Determines the temporal solution scheme used in CMS-Flow.
Implicit weighting factor	[cards=IMPLICIT_WEIGHTING_FACTOR, CMS_version=(4.10.00)] [name=**ImpWghtFac**, type=*float*, options=(0.0>=**ImpWghtFac**<=1.0), default=0.0]	Weighting factor in implicit temporal scheme. 0 – First order, 1-second order. For more details on the temporal solution scheme see the Temporal Discretization section.

Note:

- The second order implicit temporal scheme requires three time step levels. Therefore, for the first time step, the model uses the first order two-level temporal scheme. In addition, if the time step is increased or decreased during the simulation. The first order scheme is used.

Implicit Solver Options

The solvers implemented in the implicit temporal solution scheme are the SIP, ICCG, Gauss-Seidel, Gauss-Seidel with Successive-Over-Relaxation, BICGSTAB, and GMRES. Currently, the same solver is applied to flow, sediment and salinity. The default solver is the GMRES. The solver may be changed using the advanced card in the table below. The SIP and ICCG solvers are only available for non-telescoping grids. The maximum number of outer and inner loop iterations may also be changed. The outer loop is the loop over which the governing equations are solved successively, while the inner loop is the loop within the matrix solver for each governing equation.

Table 2-98. CMS-Flow cards related to the implicit solver options.

Input	Format	Notes
Maximum number of iterations for the hydrodynamic outer loop	[cards=HYDRO_MAX_ITERATIONS, dependance=(**TimeSolSch**==IMPLICIT)] [name=**HydroMaxIter**, type=*int*]	Sets the maximum number of iterations for the flow (hydro) solver (outer loop).
Matrix Solver	[cards=MATRIX_SOLVER, dependance=(**TimeSolSch**==IMPLICIT)] [name=**Solver**, type=*float*, options=(SIP,ICCG,BICGSTAB, GAUSS-SEIDEL,GAUSS-SEIDEL-SOR, BICGSTAB,GMRES), default=GMRES]	Selects the matrix solver for flow, sediment and salinity.
Maximum number of iterations for the pressure equation	[cards=PRESSURE_ITERATIONS, dependance=(**TimeSolSch**==IMPLICIT)] [name=**PresIter**, type=*float*, typical=(5><**PresIter**<30), default=none]	Sets the number of solver iterations for the pressure equation (inner loop).
Maximum number of iterations for the velocity equation	[cards=VELOCITY_ITERATIONS, dependance=(**TimeSolSch**==IMPLICIT)] [name=**VelIter**, type=*float*, typical=(5><**VelIter**<30), default=none]	Sets the number of solver iterations for the velocity equation (inner loop).
Maximum number of iterations for the sediment transport equation equation	[cards=SEDIMENT_MAX_ITERATIONS, dependance=(**TimeSolSch**==IMPLICIT)] [name=**SedMaxIter**, type=*float*, typical=(5><**SedMaxIter**<30), default=20]	Sets the number of solver iterations for the sediment transport (outer loop)
Maximum number of iterations for the pressure equation	[cards=SALINITY_MAX_ITERATIONS, dependance=(**TimeSolSch**==IMPLICIT)] [name=**PresIter**, type=*float*, typical=(5><**PresIter**<30), default=none]	Sets the number of solver iterations for the pressure equation (inner loop).

Skewness Correction

When the lines connecting cell centers do not intercept the cell-face centers, the cells are said to be skewed. When interpolating variables to the cell-face or calculating cell-face gradients a correction is needed for second order accuracy. There are several ways in which the correction can but involves some form of reconstruction of the variable on the cell face or within the neighboring cells. In CMS a linear cell reconstruction is performed within skewed cells for the correction.

Table 2-99. CMS-Flow cards related to the skewness correction.

Input	Format	Notes
Second order Skewness Correction	[cards=SKEWNESS_CORRECTION, dependance=(**TimeSolSch**==IMPLICIT)] [name=**SkewCorr**, type=*bool*, options=(ON,OFF), default=ON]	Turns on or off the second order skewness correction which account for non-orthogonality in the telescoping grids.

Advection Schemes

As in the case of the implicit solution scheme, the same advection scheme is applied for the flow, sediment and salinity transport equations. Future versions of the CMS will allow the user to select different advection schemes for different governing equations. There are several choices for advection schemes with the implicit model which are listed in the table below. The schemes range from first to third order. The hybrid scheme is fast but is the most diffusive. The exponential scheme is based on the 1D analytical solution to a steady-state advection-diffusion equation and produces very stable results. The HLPA is very stable and non-diffusive, but requires slightly more computational time. For most applications, the exponential scheme is recommended and is set as the default. The advection scheme may be change using the advanced card described in the table below.

Table 2-100. CMS-Flow cards related to the implicit solver options.

Input	Format	Notes
Advection scheme	[cards=ADVECTION_SCHEME, dependance=(**TimeSolSch**==IMPLICIT)] [name=**AdvSch**, type=*char*, options=(NONE,UPWIND,HYBRID, POWERLAW,EXPONENTIAL,HLPA, GAMMA,CUBISTA,ALVSMART,HOAB), default=EXPONENTIAL]	Specifies the advection scheme for the implicit solver. The advection scheme is applied to the momentum, and sediment and salinity transport equations.

Wetting and Drying

In CMS, a minimum depth is required for cells to be considered. A cell is classified as wet if the total water depth is larger than this depth. Cell faces are either classified as either open if the two cells neighboring cells are wet or otherwise closed (i.e. cell faces are not classified as wet or dry) , in order to improve stability.

Table 2-101. CMS-Flow cards related to wetting and drying.

Input	Format	Notes
Wetting and drying depth	[cards=DRYING_DEPTH] [name=**hmin**, type=*real*, default=0.05 'm', range=(**hmin**>0.0 'm')] [name=**hminUnits**, type=*char*, options=**DistUnits**), default='m']	Sets the minimum depth for wet cells.
Water ponding	[cards=WATER_PONDING, dependance=(**TimeSolSch**==IMPLICIT)] [name=**Solver**, type=*char*, options=(ON,OFF), default=OFF]	Turns *On* or *Off* water ponding. If water ponding is *Off*, isolated bodies of water will become dry.
Allow flow through one-cell wide channels	[cards=(ONE_CELL_WIDE_CHANNELS, ONE-CELL-WIDE-CHANNELS, NARROW_CHANNELS), dependance=(**TimeSolSch**==IMPLICIT)] [name=**PresIter**, type= *char*, typical=(5><**PresIter**<30), default=none]	Limits wetting and drying to areas with at least 3 cells wide. When turned off, the model stability is improved.

Parallelization

The CMS-Flow is parallelized for PC's with multi-core processors using OpenMP. The parallelization works by splitting the computational work into "threads" among several cores. Some cores are hyper-threaded, meaning a single core may support two threads. The number of threads is specified in the *CMS-Flow Model Control Wind*ow. The number of threads must be equal or greater to 1 and cannot be larger than the number of threads available on the machine. If a number is specified which is larger than the maximum number available on the machine, then the code will default to the maximum number available.

Table 2-102. CMS-Flow card used to specify the number of threads.

Input	Format	Notes
Number of computing threads	[cards=NUM_THREADS] [name=**NumThr**, type=*int*, range=(**NumThr**>=1), default=1]	Determines the number of threads used for parallel processing.

Note:

- The OpenMP parallelization requires that compatibility OpenMP runtime library (libiomp5md.dll) be in search paths.
- When running multiple simulations at once, the user should keep track of how many threads are being used by each simulation and make sure that the maximum number of threads on the computer is not exceeded. Exceeding the maximum number of threads on the computer will slow down the simulations and also make the computer slow.
- It is always recommended to at least leave one thread unused by model simulations so that the computer is responsive while the simulations are running.

Hot Start

The term “hot start” refers to starting a simulation with an initial condition other zero (cold start). Hot starts are used for specifying initial conditions other than the default zero value or restarting simulations at intermediate times. The hot start controls are set in the *CMS-Flow Model Control* window under the *Flow* tab in SMS 11.0 (see Figure 2-94) and in the *General* tab in SMS 11.1 and later (see Figure 2-95).

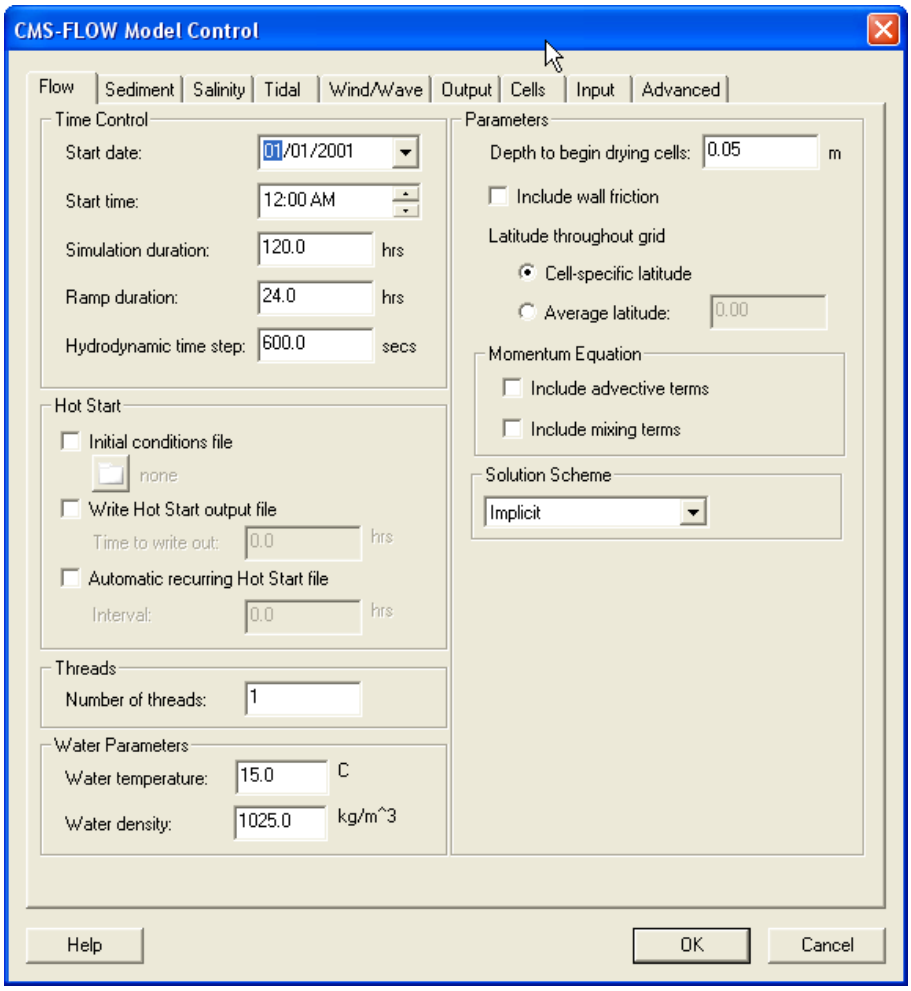


Figure 2-94. Location of Hot Start input and output options within the *Flow* tab of the *CMS-Flow Model Control* window in SMS 11.0.

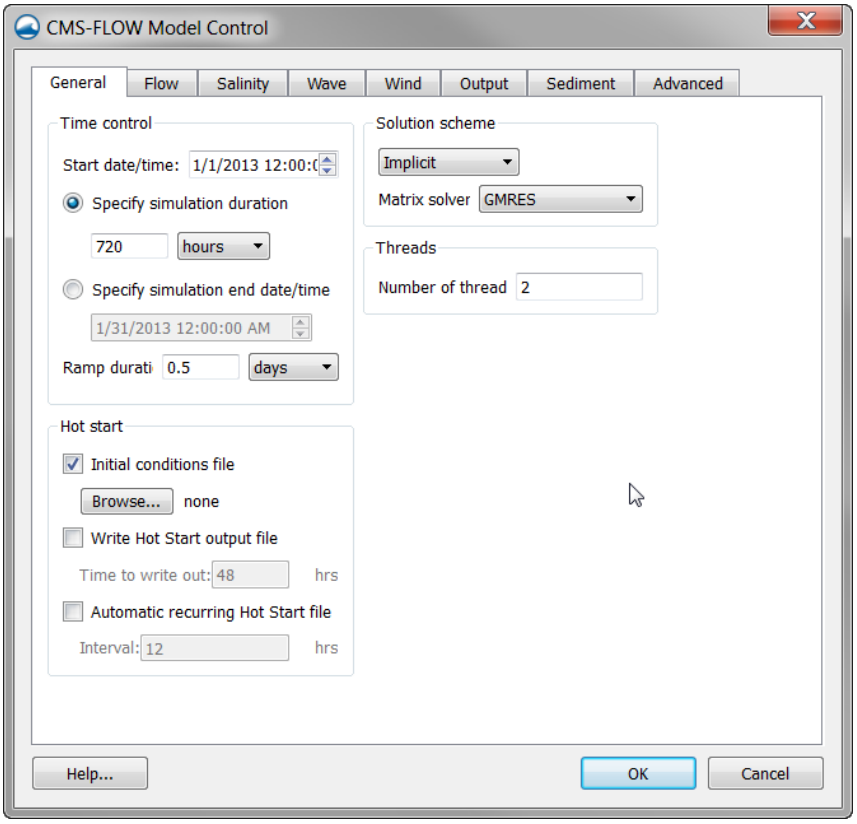


Figure 2-95. Location of Hot Start input and output options within the *Flow* tab of the *CMS-Flow Model Control* window in SMS 11.0.

The hot start options are the input *Initial Conditions* file, the output time and recurring interval for the Hot Start file. A description of these controls and options are provided in the subsequent sections.

Hot Start Output File

The CMS *Hot Start* feature lets the user restart simulations that have been ended or stopped due to for example electric outages, hardware malfunctions, or model crashes. In the case of a model crash the user, may restart the model using larger solver iterations and/or time steps to stabilize the simulation. The user has the option to specify a hot start output time or an interval for outputting a recurring hot start file. Every time the hot start file is written, it overwrites the previous information. The CMS *Hot Start* file saves information on the water elevation (pressure), and current velocities. If the sediment transport is active, the water depth and sediment concentrations are also saved for each size class. The CMS *Hot Start File* is a binary XMDF file, has the name *Hot_Start.h5* and is saved in the directory of the CMS-Flow files. Figure 2-96 shows the structure of the hot start file. After saving a CMS *Hot Start File*, it is a good idea to rename the file with a different name before using it as an initial conditions file. This way, the file will not be overwritten in future simulations.

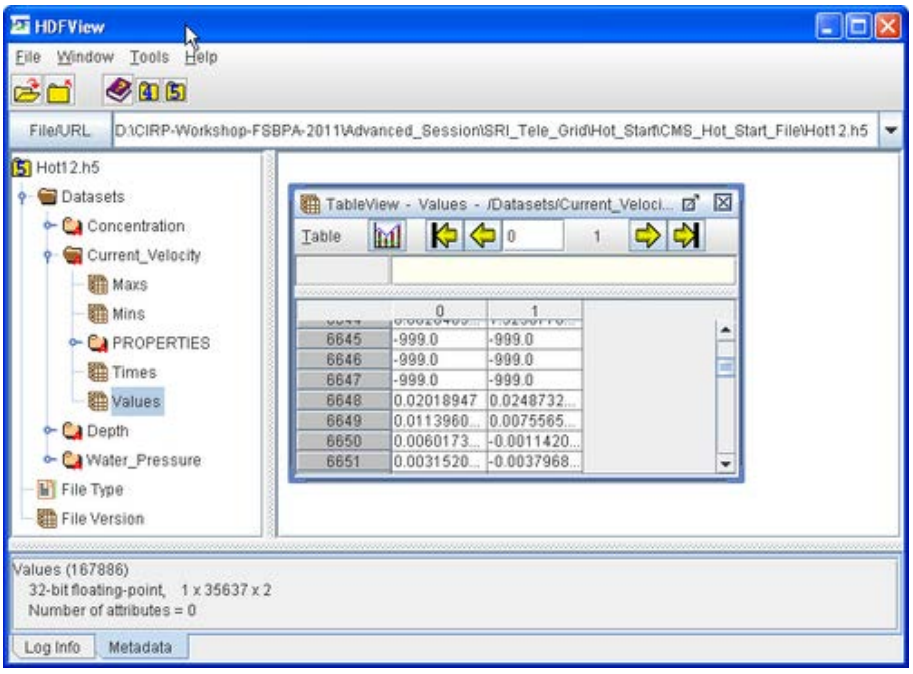


Figure 2-96. HDFView showing the structure of the CMS Hot Start File.

Table 2-103. CMS-Flow cards used to specify the hot start output file.

Input	Format	Notes
Hot start output file	[cards=HOT_START_OUTPUT_FILE, versions=(>3.75.00)] [name=**HotStartFile**, type=*char*, default="Hot_Start.h5", optional=true]	Specifies the name of the hot start output file
Hot start file single output time	[cards=HOT_START_TIME] [name=**HotStartTime**, type=*real*, default=48.0] [name= **HotStartTimeUnits**, type=*char*, optional=true, options=**TimeUnits**), default='hrs']	Specifies a single hot start time, typically at the end of a simulation
Hot start file recurring output interval	[cards=(AUTO_HOT_START_INTERVAL, HOT_START_INTERVAL)] [name=**HotStartInter**, type=*real*, default=12.0] [name= **HotStartInterUnits**, type=*char*, optional=true, options=**TimeUnits**), default='hrs']	Hot start file recurring output interval. The output times are calculated with respect to the start of the simulation

Example 2-92. Hot Start Output Options – Simple Case.

```
!Hot Start
AUTO_HOT_START_INTERVAL          24.0 hrs !default units is hrs
```

Example 2-93. Hot Start Output Options – Advanced Case

```
!Hot_Start
HOT_START_TIME          7.0 days !Units optional, default is hrs
HOT_START_INTERVAL      2.0 days !can also use AUTO_HOT_START_INTERVAL
HOT_START_OUTPUT_FILE   "myHotStartFile.h5" !default is "Hot_Start.h5"
```

Table 2-104. Path and name for hot start file variables.

Variable	Path and Name	Units	Notes
Water pressure	Datasets\Water_Pressure	m^2/s^2	Equal to the water level times gravity.
Current velocity	Datasets\Current_Velocity	m/s	Saved as a vector dataset
Total water depth	Datasets\Depth	m	
Single-size sediment concentrations	Datasets\Concentration	kg/m^3	For single-size sediment transport. Only for multiple-sized sediment transport.
Multiple-sized sediment concentrations	Datasets\Concentration_*k*	kg/m^3	Sediment concentration for the k^{th} size class. Only for multiple-sized sediment transport.
Bed layer thickness	Datasets\Thickness_(*l*)	m	Thickness for bed layer *l*. Only for multiple-sized sediment transport.
Bed composition	Datasets\Fraction_*k*(*l*)	-	Fractional bed composition for the k^{th} size class and bed layer *l*. Between 0 and 1. Sum of all fractions in each layer and cell must equal to 1.0. Only for multiple-sized sediment transport.
Salinity concentrations	Datasets\Salinity	ppt	

Initial Conditions Input File

There are several situations where it is convenient to specify a user defined initial condition (hot start) file. For example, if the user forgets to setup the model output a hot start file or when running idealized cases with known initial conditions. A hot start file can easily be created and exported by the user from the SMS interface. The model requires at water levels, current velocities, concentrations, and water depths. Any datasets that are missing from the initial file are assigned a default value which depends on the dataset. If the water level or current velocities are not specified, they are set to zero. If the depth is not specified, then it is set to the input grid depth. If the sediment concentrations are not specified, then they are set to the equilibrium concentrations. It is important to note that the names and paths of the initial condition datasets are important.

Table 2-105. CMS-Flow cards used to specify the hot start input file.

Input	Format	Notes
Input initial conditions file	[cards=INITIAL_CONDITION_FILE, versions=(>3.75.00)] [name=**InitCondFile**, type=*char*, default="Hot_Start.h5", optional=true]	Specifies the name of the input initial conditions file.
Initial condition time	[cards=INITIAL_CONDITION_TIME] [name= **InitCondTime**, type=*real*, default=none] [name= **InitCondTimeUnits**, type=*char*, optional=true, options=**TimeUnits**), default='hrs']	Specifies the initial condition starting time relative to the current simulating starting time
Initial condition date and time	[cards=INITIAL_CONDITION_DATE_TIME] [name= **InitCondDateTime**, type=*char*, default=none, example="2013/11/25 23:59:00"]	Specifies the calendar date and time

The initial conditions file contains grid datasets with initial values for water levels, current velocities, salinity and sediment concentrations, bed composition, bed layer thicknesses, and water depth. All of the input datasets are optional. If the dataset does not exist than it is set to the default value.

Table 2-106. Path and name for initial condition file variables.

Variable	Path and Name	Units	Notes
Water surface elevation	Datasets\Water elevation	m	Same as water level. Either the water level or water pressure need to specified and not both.
Water pressure	Datasets\Water_Pressure	m^2/s^2	Equal to the water level times gravity. Either the water level or water pressure need to specified and not both.
Current velocity	Datasets\Current_Velocity	m/s	Saved as a vector dataset
Total water depth	Datasets\Depth	m	
Single-size sediment concentrations	Datasets\Concentration	kg/m^3	For single-size sediment transport. Only for multiple-sized sediment transport.
Multiple-sized sediment concentrations	Datasets\Concentration_*k*	kg/m^3	Sediment concentration for the k^{th} size class. Only for multiple-sized sediment transport.
Bed layer thickness	Datasets\Thickness_(*l*)	m	Thickness for bed layer *l*. Only for multiple-sized sediment transport.
Bed composition	Datasets\Fraction_*k*(*l*)	-	Fractional bed composition for the k^{th} size class and bed layer *l*. Between 0 and 1. Sum of all fractions in each layer and cell must equal to 1.0. Only for multiple-sized sediment transport.
Salinity concentrations	Datasets\Salinity	ppt	

Example 2-94. Initial Condition Options –Hot Start File.

```
!Hot Start
INITIAL_CONDITION_FILE    "myHotStartFile.h5" !default is “Hot_Start.h5”
```

Example 2-95. Initial Condition Options – User-specified datasets.

```
!Hot_Start
INITIAL_CONDITION_FILE    "myInitCondFile.h5" !User-specified datasets
!Note: The time of the initial condition is read from the input file
```

Example 2-96. Initial Condition Options – User-specified datasets.

```
!Hot_Start
INITIAL_CONDITION_FILE    "Init_Cond.h5" !User-specified datasets
INITIAL_CONDITION_TIME     24.0 days !default units is hrs
!Note: The time of the initial condition in the XMDF file is ignored and
!set to the above value
```

Note:

- The time of the initial conditions file is with respect to a reference time of the current simulation. The default units is hours but may be changed to any units.
- The initial condition and hot start files only contain a single time step. Therefore, when selecting the second-order backward difference implicit scheme which uses two previous time steps, the first time step is calculated using the first-order backward difference scheme. In the case of the explicit temporal scheme, only the first-order forward difference scheme is available.

Creating an Initial Conditions File

The steps for creating a user defined hot start or initial condition file from a CMS-Flow solution file are outlined below.

1. Import CMS-Flow grid and solution file.
2. Sample a time step of the solution datasets for use in the initial condition
   2.1. Click on *Data | Data Calculator*
      2.1.1. Under the *Tools* section, select *Sample time steps*.
      2.1.2. Under the *Datasets* section, click on the *Water Elevation*
3. Export the initial condition datasets to an XMDF file

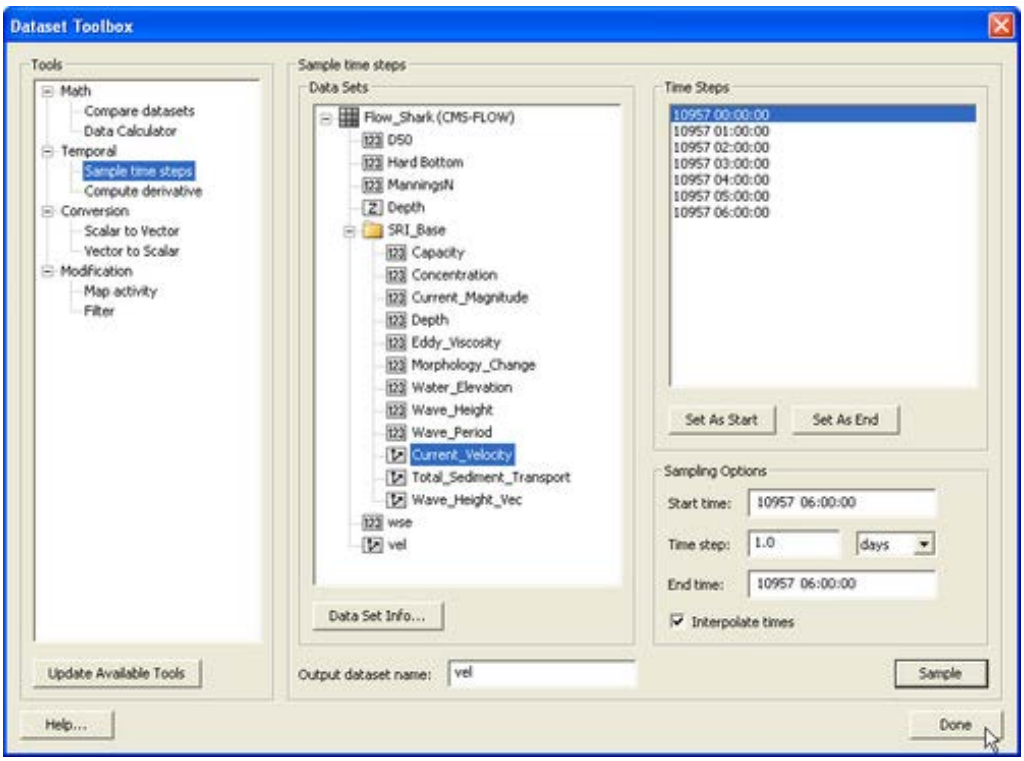


Figure 2-97. Dataset Toolbox showing a time step sample of the water elevation and current velocity datasets for use in a hot start (initial condition) file.

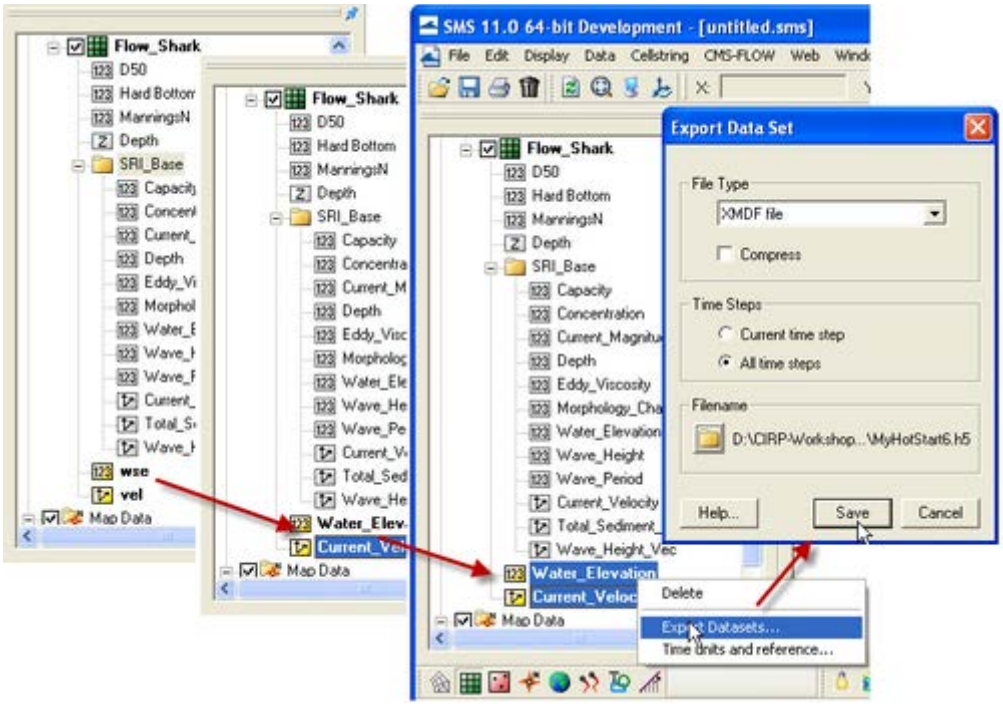


Figure 2-98. Dataset Toolbox showing a time step sample of the water elevation and current velocity datasets for use in a hot start (initial condition) file.

Table 2-107. CMS-Flow card for specifying the initial condition file.

Card	Arguments	Default	Range	Description
INITIAL_STARTUP_FILE	character	none	none	Name of initial condition file. The path must be specified if it is different from the cmcards file.

Output Options

Global Output

Global output refers to the variables that are output on every active cell on the grid. The global output options are specified in *Output* tab of the *CMS-Flow Model Control* window. More information on the global output variables, groups and CMS-Flow cards is provided in the sections below.

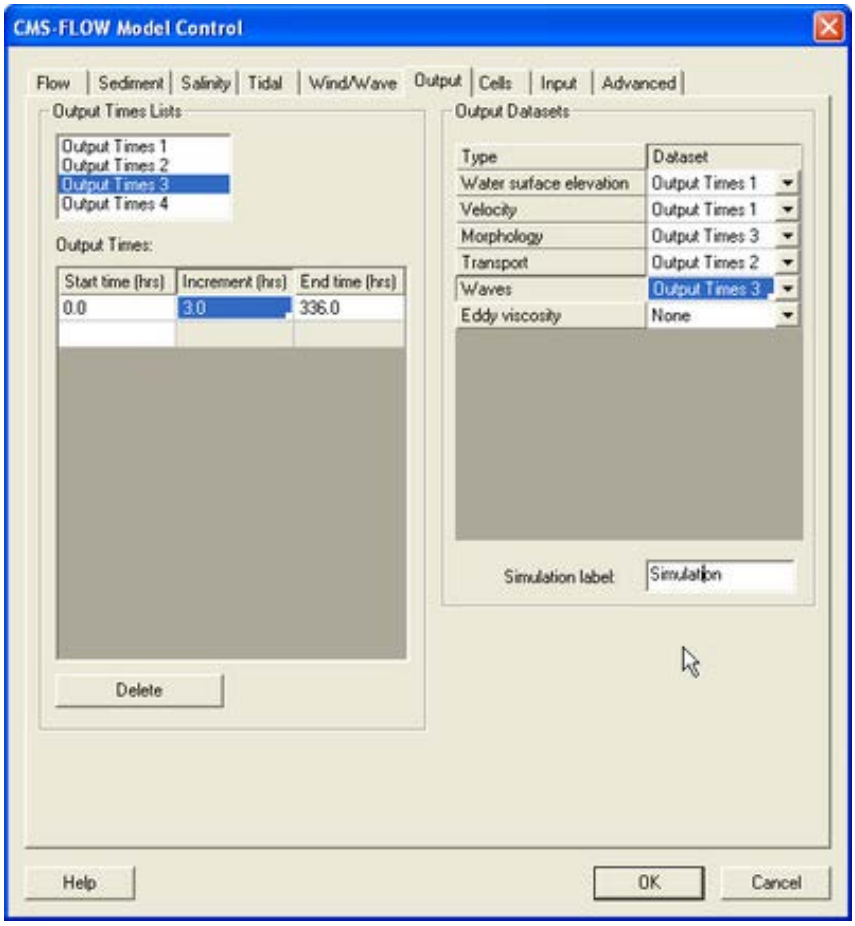


Figure 2-99. *Output* tab in SMS 11.0

Output Datasets

Global output datasets are divided into groups and each group is assigned an output times, and file. A description of the various output datasets and the associated groups is provided in the table below.

Table 2-108. CMS-Flow cards related to the output datasets.

Output Dataset	Group	Description	Units
Current_Velocity	Velocity	Depth-averaged and cell-centered current velocity vector dataset and with respect to local grid coordinates	m/s
Current_Magnitude	Velocity	Depth-averaged and cell-centered current velocity magnitude dataset	m/s
Water_Elevation	Water surface elevation	Cell-centered water surface elevation	m
Eddy_Viscosity	Eddy viscosity	Cell-centered horizontal eddy viscosity	m^2/s
Concentration	Transport	Depth-averaged and cell-centered sediment concentration	kg/m^3
Capacity	Transport	Depth-averaged and cell-centered sediment concentration capacity	kg/m^3
Total_Sediment_Transport	Transport	Depth-averaged and cell-centered total-load sediment transport	kg/m/s
Salinity	Transport	Depth-averaged and cell-centered sediment concentration capacity	kg/m^3
Depth	Morphology	Cell-centered still water depth	m
Morphology_Change	Morphology	Cell-centered morphology (bed) change. Positive is accretion and negative is erosion	m
Wave_Height	Waves	Cell-centered significant wave height	m
Wave_Height_Vec	Waves	Cell-centered significant wave height vector	m
Wave_Period	Waves	Cell-centered peak wave period	s

Output Time Series and Lists

The times at which each group is output is determined by the selecting one of four user defined output time series or lists. In SMS versions 10.1 and earlier, the output time series were used. However, because the output time series can become very large for long-term simulations, the time series have been replaced by lists in which the output times are specifying a list of starting, ending and increments. This option is more compact and also makes it easier to manually change the output options in the cmcards file.

Table 2-109. CMS-Flow cards used for specifying the time series and lists.

Card	Arguments/Format	Default value	Description
TIME_SERIES_1	[length of list 1] [output times for list 1]	0	Output time series for list 1 in hours.
TIME_SERIES_2	[length of list 2] [output times for list 2]	0	Output time series for list 2 in hours.
TIME_SERIES_3	[length of list 3] [output times for list 3]	0	Output time series for list 3 in hours.
TIME_SERIES_4	[length of list 4] [output times for list 4]	0	Output time series for list 4 in hours.
TIME_LIST_1	[number of sublists] [sublist 1: start, end, increment] [sublist 2: start, end, increment]...	0	Sublist(s) for output time series 1. For each sublist, the arguments are starting time, end time and increment in hours.
TIME_LIST_2	[number of sublist] [sublist 1: start, end, increment] [sublist 2: start, end, increment]...	0	Sublist(s) for output time series 2. For each sublist, the arguments are starting time, end time and increment in hours.
TIME_LIST_3	[number of sublist] [sublist 1: start, end, increment] [sublist 2: start, end, increment]...	0	Sublist(s) for output time series 3. For each sublist, the arguments are starting time, end time and increment in hours.
TIME_LIST_4	[number of sublist] [sublist 1: start, end, increment] [sublist 2: start, end, increment]...	0	Sublist(s) for output time series 4. For each sublist, the arguments are starting time, end time and increment in hours..
WSE_OUT_TIMES_LIST	integer	0	Output time series id for water surface elevation in m.
VEL_OUT_TIMES_LIST	integer	0	Output time series id for current velocity and magnitude in m/sec.
MORPH_OUT_TIMES_LIST	integer	0	Output time series id for evolving bed and bed change in m.
TRANS_OUT_TIMES_LIST	integer	0	Output time series id for sediment concentration, capacity and salinity concentration in kg/m^3 and sediment transport rates in m^2/sec.
WAVE_OUT_TIMES_LIST	integer	0	Output time series id for wave height in m, wave period in sec, and wave height vector in m.
EDDY_VISCOSITY_OUT_TIMES_LIST	integer	0	Output time series id for horizontal eddy viscosity in m^2/sec.

Output Group File Specification

For large grids or long simulations, outputting all of the variables to a single XMDF solution file can lead to a file that is unmanageable. To help reduce to the size of the output files each variable group listed in Table 54 can be output into separate files. The table below lists and describes the CMS-Flow cards used to specify the output file names for each variable group.

Table 2-110. Variable group output file name specification.

Card	Arguments	Default	Description
WSE_OUT_FILE	Character	none	Specifies the XMDF output file name for the water level group
VEL_OUT_FILE	Character	none	Specifies the XMDF output file name for the current velocity group
EDDY_VISCOSITY_OUT_FILE	Character	none	Specifies the XMDF output file name for the eddy viscosity group
VISC_OUT_FILE	Character	none	Specifies the XMDF output file name for the eddy viscosity group (same as card above)
TRANS_OUT_FILE	Character	none	Specifies the XMDF output file name for the transport group
MORPH_OUT_FILE	Character	none	Specifies the XMDF output file name for the morphology change group
WIND_OUT_FILE	Character	none	Specifies the XMDF output file name for the wind group

Advanced Global Output

In addition to the variables specified in the SMS interface, CMS has the option to output advanced mode output including the bed shear stress, bed composition, wind speed, etc.

Table 2-111. Advanced output datasets.

Card	Arguments	Default	Description
WIND_OUT_TIMES_LIST	integer	0	Output time series id for wind velocity and magnitude in m/s.
STRESS_OUT_TIMES_LIST	integer	0	Output time series id for mean bed shear stress in Pa.
WAVE_OUTPUT_DETAILS	ON OFF	OFF	Outputs additional wave variables including wave direction, radiation stresses, breaking dissipation and roller energy.

XMDF File Compression

The standard CMS-Flow output is written to an XMDF file with the name <Case Name>_sol.h5. The binary file may be written in compressed format using the card described in the table below.

Table 2-112. CMS-Flow card for compressing the XMDF output file

Card	Arguments	Default	Description
XMDF_COMPRESSION	ON OFF	OFF	Compresses the XMDF file

ASCII Files

In addition to the XMDF output file, CMS-Flow provides the output two types of ASCII output files:

1. Tecplot snap shot (*.dat), and history files (*.his)
2. SMS Super ASCII files (*.sup, *.xy, *.dat)

The CMS-Flow cards used for outputting these two types of files are described in the table below.

Table 2-113. CMS-Flow cards used to output Tecplot and SMS Super ASCII files.

Card	Arguments	Default	Description
GLOBAL_TECPLOT_FILES	ON OFF	OFF	Outputs Tecplot ASCII files
GLOBAL_SUPER_FILES	ON OFF	OFF	Outputs Super ASCII files

Observation Cells (Save Points)

Time series at selected *Observation* cells is set in the *Cells* tab of the *CMS-Flow Model Control* window. A description of CMS-Flow cards used for specifying *Observational* cells are described below.

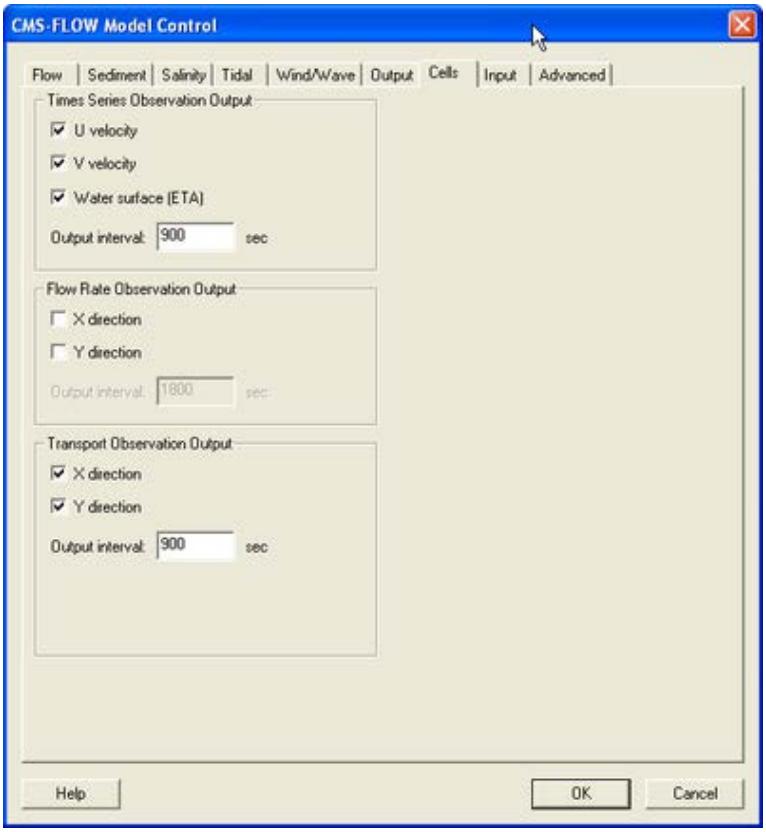


Figure 2-100. *Cells* tab in SMS 11.0

Table 2-114. CMS-Flow card for compressing the XMDF output file

Card	Arguments	Description
TIME_SERIES_INCREMENT	real	Sets the output time increment for the *Time series Observation* points.
ELEV_OBS_CELLS_BEGIN	none	Marks the beginning of a *Times series Observation* point list.
ELEV_OBS_CELLS_END	none	Marks the end of a *Times series Observation* point list.
FLOW_RATE_INCREMENT	real	Sets the output time increment for the *Flow rate Observation* points.
FLOW_OBS_CELLS_BEGIN	none	Marks the beginning of a *Flow rate Observation* point list.
FLOW_OBS_CELLS_END	none	Marks the end of a *Flow rate Observation* point list.
Q_TRANS_INCREMENT	real	Sets the output time increment for the *Flow rate Observation* points.
Q_TRANS_OBS_CELLS_BEGIN	none	Marks the beginning of a *Transport Observation* point list.
Q_TRANS_OBS_CELLS_BEGIN	none	Marks the end of a *Transport Observation* point list.

Simulation Statistics

CMS V4.0 has the option to calculate statistics over the whole model domain for a user-specified time period. This option is accessed using the advanced cards. The starting time, end time, and time interval should be specified in hours with respect to the model start time. The time interval should be larger or equal to the hydrodynamic time step. When activated the statistics are written to the XMDF Global Solution File in a subfolder named *stats*. The CMS-Flow cards related to the statistics are described below.

Table 2-115. CMS-Flow cards related to output statistics

Card	Arguments	Description
GLOBAL_STATISTICS	[start time] [end time] [interval]	Turns on the statistics for hydrodynamics, sediment transport, and salinity.
FLOW_STATISTICS	[start time] [end time] [interval]	Turns on the statistics for hydrodynamics transport.
SEDIMENT_STATISTICS	[start time] [end time] [interval]	Turns on the statistics sediment transport.
SALINITY_STATISTICS	[start time] [end time] [interval]	Turns on the statistics for salinity.

Notes:

- The units of the starting time, end time, and interval are hours.
- If the interval is not specified, the initial time step is used as the interval.

A description of the hydrodynamic, sediment, salinity statistics is provided in the tables below.

Table 2-116. Hydrodynamic Statistics

Statistic Name	Dataset	Description
Maximum Current Velocity Vector	*Current_Max*	Maximum current velocity vector in m/s. Can be either negative or positive.
Maximum Current Velocity Magnitude	*Current_Max_Mag*	Maximum current velocity magnitude in m/s.
Maximum Water Level	*Water_Level_Max*	Maximum water level in m.
Residual Current Velocity Vector	*Current_Velocity*	Average current velocity vector in m/s.
Residual Current Velocity Magnitude	*Current_Velocity_Mag*	Average current velocity magnitude in m/s.
Hydroperiod	*Hydroperiod*	Fraction of time a computational cell is wet (i.e. 0-never dry, 1-always wet).
Maximum water level gradient magnitude	*WSE_Grad_Max_Mag*	Maximum water level gradient magnitude in m/m.
Maximum current velocity gradient magnitude	*Cur_Grad_Max_Mag*	Maximum current velocity gradient
Average *x*-Velocity Normalized L-2 Norm	*Vx_Norm_Res*	Average normalized L-2 norm of the momentum equation in the *x*-direction
Average *y*-Velocity Normalized L-2 Norm	*Vy_Norm_Res*	Average normalized L-2 norm of the momentum equation in the *y*-direction
Average Pressure Normalized L-2 Norm	*Pres_Norm_Res*	Average normalized L-2 norm of the pressure correction equation. The pressure is in units of m^2/s^2.

Table 2-117. Sediment Statistics

Statistic Name	Dataset	Description
Maximum Total-load Sediment Transport Rate vector	*Sed_Transp_Max*	Maximum total-load sediment transport rate vector in units of kg/m/s. Can be either negative or positive.
Maximum Total-load Sediment Transport Rate Magnitude	*Sed_Transp_Max_Mag*	Maximum total-load sediment transport rate magnitude in units of kg/m/s.
Net Total-load Sediment Transport Rate Magnitude	*Sed_Transp_Net*	Net total-load sediment transport rate vector in units of kg/m/s. Can be either negative or positive.
Net Total-load Sediment Transport Rate Magnitude	*Sed_Transp_Net_Mag*	Net total-load sediment transport rate magnitude in units of kg/m/s.
Gross Total-load Sediment Transport Rate Magnitude	*Sed_Transp_Gross*	Gross total-load sediment transport rate vector in units of kg/m/s. Can be either negative or positive.
Gross Total-load Sediment Transport Rate Magnitude	*Sed_Transp_Gross_Mag*	Gross total-load sediment transport rate magnitude in units of kg/m/s.
Maximum Bed Slope Gradient Magnitude	*Bed_Grad_Max_Mag*	Maximum gradient of the bed slope in units of m/m.

Table 2-118. Sediment Statistics

Statistic Name	Dataset	Description
Average Salinity Concentration	*Salinity_Avg*	Average salinity concentration in units of ppt.

Diagnostic File

To Be Completed

```
 ************************************************************
              U.S. Army Corps of Engineers
            Coastal Inlets Research Program
                Coastal Modeling System
            CMS2D, Version  4.10.23 RELEASE
                Last updated - 04/24/2013
         For the latest version of CMS please visit
            http://cirp.usace.army.mil/products/

    By using this software the user has agreed to the
    terms and conditions of CMS license agreement.
    A copy of the license can be obtained from the
    website shown above.
 ************************************************************


CMS-Flow Path:                  C:\CMS\verification\aquaveo_tests\sediment\m
CMS-Flow Card File:             imp_r03.cmcards
Input Version:                  4.00

Grid
  Grid File:                    imp_r03_grid.h5
  Depth Dataset:                SimpleCMS-FLOW/Datasets/Depth
  Orientation:                  360.000 deg
  x-Origin:                            0.000 m
  y-Origin:                            0.000 m
  Average Latitude:             0.00 deg
  Total cells:                   1906
  Active cells:                  1676
  Telescoping:                  OFF
  Horizontal Projection
    Coordinate System:          LOCAL
    Units:                      METERS

Hydrodynamics
  Water Properties:
    Temperature:                15.00 deg C
    Salinity:                   35.00 ppt
      Note: Used for constant viscosity and density if not specified
    Density:                    1025.00 kg/m^3
    Kinematic Viscosity:        1.182E-06 m^2/s
  Timing
    Simulation Start Time:      2013-01-01 00:00:00 UTC
    Hydrodynamic time step:     1800.000 sec
         Note: Same time step for sediment, morphology change and salinity
    Simulation Duration:        720.000 hours
    Ramp Duration:              12.000 hours
  Wetting and Drying
    Drying Depth:               0.050 m
    Water Ponding:              OFF
    Narrow Channels:     OFF
  Coriolis Approximation:       F-PLANE
    Average Latitude:            0.000 deg
    Constant Value:             0.000E+00 1/s
  Turbulence Model:             SUBGRID
    Coefficients
    Constant:                   0.100E-05
    Current Bottom Shear:       0.067
    Current Horizontal Shear:   0.200
  Numerical Methods
    Solution Scheme:            IMPLICIT
    Predictor-Corrector Scheme: OFF
    Temporal Scheme:            TWO-LEVEL
    Advection Scheme:           EXPONENTIAL
```

```
ntime=   22  dt=1800.000, time= 3.96000E+04
 Flow: iter    p_res          U_res          V_res
       20     6.4728E-06     5.7234E-05     4.1863E-05
   WSE(   933)=   1.2777, U(   919)= -0.5228, V(  1134)= -0.2293
 Sediment: iter      Ctk_res       errpbk
             15    0.5781E-08    0.1132E-05
   Ct(   919)=  0.05279, dzb(   922)= -0.00027, dpbk(   922)=  0.00111

 WRITING GLOBAL OUTPUT -    11.0000 hrs.

ntime=   23  dt=1800.000, time= 4.14000E+04
 Flow: iter    p_res          U_res          V_res
       20     2.8944E-06     3.5875E-05     2.2339E-05
   WSE(   930)=   1.5092, U(   918)= -0.3933, V(  1221)=  0.1687
 Sediment: iter      Ctk_res       errpbk
             14    0.1758E-08    0.6407E-06
   Ct(   918)=  0.01274, dzb(   916)=  0.00012, dpbk(   967)= -0.00052

 WRITING GLOBAL OUTPUT -    11.5000 hrs.

ntime=   24  dt=1800.000, time= 4.32000E+04
 Flow: iter    p_res          U_res          V_res
       20     5.5712E-07     1.0700E-05     9.2618E-06
   WSE(   903)=   1.6584, U(   911)= -0.2587, V(   532)= -0.1556
 Sediment: iter      Ctk_res       errpbk
             12    0.1029E-09    0.5364E-06
   Ct(   911)=  0.00074, dzb(   967)=  0.00005, dpbk(   861)= -0.00016

 WRITING GLOBAL OUTPUT -    12.0000 hrs.
```

3 CMS-Wave Model Setup

Spectral waves or wave parameters can be generated for the wave grid forcing, or wind direction and speeds can provide the necessary information for wind- wave generation. Full (directional) spectra can be imported into the SMS for the CMS-Wave, as well as simplified wave parameters (angle, wave height, and period, etc). The spectral energy file format is given in Appendix A, as well as guidelines for using automated programs to generate these files from NOAA/NDBC raw buoy data.

Adding Wave Parameter Generated Spectra to CMS-Wave

1. Click on *CMS-Wave*, *Spectral Energy*, and select *Create Grid* (wave spectra can be imported),
2. Click *OK* for the default spectral properties, and then click *Generate Spectra* to bring up the window to input wave parameters (Figure 101),
3. Open the Excel spreadsheet *44025buoy_199902.xls* and select the wave parameters (1 month, Feb 1999), copy and paste this into the *Generate Spectra – Spectral Parameters* section (Figure 101), click *Generate*, *OK*.
4. Go to CMS-Wave, Model Control, and turn on *Allow wetting and drying* and *Bed friction* (Figure 102),
5. Users can also specify constant or varied forward and backward reflection coefficients in *Settings*.
6. Water level and wind information are optional source as specified under *Wave Source* in addition to the spectral input data.
7. *File*, *Save As*, Wave.sim (selecting the *Save As Type* as a .sim for simulation) in the folder with the CMS-Flow grid.

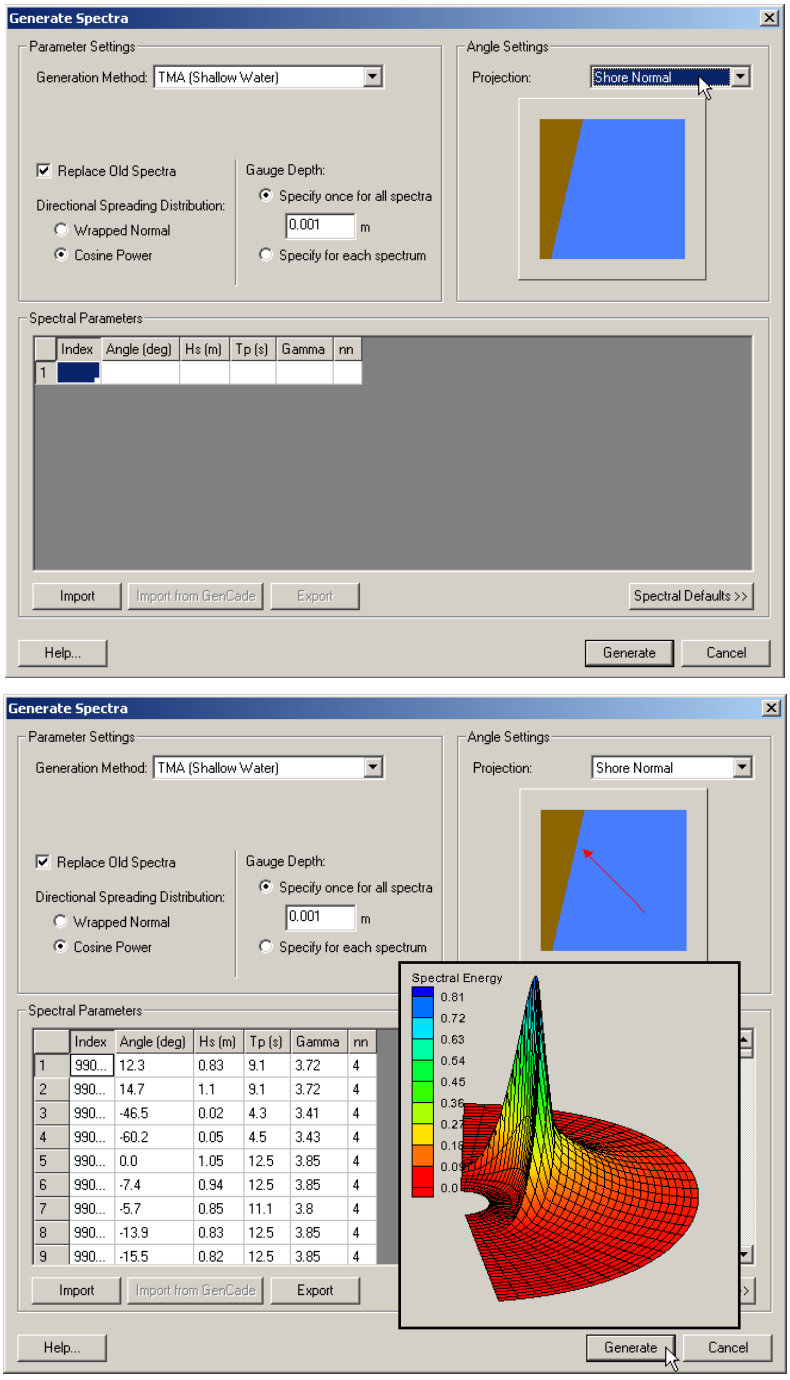


Figure 101. Top Generating wave spectra from wave parameter input; Bottom: Generated wave parameters with a snapshot of spectral output.

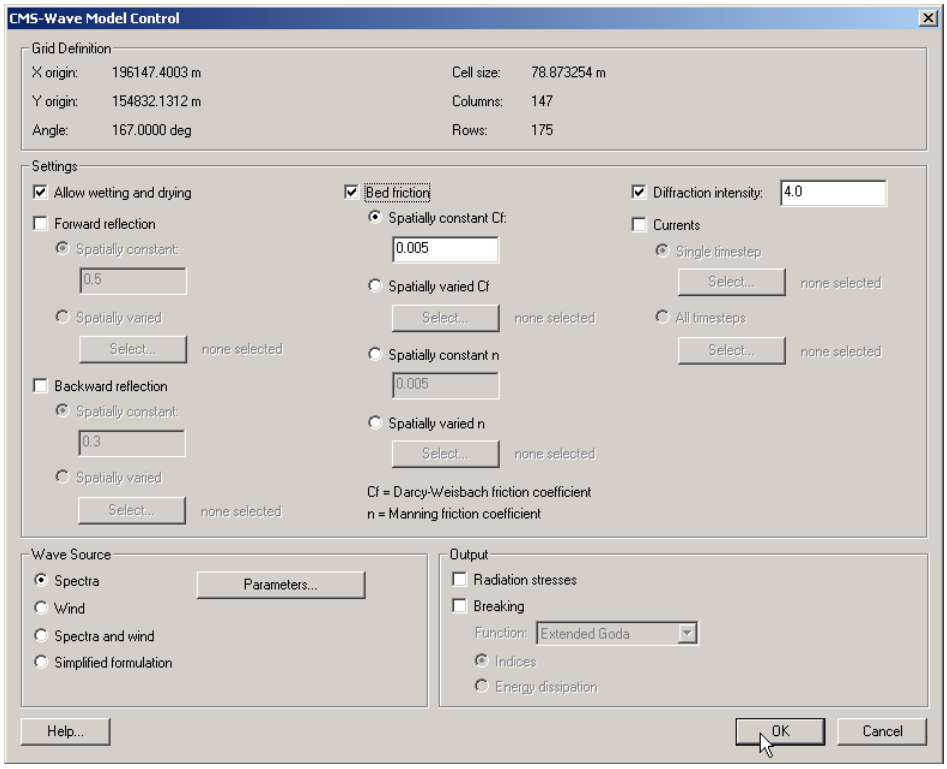


Figure 102. CMS-Wave model control options

Advanced CMS-Wave Features

The most recent CMS-Wave code developed is Version 3-2. Several new capabilities and advanced features in this version include:

- Full-plane
- Automatic wave run-up calculation
- Infra-gravity wave
- Nonlinear wave-wave interaction
- Muddy bottom
- Binary file output
- Selection of multiple processors
- Permeable structure
- Spatially varied wind input
- Spatially varied spectral input
- Grid nesting
- Wave surging (roller) in surf zone

Full-plane

In this mode, CMS-Wave performs two half-plane runs in the same grid. The first run is in the half-plane with the principle wave direction toward the shore. The second run is in the seaward half-plane. Upon the completion of the second run, two half-plane results are combined to one full-plane solution. Because the run time for the full-plane is approximately twice of the regular half-plane, users shall consider the full-plane mode only if the full-plane features like wave generation and propagation in a bay or around an island. An example is to run the Shark River wave case, 2009.sim, in the full plane (modify 2009.std).

Wave Run-up, Infra-gravity Wave, Nonlinear Wave-Wave Interaction, Muddy Bed, Spatial Wind Input

To include (trigger) either of wave run-up, infra-gravity wave, nonlinear wave-wave interaction, binary (xmdf) output, multiple processors, muddy bed, and spatial wind field input is just a one-click step in the SMS11.1 interface. Additional files are required for the muddy bed and spatial wind field input.

If the muddy bed calculation is required, users shall prepare a mud.dat file or *.mud (in the same format as *.dep) to list the spatial maximum kine-

matic viscosity for the entire grid (recommended maximum kinematic viscosity for mud is 0.04 m^2/sec)

If the spatial wind field input is required, users shall prepare a wind.dat file or *.wind (in the same format as *.cur) to provide the x- and y-component wind speed data corresponding to the incident wave conditions in the model grid.

Permeable Structures

Users will need to select and specify permeable structure cells through SMS11.1 CMS-Wave *Assign Cell Attributes* and select *Permeable Breakwater* (see Figure 3-103). In SMS11 or lower version which does not have the permeable structure cell feature, users will need to modify the *.struct to manually assign the permeable structure cells of interest. Recall that each feature cell is described by four parameters, istruc, jstruc, kstruc, and cstruc in a line format in *.struct (CMS-Wave Technical Report CHL-TR-08-13).

istruc = i-th column in the grid

jstruc = j-th row in the grid

kstruc = feature cell identity

= 1, for adding alternative feature or structure (immersed or exposed) without modifying the input depth

= 2, for calculation of wave runup and overwash on beach face or structure, and adjacent land

= 3, for calculation of transmitted waves of a floating breakwater

= 4, for vertical wall breakwater

= 5, for composite or rubble-mound breakwater

= 6, for a highly permeable structure like the pier or bridge

=7, for a low-permeable structure, like the rubble-mound breakwater

cstruc =feature structure characteristic length

= feature structure depth, for kstruc = 1 (assume a land cell if not provided)

= beach/structure elevation above mean water level, for kstruc = 2 (use the input depth if not provided; no effect for cstruc < 0)

= floating breakwater draft, for kstruc =3 (skip if not provided or cstruc < 0.05 m)

= breakwater/structure elevation, for kstruc = 4 or 5 (use the input depth if not provided; immersed if cstruc < 0)

= the permeable portion (>0, the section below the mean water depth) of a high-crest structure for kstruc = 6 or 7

In the Figure 2 example, users can modify 2009.struct to assign South Jetty 6 seaward end breakwater cells as permeable ones. The top 10 lines of the modified 2009.struct is shown below (the number 191 in the first row is the total structure cells in *.struct)

- 191
- 76 110 7 1.5
- 77 110 7 1.5
- 78 110 7 1.5
- 79 110 7 1.5
- 76 111 7 1.5
- 77 111 7 1.5
- 91 10 5
- 92 10 5
- 93 10 5

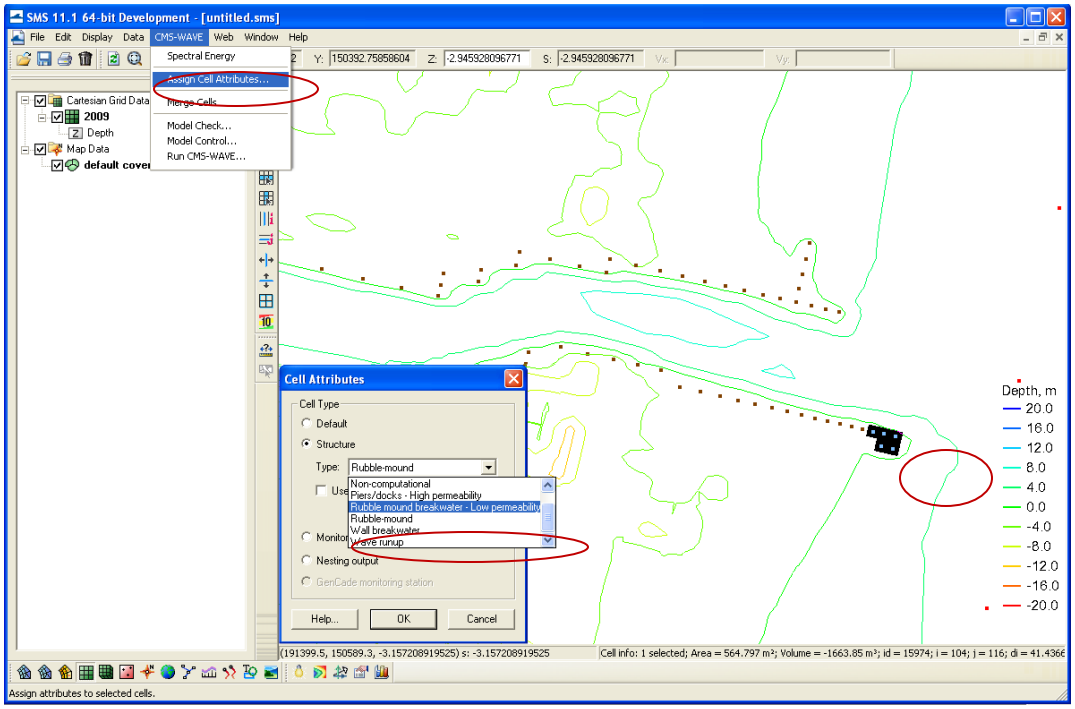


Figure 3-103. CMS-Wave *Assign Cell Attributes* in SMS11.1.

Grid Nesting

Grid Nesting – Users can use the CMS-Wave *Assign Cell Attributes* and *Nesting Output* (Figure 3-103) to specify the wave information output cells for saving spectrum data file (to serve as wave input to a child grid run). Figure 3-104 shows 6 nesting output locations (blue triangle) using the Shark River 2009.sim case. The nesting output file is *.nst (in the case of running CMS steering, an additional file nst.dat is automatically generated that merge all individual cycle *.nst files).

Figure 3-105 shows a child grid domain (c2009.sim) within the parent grid (the child grid was generated based on scatter points converted from the parent grid). The child grid wave input file (2009.nst, as generated from the parent grid) shall be assigned in the child *.std. This can be done by manually editing the child *.std or using the SMS *CMS-Wave* and *Nest Grid* menu (Figure 6-4).

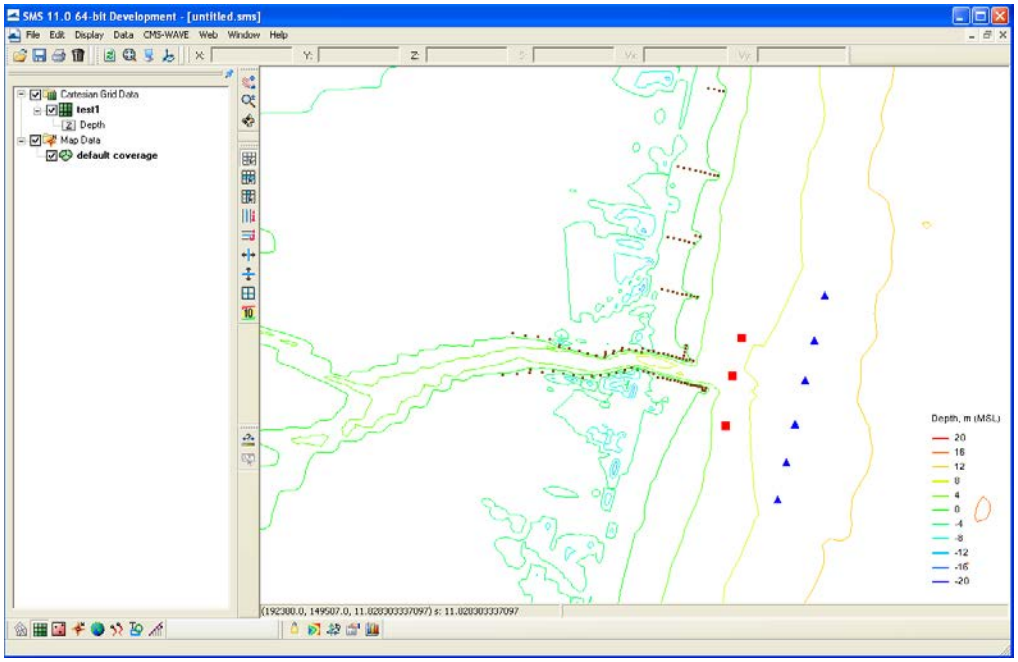


Figure 3-104. Nesting output 6 locations (blue triangle) and monitoring output 3 stations (red square).

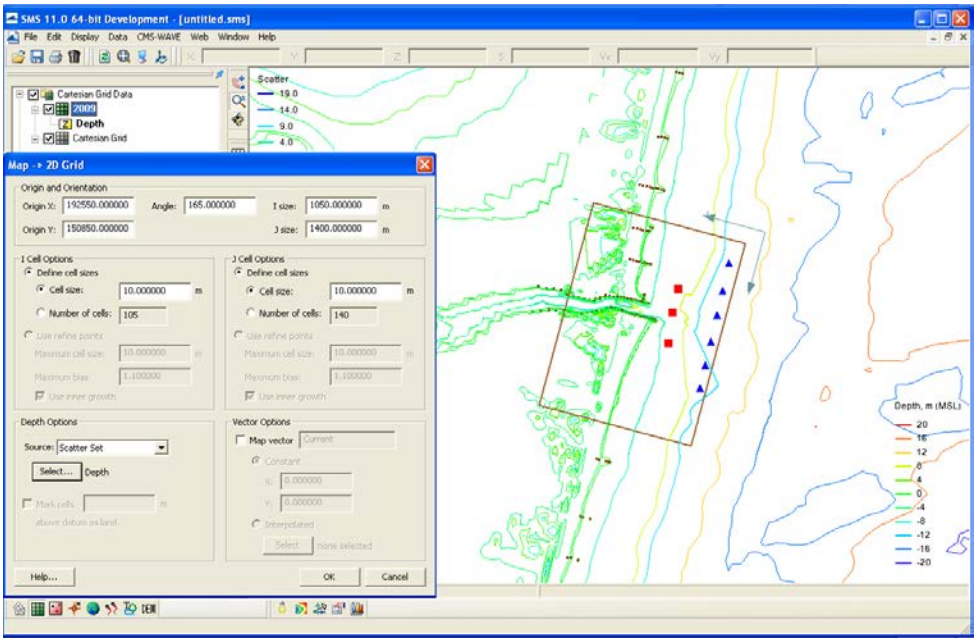


Figure 3-105. The child grid domain and spectral input stations (blue triangle).

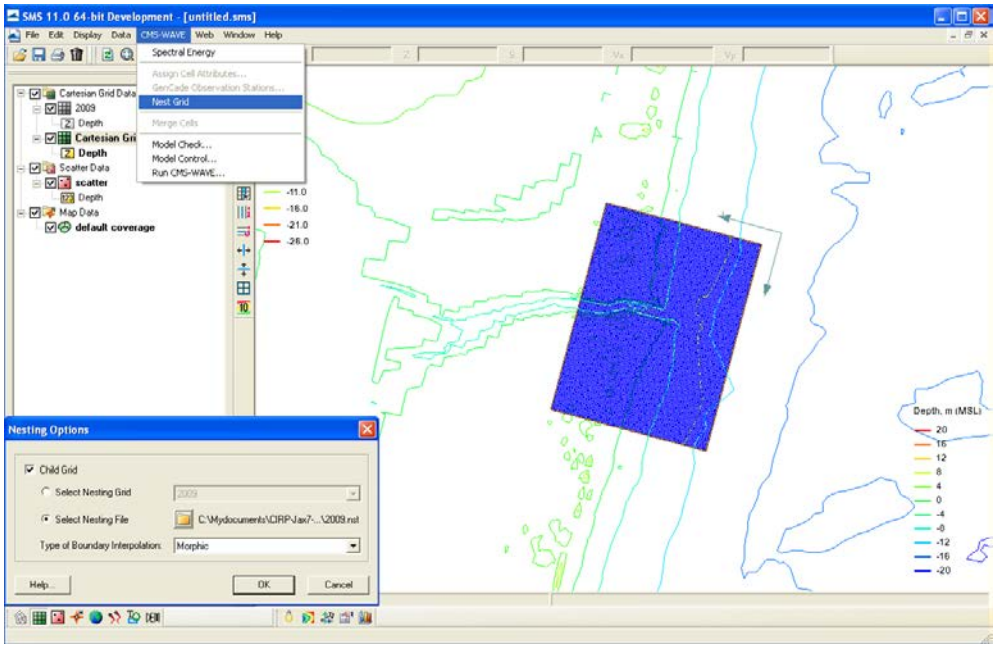


Figure 3-106. CMS-Wave *Nest Grid* and *Nesting Options* menu in SMS11.1.

The child wave input file format is almost identical to the parent *.eng. The only difference is that the child wave input has additional 3 parameters (the local x and y coordinates, and local significant wave height at the spectral wave input location) in the individual spectral header along with the regular 5 parameters (spectral id, wind speed, wind direction, spectral peak frequency, water level adjustment) in the parent *.eng. The top 10 lines of 2009.nst are shown below (notice the 8th line is a spectral header for the 1st individual wave input spectrum):

```
 30       35       6   167.00
0.04      0.05     0.06     0.07     0.08     0.09     0.10     0.11     0.12     0.13
0.14      0.15     0.16     0.17     0.18     0.19     0.20     0.21     0.22     0.23
0.24      0.25     0.26     0.27     0.28     0.29     0.30     0.31     0.32     0.33
  9120103      9.80  -221.0  0.1200  0.00    192440.33    150712.28  0.563
   0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
   0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
```

To run the child grid in the steering mode, the spectral wave input file needs to be renamed to a default “nest.dat” (overwrite the wave input filename in the child *.std). It is noted that this “nest.dat” is only required for the child steering run. The parent grid run must be conducted and com-

pleted first to start a child grid run irrespective of whether CMS-Wave is or is not coupled with CMS-Flow (see more information in ERDC/CHL CHETN-IV-76).

Spatially Varied Spectral Wave Input

Spatially varied spectral input – This is simply the case as in a child grid that spatially varied wave spectra are permitted to assign at user specified locations along or near the seaward boundary of the child grid. To apply spatially varied spectra for wave input without a parent grid, users will need to prepare the wave input file with the format as described in the child grid run.

A FORTRAN program **merge-eng-to-nst.exe** is provided to combine all wave spectra files (*.eng) from individual locations into a single wave input file in the format for spatially varied spectral input to CMS-Wave. Figure 3-7 shows the map of two locations that each location has a wave input files available, 2009-ndbc.eng at Pt 1 (coordinates are 192,602 m and 151,037 m) and 2009-sp154.eng at Pt 2 (coordinates are 192,315 m and 149,579 m) – recall that 2009-ndbc.eng and 2009-sp154.eng were originally generated for the parent grid. Figure 3-8 shows running **merge-eng-to-nst.exe** in DOS to combine two wave input files into one single wave input file (spatially varied spectral wave input to the child grid). Because 2009-ndbc.eng and 2009-sp154.eng were generated respect to the shore-normal direction at 167 deg and the local child grid orientation is 165 deg, a -2 deg direction adjustment is needed in running **merge-eng-to-nst.exe** here.

It is required that all individual wave input files must cover the same period and timestamps (users must edit the files to fill the missing data). In the example, wave spectra at time stamps 09122000, 0912003, and 0912006 are missing in 2009-ndbc.eng, and wave spectra at timestamps 09120400 and 09121000 are missing in 2009-sp154.eng. Two revised files, 2009-ndbc-edit.eng and 2009-sp154-edit.eng (cover the time period from 09120103 to 09123121 in 3-hr interval) are actually used in **merge-eng-to-nst.exe** to generate c2009.nst.

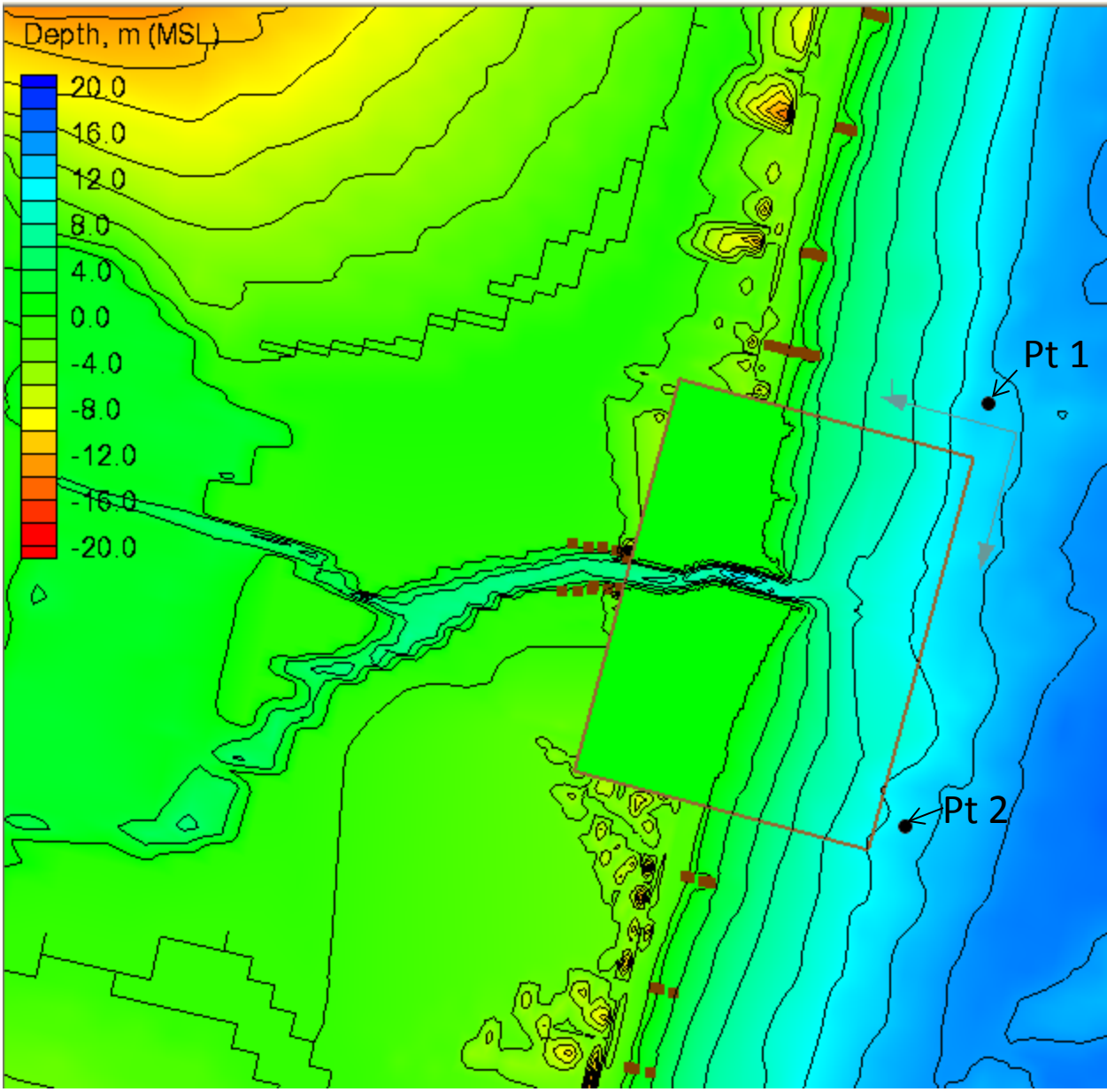


Figure 3-107. Child grid domain and two wave input locations Pt1 and Pt2.

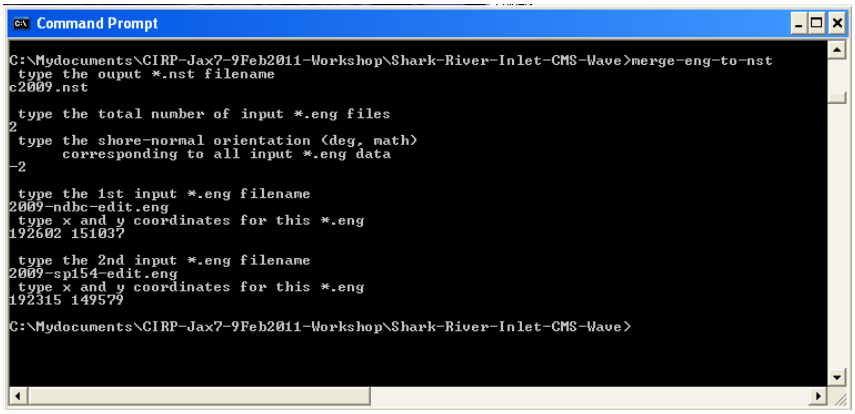


Figure 3-108. Example of running merge-eng-to-nst.exe in DOS.

4 Surface Roller Model Setup

The surface roller model is run after each CMS-Wave and before each CMS-Flow run. It runs on the wave grid but is a separate model and module within the CMS. The surface roller model parameters are set within the CMS-Flow cmcards for convenience and the cards are described in the table below. For more information on the surface roller model see section Surface Roller.

Table 4-1. CMS-Flow cards related to the surface roller.

Input	Format	Notes
Surface Roller Model activation	[card=CALC_ROLLER] [name=**RollModel**, type=*char*, options=(ON,OFF), default=OFF]	Turns On or Off the surface roller model.
Surface Roller Model Dissipation Coefficient	[card=ROLLER_DISSIPATION_COEFFICIENT] [name=**RollDissipCoef**, type=*float*, dependence=(**RollModel**==ON), default=0.1, typical=(0.05><**RollDissipCoef**<0.15)]	Specifies the roller dissipation coefficient. Controls the energy dissipation in the roller model. Higher values produce stronger dissipation and reduce the net effect of the roller.
Surface Roller Model Efficiency Coefficient	[card=ROLLER_EFFICIENCY_COEFFICIENT] [name=**RollEffecCoef**, type=*float*, dependence=(**RollModel**==ON), default=0.5, typical=(0.5><**RollEffecCoef**<1.0)]	Specifies the roller efficiency coefficient. Controls the amount of wave breaking energy which is transferred to the roller. A value of 1.0 indicates that all of the wave breaking energy is transferred to the roller.
Surface Roller Scheme	[cards=ROLLER_SCHEME, dependence=(**RollModel**==ON)] [name=**RollSch**, type=*char*, options=(UPWIND1,UPWIND2), default=UPWIND1]	Specifies the surface roller transport scheme. UPWIND1is a first-order upwind scheme and UPWIND2 is a second-order upwind scheme.

Example 4-1. Specifying the surface roller cards.

```
CALC_ROLLER                     ON !ON | {OFF}
ROLLER_DISSIPATION_COEFFICIENT  0.1  !default = 0.1, range = 0.05-0.15
ROLLER_EFFICIENCY_COEFFICIENT   0.5  !default = 0.5, range = 0.5-1.0
ROLLER_SCHEME                   UPWIND1 !{UPWIND1} | UPWIND2
```

5 Coupling of CMS-Flow and CMS-Wave

Steering refers to the coupling process between CMS-Flow and CMS-Wave. In CMS-Flow versions v3.75 and older (explicit CMS-Flow), the steering was done by the SMS interface. The new inline CMS contains both CMS-Flow and CMS-Wave and performs the coupling process internally. In either, the steering process is similar. First the wave model is run twice at time zero to the first steering interval. The wave information is then interpolated on to the flow grid and the flow model is run from time zero to the first steering interval. The flow information is then interpolated on the wave grid and the wave model is run for the second steering interval and the process is repeated until the simulation is complete.

Before running steering, it is a good idea to test the CMS-Flow and CMS-Wave separately to make sure there are no problems with their grids, or input parameters. Once the CMS-Flow and CMS-Wave models have been setup properly and loaded in SMS, the steering can be initiated.

CMS Versions 4.0 and newer the steering process is done internally by the CMS. This means that that both CMS-Flow and CMS-Wave are contained within a single code or executable. Even though CMS-Flow and CMS-Wave use different grids, the two models are in a single code which facilitates the model coupling and speeds up the computation by avoiding communication files, variable allocation and model initialization at every steering interval. The inline CMS can be launched from the SMS Steering Wizard or as a command line with arguments specifying the input files and steering options. The table below describes the CMS-Flow cards used for the steering process in the inline CMS.

Table 5-1. CMS-Flow cards related to steering

Card	Arguments	Default	Range	Description
CMS-WAVE_SIM_FILE WAVE_SIM_FILE	character	none	none	File name including path for the CMS-Wave sim file.
STEERING_INTERVAL	real	none	none	Sets the recurring hot start output time.
WAVE_WATER_LEVEL	character	TIDAL_PLUS_VARIATION	LAST TIDAL TIDAL_PLUS_VARIATION	Determines the method used to calculate the water levels passed to the wave model.

FLOW_EXTRAPOLATION_DISTANCE	real	Calculated based on grid geometry	none	Determines the extrapolation distance used for flow variables on the wave grid.
WAVE_EXTRAPOLATION_DISTANCE	real	Calculated based on grid geometry	none	Determines the extrapolation distance used for wave variables on the flow grid.

Notes:

1. For both the SMS steering and inline steering, the CMS-Wave input spectra need to be spaced at regular time intervals and begin at the same time as the CMS-Flow model.
2. The way variables are interpolated and extrapolated both in space and time are slightly different between the SMS steering and inline steering versions of CMS.
3. Currently, the inline version of CMS only contains the implicit CMS-Flow solution scheme. Therefore, if the user decides to switch from explicit to implicit solvers, the user must also use different executables.
4. The wave grid for most inlet and coastal cases will be not much longer alongshore than the flow model. If the wave model does not extend far enough, there are model parameters to smoothly interpolate the wave data to some distance alongshore in the flow model.
5. Because the grid boundary is often the location of a wave buoy, the cross-shore domain tends to extend further.
6. In some cases the best IJ location for the half-plane model may not be apparent, especially for open coasts. Often the degree of energy in the open ocean is dominantly from one direction or the other, and the i-direction should be aligned to that direction.
7. The maximum length of the steering interval is a function of the tide (i.e. range and type of cycle such as diurnal, semidiurnal, or mixed), wave conditions, and nearshore bathymetry. In general, larger tide ranges and semidiurnal tides require shorter steering intervals. Larger wave heights and storm events require smaller steering intervals. Nearshore bathymetry is gentle slopes also requires shorter steering intervals because it small variation in water depth can lead to a large change in the location of the breaker.

Interpolation Files

When running the inline CMS with flow and waves, the CMS steering module write out two files named:

- Intpcoef_flwav.bin
- Intpcoef_wavfl.bin

These files contain the interpolation information between the CMS-Flow and CMS-Wave grids. Because calculating the interpolation information can take several minutes, saving this information in files allows the model to quickly read this information and avoid their computation for subsequent runs when using the same CMS-Flow and CMS-Wave grids. When the model is restarted it will automatically detect these files and read them if the grids are the same size. If changes have been made to the grids but the grid size is the same, the steering module will not be able to detect the changes and the interpolation information will be incorrect. Therefore, it is recommended to delete the interpolation files every time a change is made the either the CMS-Flow or CMS-wave grid. In the future, this problem will be avoided by writing a counter to the CMS-Flow and CMS-Wave files every time a change is made to them from the interface.

6 Running CMS

Recommended Folder Structure

Any folder structure may be used. However, for purposes of organization and ease of use, it is recommended that a folder structure similar to that proposed here be used. For large modeling projects with many alternatives and simulations, a more complex folder structure may be warranted, but should be based on that below.

While the CMS accepts file names and paths with spaces, other software may have issues with spaces. Therefore, it is recommended that spaces are not used in the simulation path and filename without prior testing.

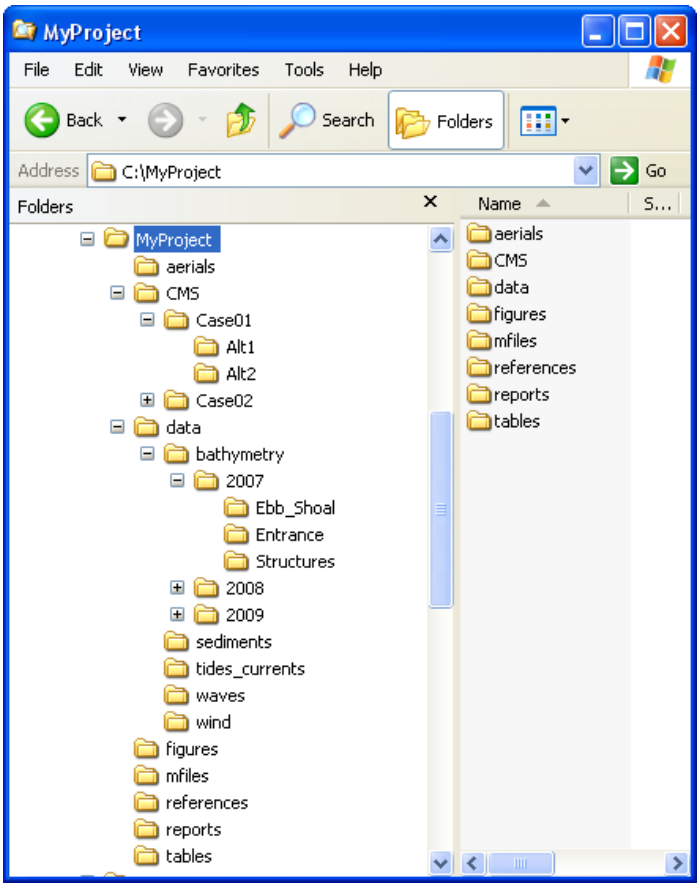


Figure 6-1. Recommended folder structure for CMS projects.

SMS CMS-Flow Menu

The first option for launching CMS versions 4.0 and newer is by simply running from the SMS CMS-Flow menu. If necessary the steering information can be entered in the advanced cards and CMS-Flow will automatically call CMS-Wave and perform the model coupling internally. The steps for running CMS-Flow from the SMS CMS-Flow Menu are outlined below.

- Make sure both the CMS-Flow and CMS-Wave grids are loaded in SMS.

- Check the CMS executable file name under the SMS preferences menu.
    1. Click on *Edit* | *Preferences*.
    2. Under the *File Locations* tab, in the section called *Model Executables* check the file names for CMS-Flow and CMS-Wave and make sure they are consistent with the latest releases (http://cirp.usace.army.mil/products/index.html CIRP Products).

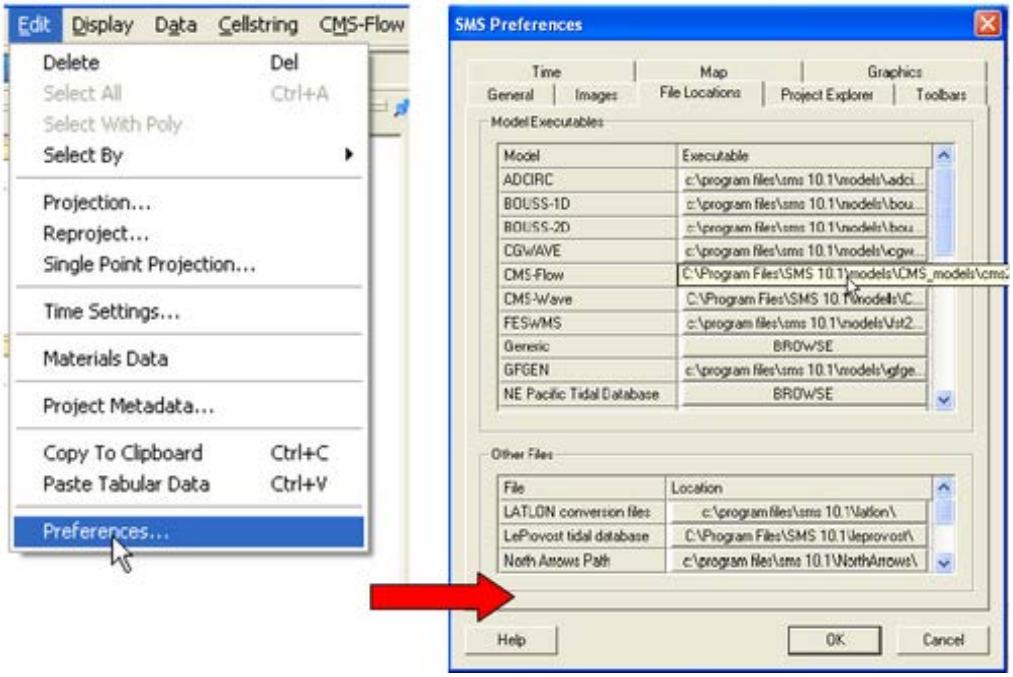


Figure 6-2. Changing the CMS-Flow model executable.

- Start CMS from the CMS-FLOW Menu
    1. Click on *CMS-FLOW | Run*
    2. In the *SMS Steering Wizard* select the *CMS INLINE* option and click on the *Next>* button.
    3. Enter the *Steering Interval* under the *Time* section and click on the *Start* button.

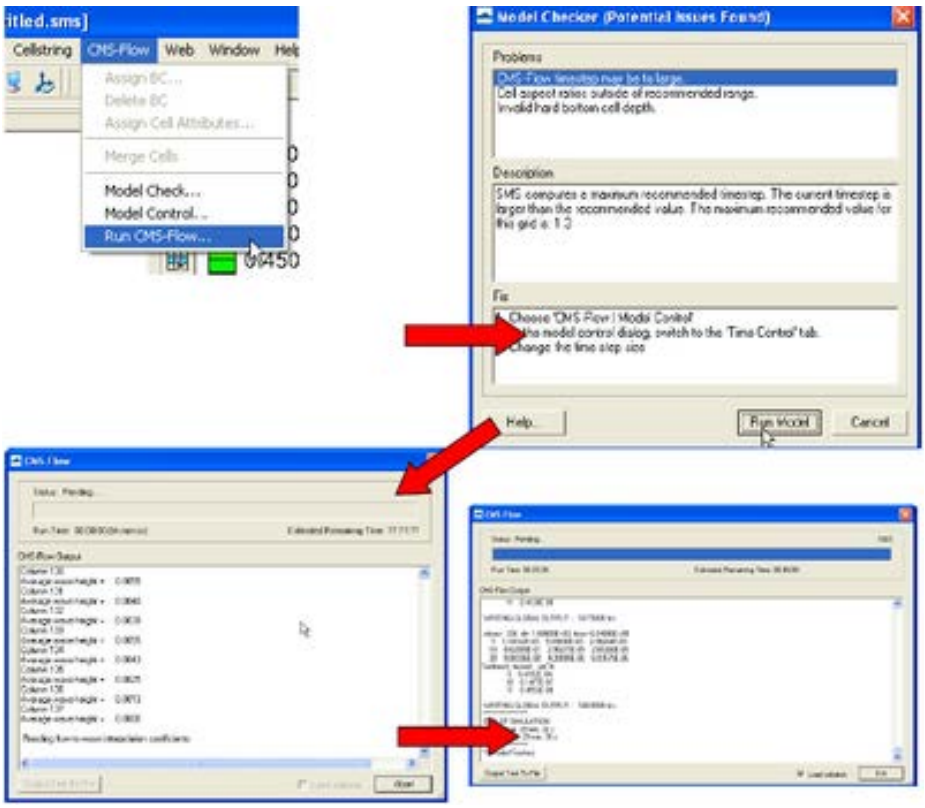


Figure 6-3. Example of launching the CMS from the CMS-Flow menu.

Standalone Program

Standalone refers to the fact that the CMS executable is not part of a larger software package nor does it require a network connection, and support or services of the operating system or other software. If the CMS were run from the SMS it would not be considered a standalone program.

Because the inline CMS does all of the steering internally, it is a standalone program and there is really no need for CMS to be launched from SMS. Running CMS outside of SMS is useful because it allows the user to launch CMS from script files and also to pause the model for checking model results during the model simulation without causing access errors. When running several models, pausing the some of them will free up some of the computer to do other tasks such as plotting and checking model results. Running CMS outside of the SMS, also avoids the extra memory and work requirements from the interface.

There are three mean approaches for running CMS as a standalone program

1. From a command prompt
2. Double-click on the CMS executable or shortcut to the executable.
3. Drag-and-drop the CMS input files on the CMS executable or shortcut to the executable.

Notes:

1. For advanced users, it is recommended to put a copy of the CMS executable in the project directory and running the CMS from a command prompt. This keeps a record of the executable used for the project, facilitates making and transferring script files for running multiple project alternatives and keeps the window open after the model has completed or even crashed.
2. To pause the simulation, press the "Pause/Break" button on your keyboard or press and hold the “Ctl” key and press the “S” key.
3. To stop the model simulation, press and hold the "Ctrl" key and press the "C" key.

Opening a Command Prompt

The first step is to open a command prompt. There are two ways of doing this:

The first method for opening a command prompt is

1. Click on Windows *Start* | Accessories | Command Prompt

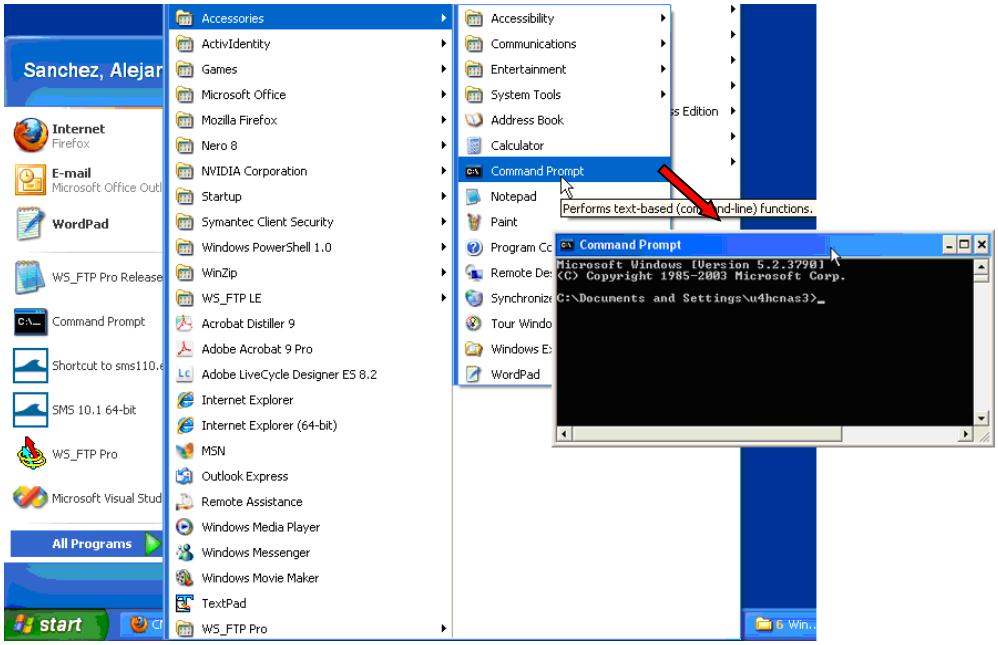


Figure 6-4. Example of launching the inline CMS-Flow steering run.

The second method for opening a command prompt

2. Click on Windows *Start* menu then the *Run...* utility
3. In the *Run* window, enter the command *cmd* and click *OK*

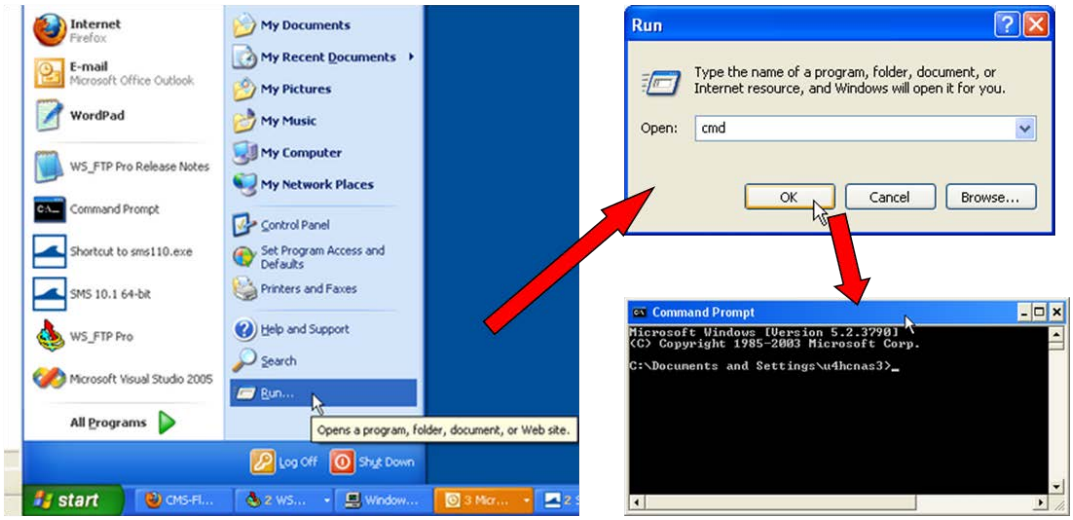


Figure 6-5. Example of launching the inline CMS-Flow steering run.

Note:

- It is recommended to put a shortcut to the Command Prompt on the desktop to more easily open the Command Prompt.

Once the command prompt is open the CMS can be launched using one of the following syntax

>> [cms2d_*exe] [*.sim or *.cmcards file] [*.sim or *.cmcards file] [steering interval] [wave water level option]

where the steering interval is in hours and the wave water level option is either

1. Wave water levels are estimate as the last water levels from the flow model (i.e. WAVE_WATER_LEVEL option equal to LAST)
2. Wave water levels are estimated as the mean tidal water level at the wave time step (i.e. WAVE_WATER_LEVEL option EQUAL TO TIDAL).
3. Wave water levels are estimated as the mean tidal water level at the wave time step plus the water surface variations estimated from the last flow time step (i.e. WAVE_WATER_LEVEL option EQUAL TO TIDAL_PLUS_VARIATION).

Notes:

- The *.sim and *.cmcards files must contain the full or relative path with respect to the executable if different from the executable.
- If no input arguments are specified, than the user will be prompted to manually enter the name of the CMS-Flow and CMS-Wave files and steering information.
- If the cmcards file is specified the CMS will check for the steering cards. If the sim file is found, than it will run in steering.
- If no steering interval is specified in the cmcards file or the command line, than a default value of 3.0 hours will be used.
- If no wave water level option is specified in the cmcards file or the command line, than a default method equal to three.
- It is possible to create a short-cut to the model executable and simply drag-and-drop the cmcards file and or sim file with the steering options specified in the cmcards file.

Below are some examples

```
>> cms2d_v4p1r19-x64p.exe
>> cms2d_v4p1r19-x64p.exe Flow.cmcards
>> cms2d_v4p1r19-x64p.exe Flow.cmcards Wave.sim
>> cms2d_v4p1r19-x64p.exe Wave.sim Flow.cmcards
>> cms2d_v4p1r19-x64p.exe Flow.cmcards Wave.sim 1.0
>> cms2d_v4p1r19-x64p.exe Wave.sim Flow.cmcards 3.0 1
```

Drag-and-Drop

To launch CMS as a standalone application using the drag-and-drop method:

1. Select all of the CMS input files by holding the "Ctrl" key and single-clicking on each input file.
2. Drag all of the input files on the CMS executable or shortcut to the executable by holding the left mouse button.
3. Drop the files by letting go of the left mouse button.

Note:

- If running steering, make sure all of the steering options are in the cmcards file when using the method.

Double Click

This is one of the easiest ways of running the CMS but also one of the most time consuming because it requires the user to type the name and path (if different from executable) for all of the input files and if necessary the steering options.

- To run the CMS by using the Double-Click method:
  1. Double-click on the executable
  2. Follow the instructions on the screen.

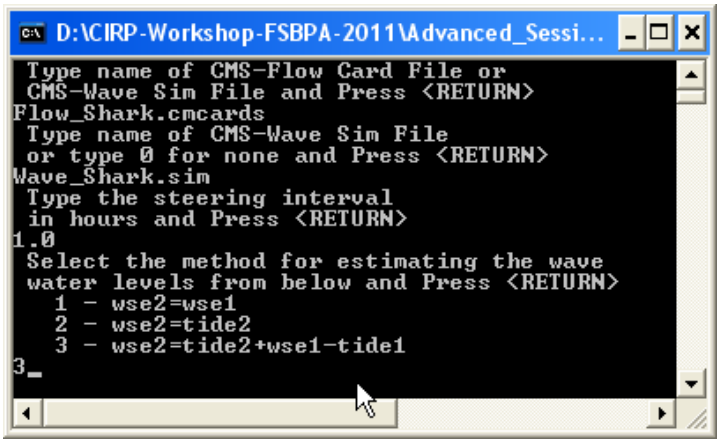


Figure 6-6. Example of launching the inline CMS by double-clicking on the executable.

Note:

- If CMS input files are not in the same folder as the executable, the file

Windows Priority Levels

In Windows NT/2000/XP/7 it is possible to assign a process a different priority level using the Task Manager. This is useful for running CMS in the "background" without slowing down the computer. Windows NT offers three different priority levels while Windows 2000/XP/7 offers five. The priority level can be set when launching the executable from a Command Prompt or batch file or changed after the model is started from the Task Manager. To set the priority level from the Command Prompt use one of the following switches /low, /belownormal, and /abovenormal.

When initiating CMS simulations from a batch file, precede each of the lines in the above example with “start "CMS" /wait /low” as shown below. This initiates a separate Console Window for each simulation on a low priority. You can also see which simulation is active by viewing the primary Console Window. The /wait option is necessary to force the next simulation not to start until the current one is complete. It is not recommended to use /high priority since this may cause the machine to freeze up. To change the priority level of simulation manually, open Task Manager (see your System Administrator if you’re not sure how to do this), click on the Processes Tab and find the CMS executable process you wish to change, right click on the process, choose Set Priority, then set the priority.

7 Viewing and Post-Processing Results

Calibration of hydrodynamic and morphologic datasets requires use of post-processed data in order to compare to measurements and make incremental changes. This section gives guidance on how to process or extract data from model results, and examples of measured data to compare results with. Of course, the post-processed data can be exported and comparisons can be made in another program, but there are some options within SMS that can help with repetitive testing and comparisons.

Water Levels & Currents

The first stage in calibration of a numerical hydrodynamic model is to accurately represent measured tides (or water levels) and currents. The number of calibration measurement sites required for sufficient model calibration increases with larger domains and increased complexity in processes.

Of the three main data sources required for good hydrodynamic calibration (geomorphic, forcing, and field data), inaccurate knowledge of, or mistakes in conversion between datums is the most common underlying factor in poor model calibration. Model calibration cannot be achieved without proper representation of datums. Some examples of field data typically used for calibration of hydrodynamics include fixed water level gauges, fixed acoustic Doppler velocimeters (ADV) and acoustic Doppler current profilers (ADCP), and roaming or boat-mounted ADCPs. Both fixed and roaming data can be assessed over a time series, where boat-mounted data are collected at specified locations over time intervals. However, roaming data that were collected continuously must be assessed as a function of location in three-dimensional space.

For hydrodynamic data, the most common forms of measurements are recorded either temporally at a station, temporally at a fixed spatial extent, or spatially at relatively singular times. For example, a tide gage is stationary, but a horizontal acoustic Doppler current meter (H-ADCP) may collect temporal measurements at multiple points. Of course, the multiple locations are depth-averaged, as would be an upward-looking ADCP, in order to compare to the 2D CMS. Similarly, a field campaign of depth-

averaged current measurements collected at transects are all contemporaneous (within reason), but there are multiple time periods to compare.

With good forcing data and accurate corresponding datums, the final check before hydrodynamic calibration can be achieved is the accuracy and detail of the geomorphic parameters such as bathymetry, sediment characterization, and structure definition. The most important of these is bathymetry, which can have several issues which are typically associated with quality control, density, interpolation, and final grid resolution and domain size. A good example of this is a poorly resolved small channel in the far reaches of a bay, such as that in the northern portion of the Shark River estuary (Figure 4-51). Although lower velocities may be passing through this channel, and there may be little interest in morphology change here, it may be integral to providing the conduit of tidal prism to a shallow bay platform.

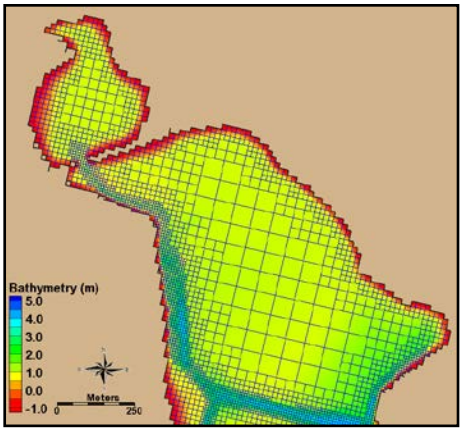


Figure 7-1. Northern portion of the Shark River estuary.

Figure 4-53 shows the closest locations of water-level gauges to Shark River Inlet. Sandy Hook and Atlantic City, NJ, are two ocean pier-mounted gauges approximately 30 and 100 miles away from Shark River Inlet, respectively. Both NOAA gauges were evaluated against the Belmar tide gauge using the tidal constituents of one year, and Sandy Hook was found to have the closest amplitudes and phases. Measured tides from Sandy Hook were used to force the offshore boundary condition. The calibration period that the model was evaluated on was for the 13-hour tidal cycle over which field measurements were collected on 20 August 2009.

Figure 7-2. Location of NOAA (red) and USGS (blue) water level gauges.

Display Options

The results of the CMS calculated hydrodynamics can be viewed and compared to measured results in a number of ways. Scalar results are typically in planar view and are illustrated by color representing magnitude, and arrow lengths representing magnitude and direction. Color setting can be manipulated in the overall Display Options, or individually for each dataset.

1. Open a solution file for the hydrodynamic output and right click on the *WSE* dataset, select *Display Options*.
2. In the *Contours* tab, adjust the color settings and the value range to that in Figure 4-53.

Figure 4-54 illustrates the new color scheme for displaying the water surface elevation.

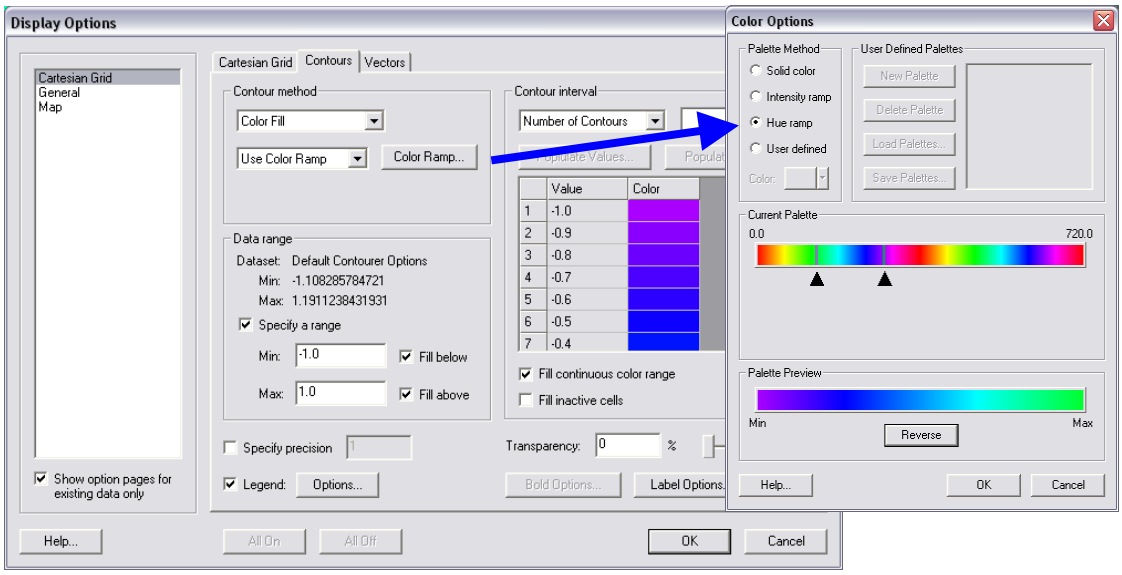


Figure 7-3. Contour tab settings for color display and value range. Note the blue colors were selected for the visible range.

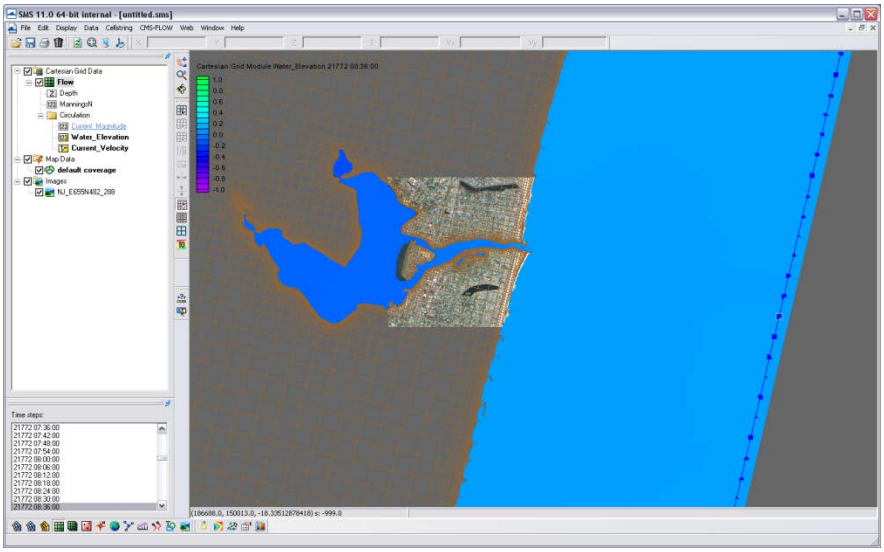


Figure 7-4. Water surface elevation color contours modified.

Currents are automatically given in vector format as the solution files are generated. These can be converted into Vx and Vy components, or into magnitude and direction. If the modeled results are compared to simple magnitude measurements, then the magnitude dataset is appropriate. If the modeled results will be compared to measurements collected in easting and westing, care should be taken to compare the correct derivative of

the vector format. Because the vectors are based on the Cartesian grid's orientation, be sure to convert with respect to the axis difference.

1. Right click on the current vector dataset and select *Vector to Scalar*. Select Vx and Vy, and this gives the x and y components of the current velocity.
2. Next, convert the vector to *Magnitude and Direction*. The color component of the magnitude can be selected along with the vector of the current velocity.
3. Open the *Display Options*, and go to the *Vector* tab. Change the vectors to the below settings (Figure 4-55) to view the vectors over a prescribed gridding. (Note that individual vectors at each cell is difficult to visualize for the whole domain and often only useful when zoomed in very close.)

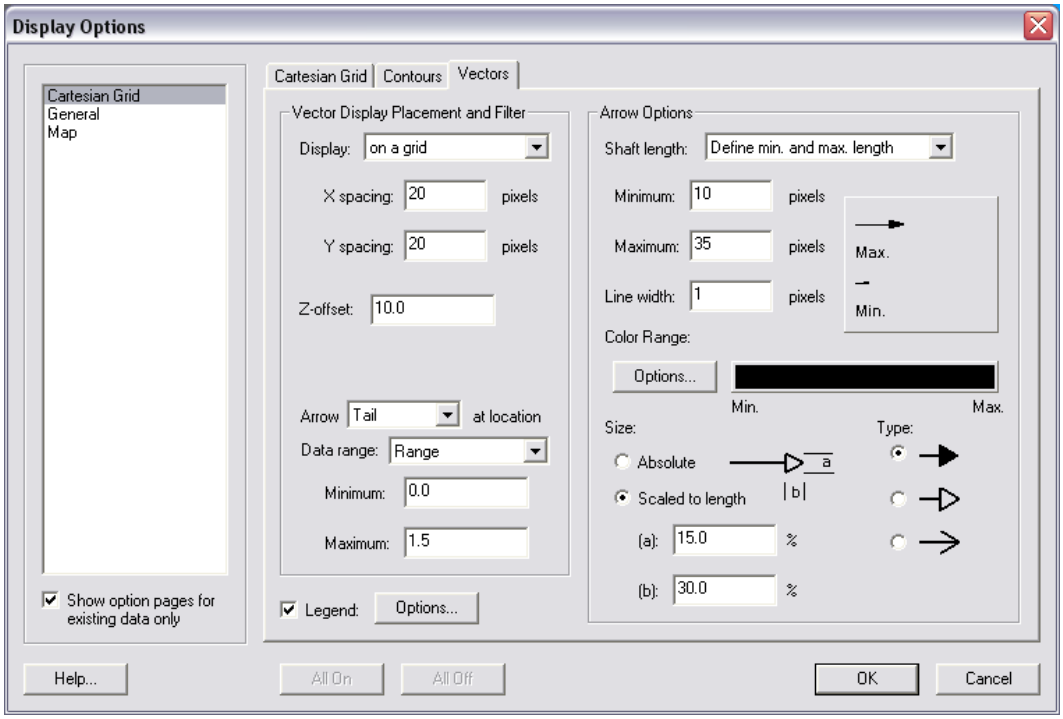


Figure 7-5. Vector Display options set for a gridded display across the domain.

Figure 4-56 shows an example of displaying current magnitudes with color contours and the current direction with vectors.

Figure 7-6. Vectors and color contours set for displaying current magnitude (color) and direction (vectors). Note that the vectors are slightly scaled on this setting to reflect magnitude.

Importing water level measurements into SMS for comparison

To compare the measured data within SMS, the measurements must be brought into the program and into a comparable format. SMS will import a variety of ASCII, columnar data that can be categorized and labeled in a scatter dataset (scatterset).As a simple example of comparing tidal data to calculated tides, the USGS Belmar tide data is given in the folder Hydro\Water Level Calibration in the Tide_Comparison_AUG09_Workshop.xls file. This is the only tide measurement for Shark River Estuary to compare the bay tidal harmonics. To import this data, it must be brought in as observational data within the *Observation Map Module*.

1. In the Cartesian Module, go to *Display*, *Plot Wizard*, and select *Time Series Plot*. Select scalar, and the WSE as the dataset. (Note it takes some time to populate a plot in SMS.)

2. Change the map module *Default Coverage* type to *Observation*, and add a *Feature Point* to the approximate location of the tide gauge shown in Figure 4-57.

3. Right click on this feature point (Figure 4-57), and select attributes.

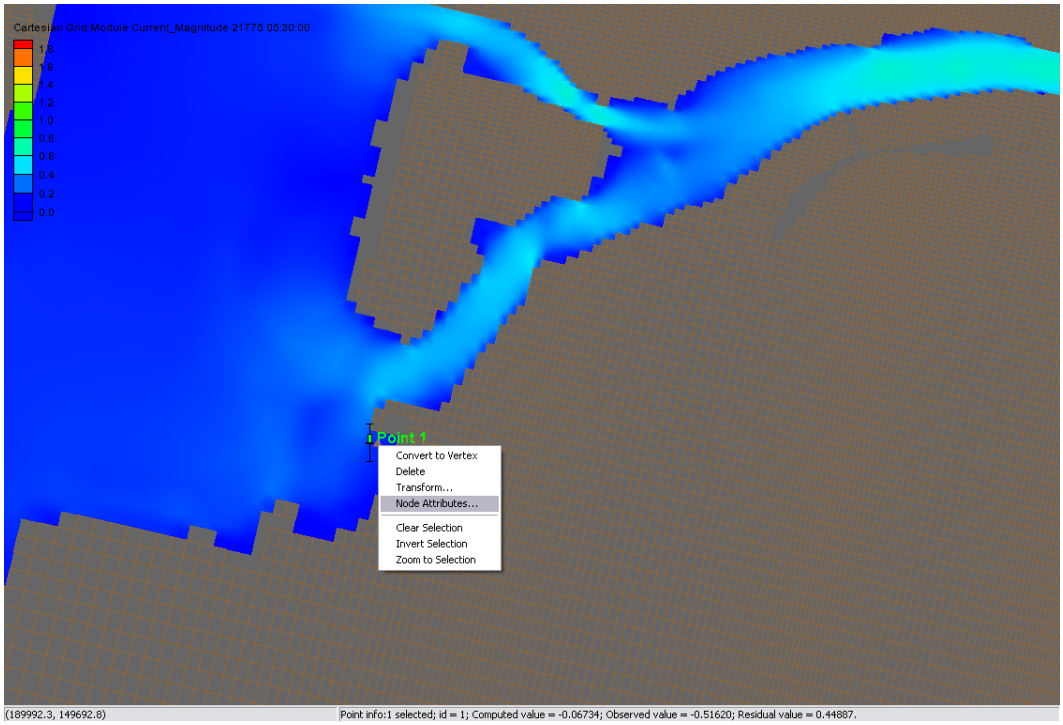


Figure 7-7. Observation point placed at the location of the Belmar tide station.

4. In the *Attributes*, check the Trans box in measurements, and select 2D Cartesian under Module, and the Water_Elevation dataset (Figure 5-58). If your Observe Point is checked, an Options box should appear under the Time Series column. Select this and import the Belmar_Tide.xys file.

Time series of the measured water levels can be manually pasted here. A file of measured water levels is provided and available for *Import*: Belmar Tide Input.tsd.

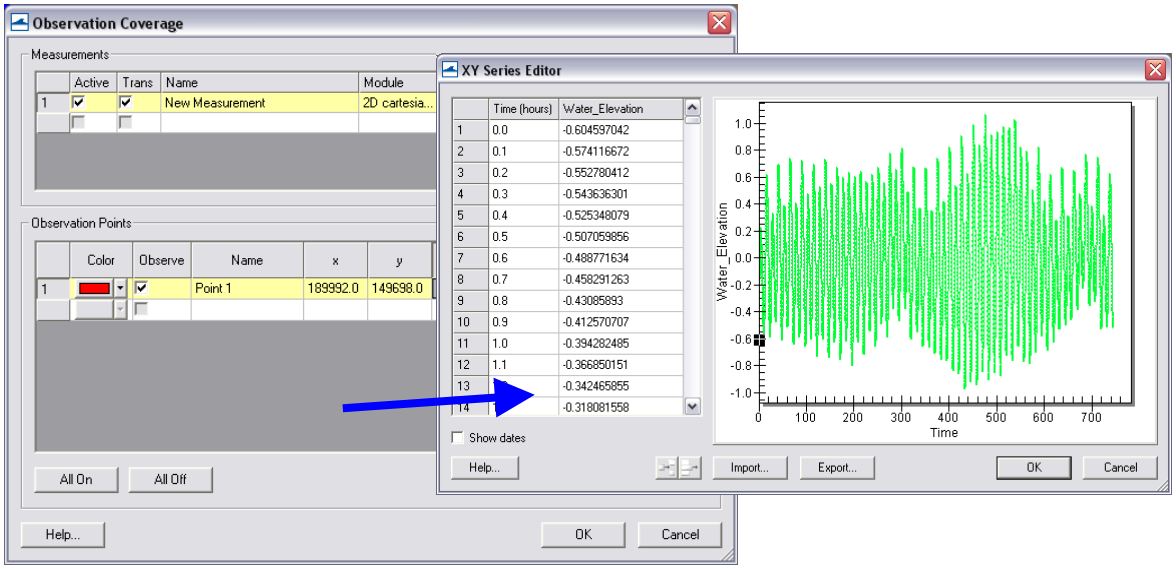


Figure 7-8. Input of time series water surface elevation data into an observation point.

Water level data plotting from Feature Points

Time plots can be compared within SMS through their software plotting package. To create a plot:

1. Select *Display*|*Plot Wizard*|*Time* Series and select *Next*. The *Function* is *Scalar*, and the *Water Elevation* scalar dataset should be selected with the full time period (Figure 4-59).

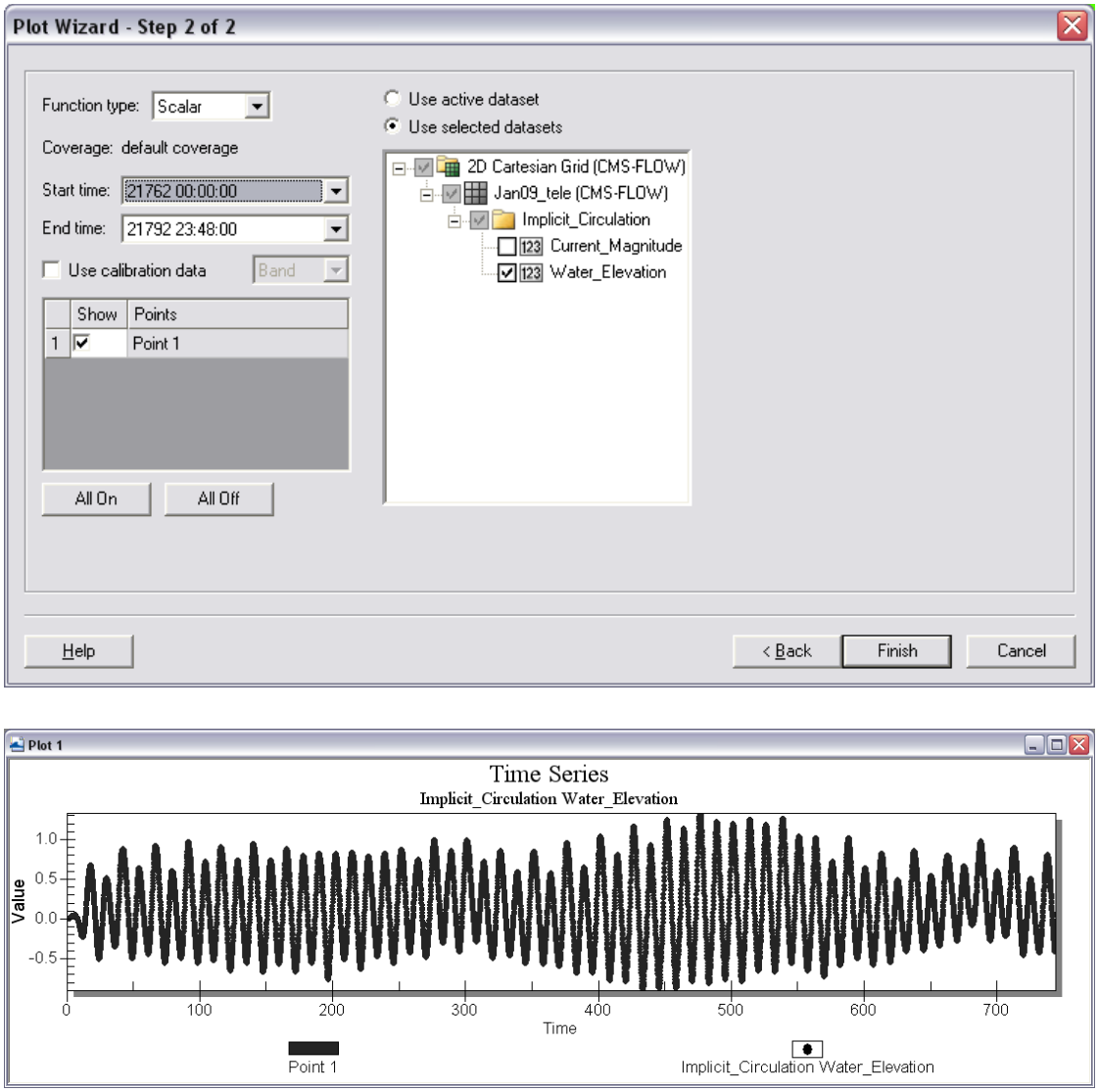


Figure 7-9. Plot Wizard for time series data (top); Time Series plot of Water Elevation (bottom).

2. The plots can be updated by right-clicking the graph window and selecting *Plot Data*. Also, the time frame can be modified.

3. Changing the location of the Feature Point by dragging the point will also automatically update any graph open within SMS. Note that moving Feature Points that are extracting large datasets may take a considerable amount of time to reload.

Extensive statistical analysis of time series water levels is not available in SMS. However, it is convenient to extract this information from SMS which can be used in other software or Microsoft Excel.

4. To extract the raw data for use in other software right-click on the graph window and select *Export*. In the *Exporting Time Series* window, select *Export|Text/Data*, *Export Destination|File* (and specify a location), and select *Export*. Change the *Export Style* to *Table* and *Row vs. Column* to *Points/Subsets* (Figure 4-60). Export.

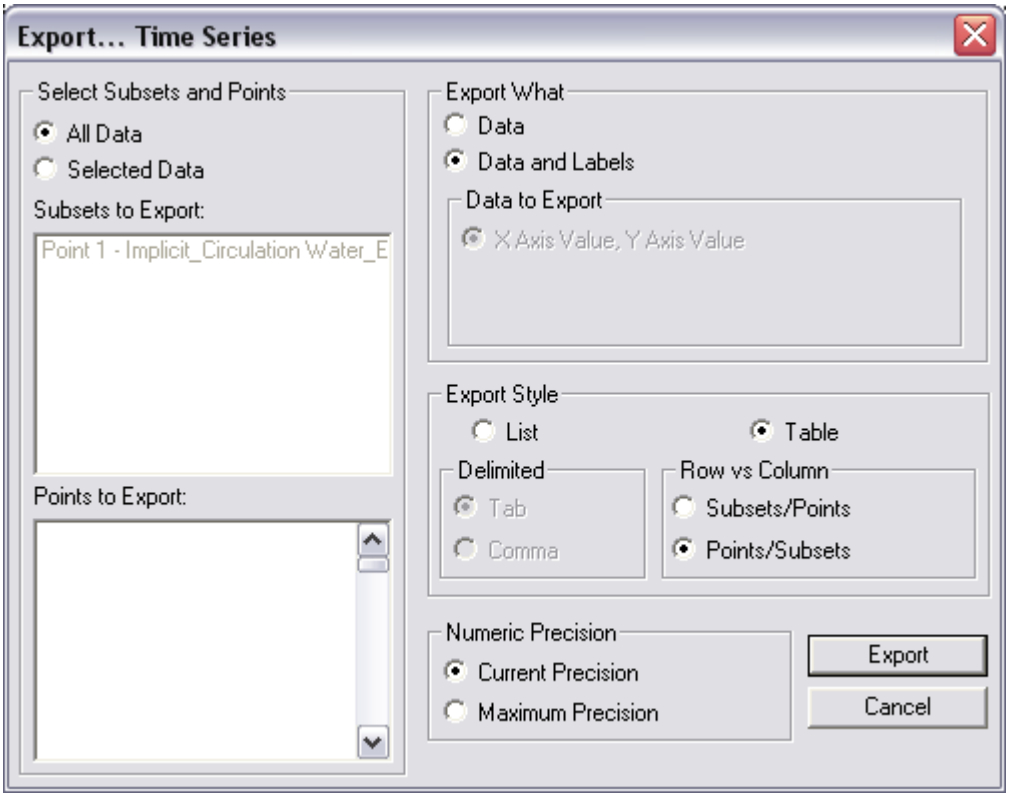


Figure 7-10. Export Wizard for plotted data.

5. To view the raw data (and another way to extract) right-click on the graph window and select *View Values*. The data are presented in a table that can be highlighted and copied to another program.

Other options for the display plot can be found on the XMS Wiki (http://www.xmswiki.com/xms/SMS:2D_Plots).

CMS-calculated water level variation is compared with water levels from the Belmar gauge (location shown in Figure 4-57) and given in Figure 7-11. Because the Sandy Hook gauge is located 30 km north of Shark River, the calculations have a slight phase advance in comparison to the measurements because the tidal wave propagates from north to south on this coast. The ocean gauge typically leads the bay gauge by 20-30 min.

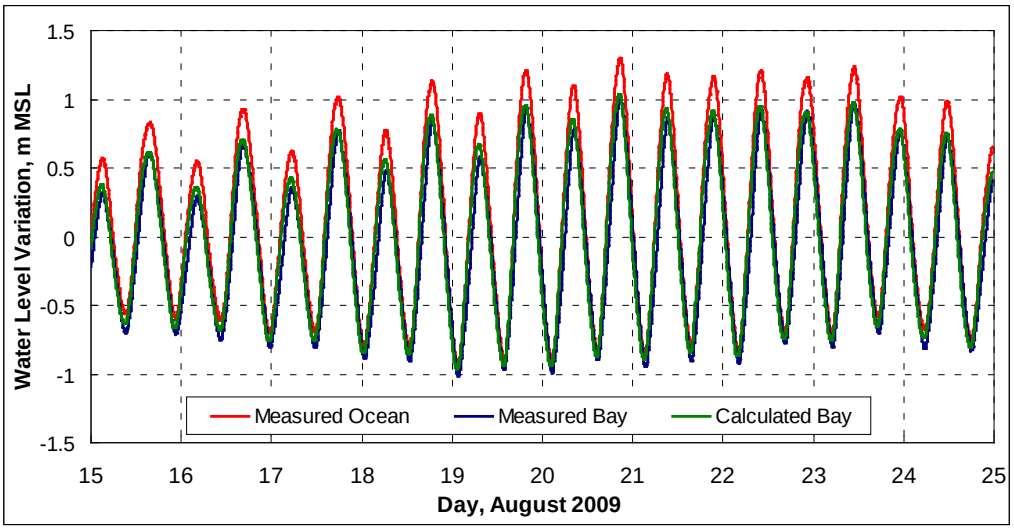


Figure 7-11. Observed time series of water level at Sandy Hook and Belmar ("Bay") and calculated water level at Belmar.

If time series data for the model were not set to the same interval as the measured data, there are filtering options within the data calculator. Reducing output (water level) from the solution:

3. To create a 30 minute dataset of water levels from the original 6 minute output, go to *Data*, *Data Calculator*, and under the *Tools* section, select *Temporal*, *Sample Time Steps* (Figure 4-62).

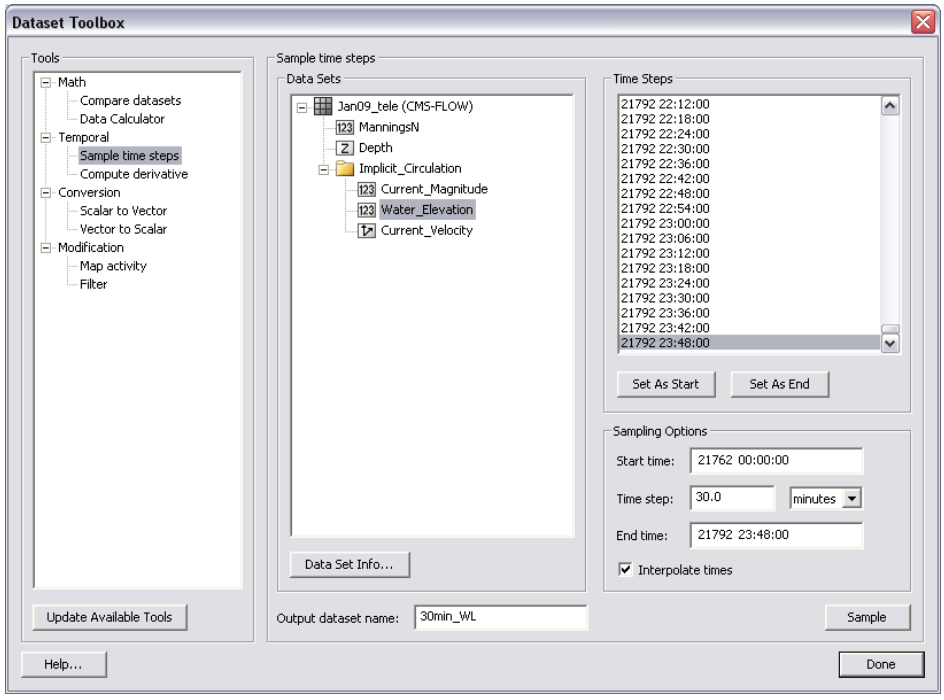


Figure 7-12. Sampling time steps of the Water_Elevation scalar dataset in the Dataset Toolbox.

4. Select the *water level* dataset under the solution datatree, and create an *Output dataset name* at the bottom.

5. Set the Start time to the beginning of the series and the end time to the end by selecting the time under *Time Steps* and selecting the associated button below. Change the time step to 30 minutes (or 0.5 hours), select *Sample*.

Depth-average down-looking ADCP current data were measured at three transects, or 2-D profiles along transects, in Shark River Inlet during the calibration time period in August, 2009. This data was processed with off-the-shelf software, where velocity was binned into depth measurements throughout the water column, and converted into a .GIS file. To display preprocessed measured velocities, they must be converted into an XYZ format with additional data (velocity and direction, or Vx and Vy) in additional columns. SMS will recognize both the XYZ data, and assign extra scalar or vector datasets to the imported files.

Importing GIS-type files from D-ADCP current measurements output into SMS:

1. Open the provided file in the section folder SRI082009_DAV_1300.GIS, and select *Use Import Wizard* (An excel version of the text file is also provided in the folder). Select *space/tab* delimited if not already checked and click *Next*, and select the X, Y, Z, Vector X and Vector Y columns (Figure 7-13). Be sure that the Header of the Vector X & Y columns is the same. Change the headers so they are the same (Vector), otherwise this incurs an error when importing in SMS.

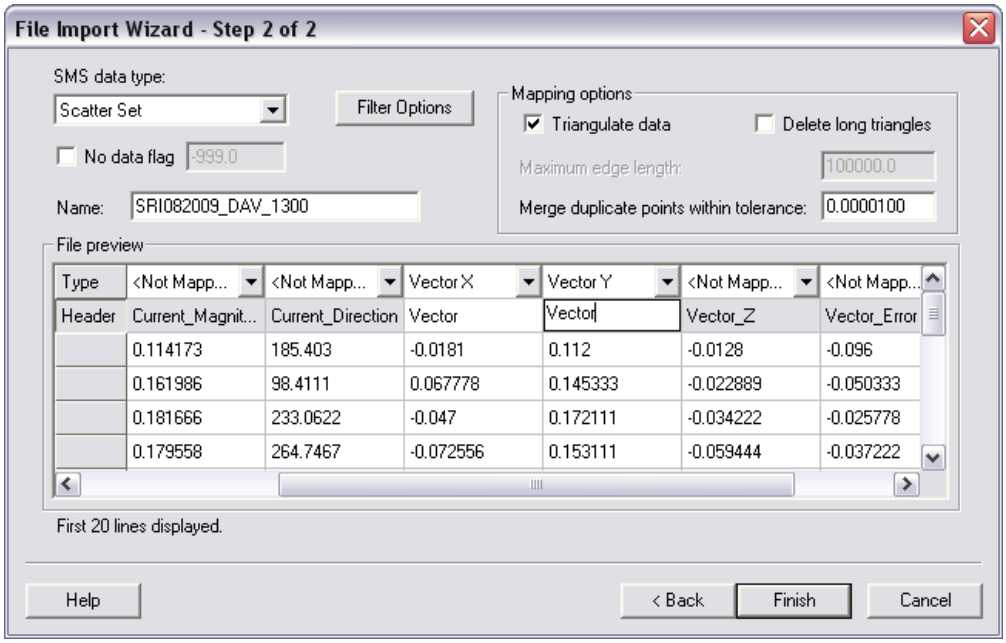


Figure 7-13. Sampling time steps of the Water_Elevation scalar dataset in the Dataset Toolbox.

All of the files were imported into SMS similarly as the time series data in the above section and are provided for you. All the data imported included X, Y, Z, Vx, Vy, Mag, and Dir for all three transects at a particular measurement time period. The files (under Hydro\Water Level Calibration\Measured D-ave velocity GIS\SMS) have been converted into scatter datasets for quick import in SMS for viewing.

2. Import (drag and drop) one or two of the .h5 files (e.g. *SHARK820_HOURLY - SRI082009_DAV_1100.h5*). In

Display Options, under *Scatter*, select velocity vectors and adjust the vector settings tab (Figure 7-14).

3. For a general estimate of comparison of the magnitude, convert one of the measurement files' vectors to magnitude and direction (right click, *vector to scalars*). This can be compared to the calculated magnitude for a rough estimate.
4. The scalar value of the measured data is in centimeters. Create a new dataset in meters using the data calculator (Figure 7-15).

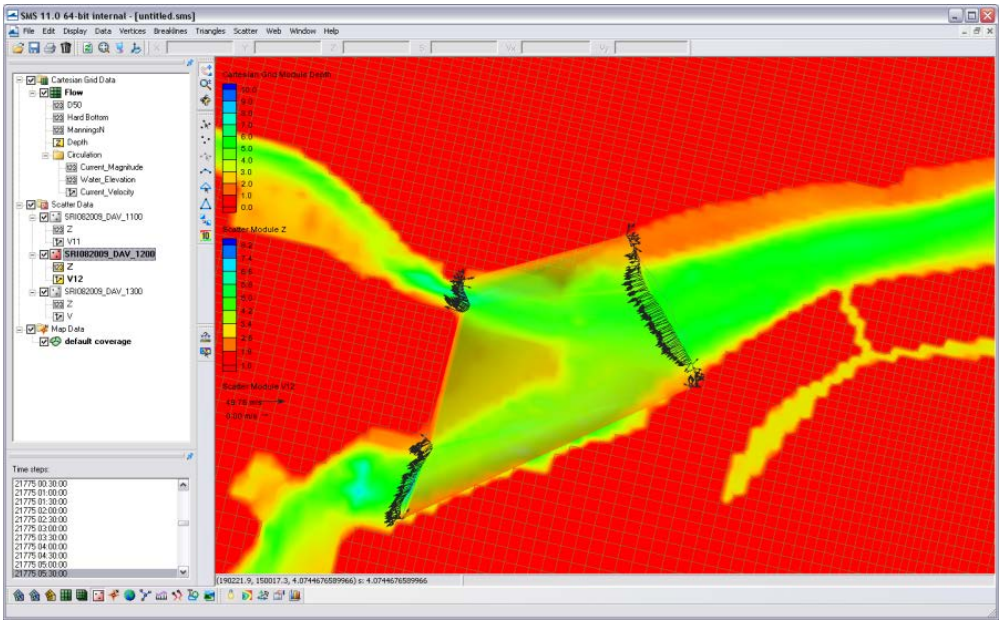


Figure 7-14. Example of the vector display along the measured transects. Note the vectors are scaled to the magnitude, and these scales can be manipulated in the Scatterset Vector tab.

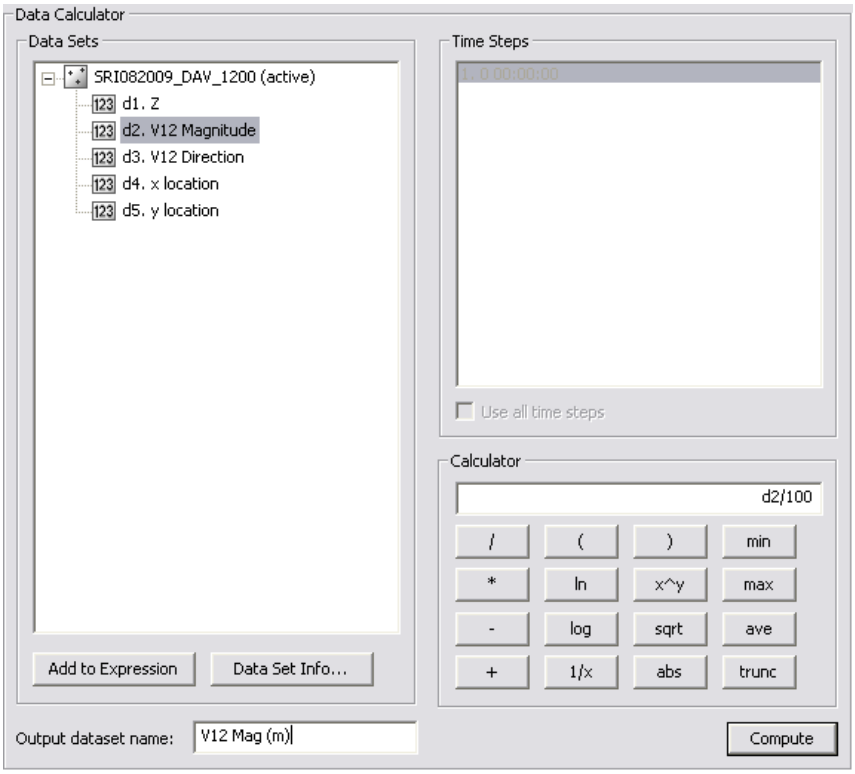


Figure 7-15. Data calculator used to convert the vector magnitude from centimeters to meters for comparison to the calculated velocity magnitude.

6. To display the scatter points with colors representing the flow, go to the *Scatter* Tab under the *Display Options*, and select the *Use Color Contour Scheme* under *Points*. Make the points a size 10 and they will be visible and the color differences noticeable as in the figure below.

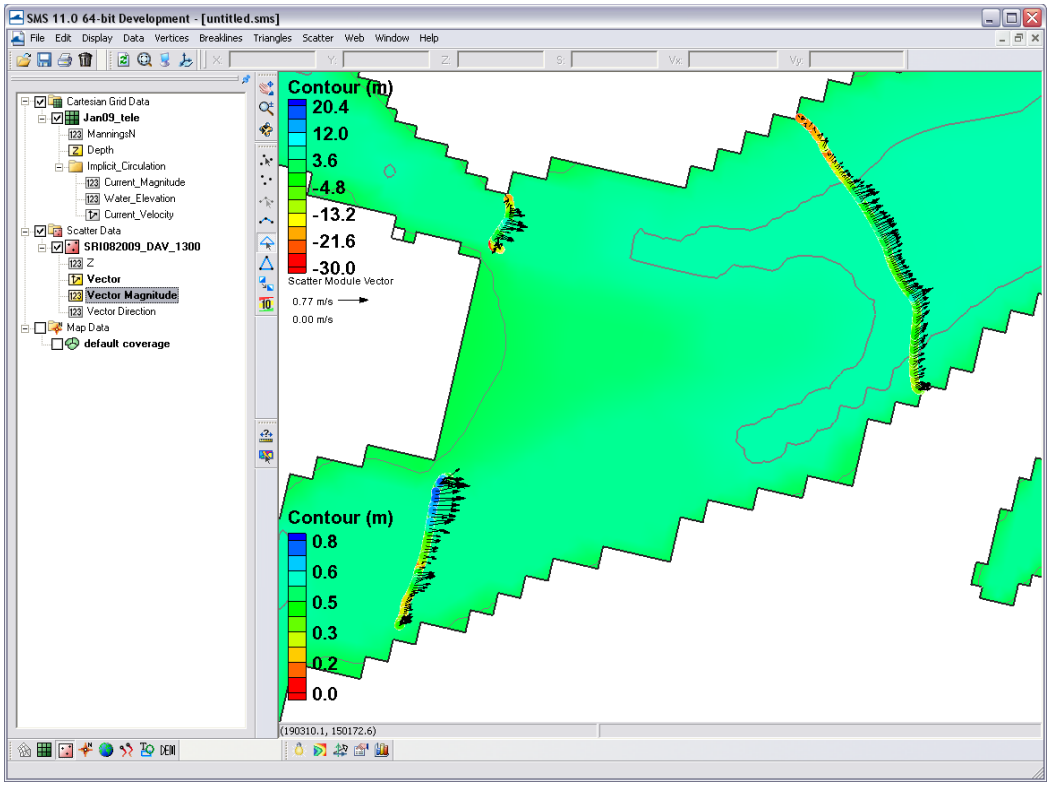


Figure 7-16. Color options turned on for scatter points.

To compare the measured current velocity data to calculated, the data must be in SMS in this format, where xyz data overlay the proper region and the units of the measurements are the same. Both datasets can be viewed and extracted within the SMS through the use of observation arcs. The arcs must be set along the transect line and have enough points to illicit one value per cell, but not too many where there will be duplicate calculated values. The imported measured datasets are much denser than the cell coverage. Below describes how to plot a transect, or observation arc, through the measured points that will display both measured point data and calculated data across a distance.

Displaying measured and calculated currents in SMS.

1. Following the above section, the current velocity file is displayed similar to Figure 7-17. Draw three observation arcs over each of the three transects. Directional arrows will display, and the direction of the arc should reflect ebbing or flooding. If the arrows face the flooding direction, flood currents will be positive, and ebb currents will be negative.
2. Redistribute the vertices on the arcs (they should not have any yet) to a value similar to the number of cells across each transect. E.g. Transect 1 (main channel) should have 25-30 vertices.

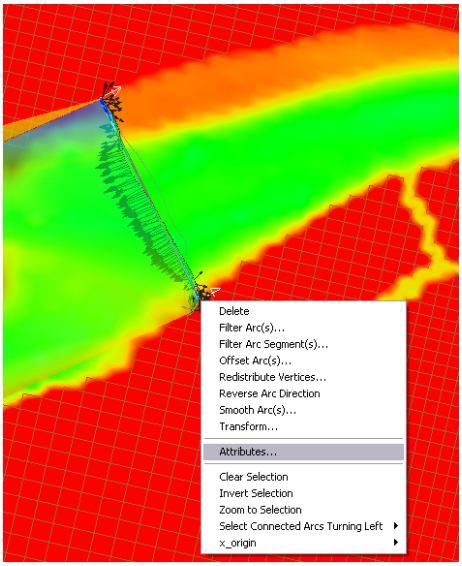


Figure 7-17. Example of a feature arc with arrows in the ebb direction. Redistribute Vertices, Reverse Arc, and other options are available in the drop down menu.

3. To plot, select *Plot Wizard* from *Display*, and select an *Observation Profile*. In Step 2, select one of the arcs (uncheck others), and under *Extra Profile* select *Model Intersections*. Figure 7-18 illustrates everything selected for plotting the measured and calculated data for the specific time of the transect measurement.

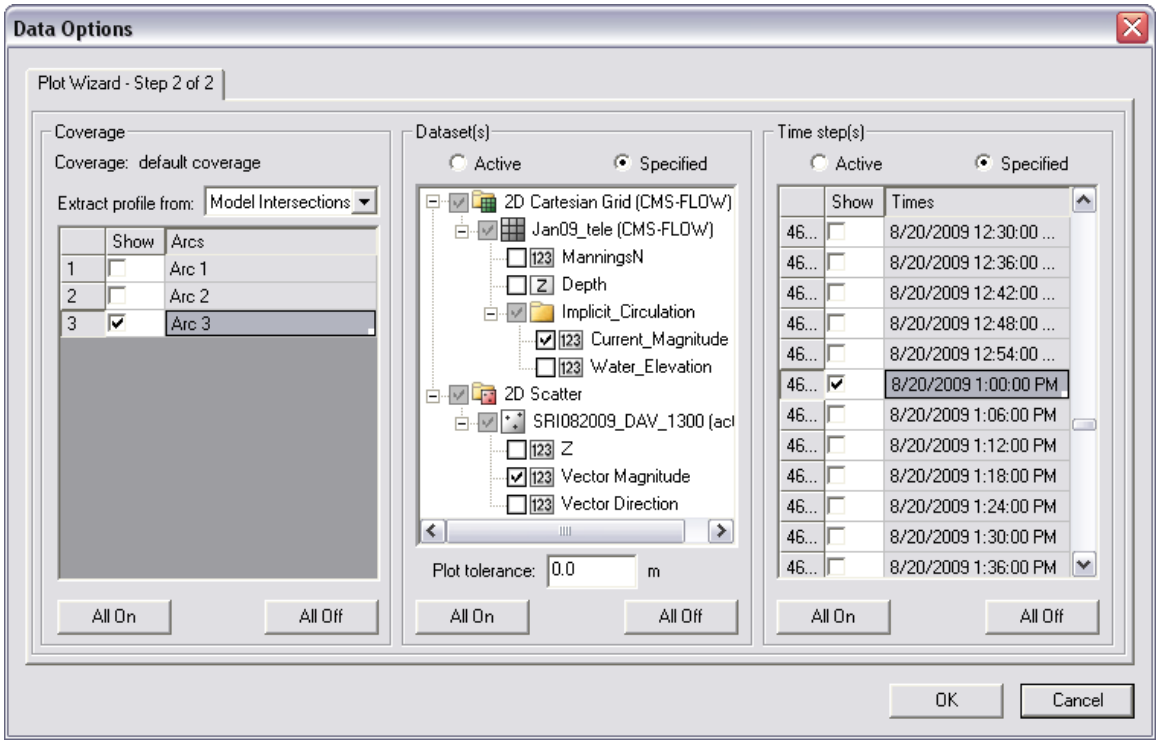


Figure 7-18. Plot Wizard data options for displaying measured and calculated data from an observation arc.

4. Specify the datasets to be plotted, the *Current_Magnitude* under the grid and the *Vector Magnitude* under the scatterset data. Select the date and time the transect was measured, which was 20 August 2009 at 1PM GMT.

5. The resultant plot is shown in Figure 7-19, where the closest points from the measured data were plotted as distance across the arc, and model grid cell centers are where numbers are extracted for the calculated data. Data can be extracted from the plot similarly to the ways defined in the water level extraction section at the beginning of this section.

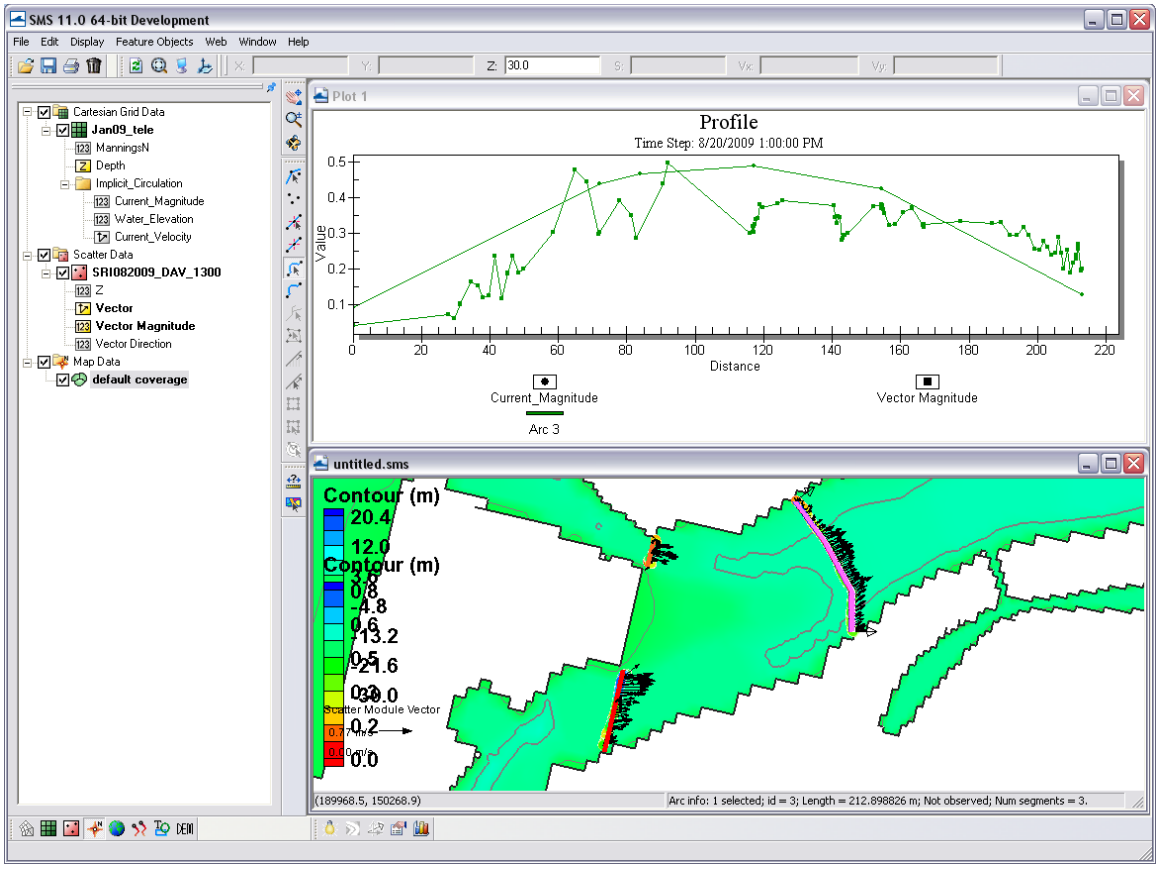


Figure 7-19. Plotted measured and calculated data from the larger observation arc to the East. Note that the positive direction reflects the observation arcs direction.

Sediment Transport

Sediment transport rates can be used to calculate sediment statistics, sediment pathways, fluxes across arcs, and balances over polygonal areas. Several sediment transport statistics can be calculated during the simulation and output at any time. These statistics include the net and gross total sediment transport rates. For additional details see the Simulation Statistics section.

Calculating Sediment Transport Roses at Observation (Save) Points

Sediment transport roses are plots in which transport is integrated over directional bins and plotted over a map of the site in order to observe the amount and direction of sediment transport at a specific point. A general picture of the sediment transport pathways can be obtained by plotting

several sediment transport roses in the areas of interest. Below is a step-by-step example for Shark River inlet on how to plot sediment transport roses:

1. Setup Observational Cells
2. Click on the Select Cells tool.
3. Select a cell and left-click
4. Select Cell-attributes... (see Figure below).

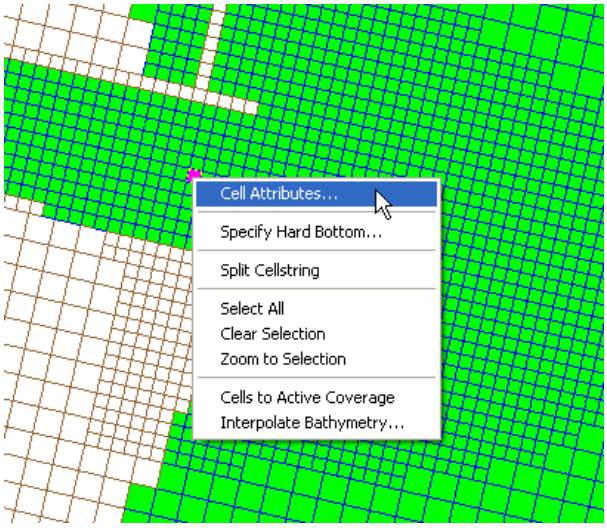


Figure 7-20. Opening the CMS-Flow Cell Attributes window.

The *CMS-Flow Cell Attributes* window will appear (see Figure below).

5. Check the box next to *Transport output*, so that the model will output the sediment transport rates.
6. Save the project

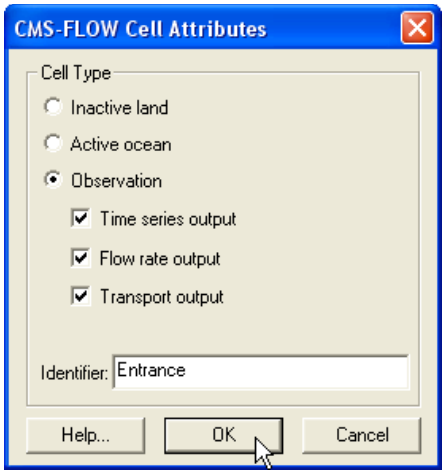


Figure 7-21. CMS-Flow Cell Attributes window.

Example observational cell cards:

```
!Observation Cells
TIME_SERIES_INCREMENT                900.0
FLOW_RATE_INCREMENT                  900.0
Q_TRANS_RATE_INCREMENT               900.0
ELEV_OBS_CELLS_BEGIN
     22943 "North Beach"
     28784 "Bay"
     30926 "Bridge"
ELEV_OBS_CELLS_END
FLOW_OBS_CELLS_BEGIN
     22943 "North Beach"
     28784 "Bay"
     30926 "Bridge"
FLOW_OBS_CELLS_END
Q_TRANS_OBS_CELLS_BEGIN
     22943 "North Beach"
     28784 "Bay"
     30926 "Bridge"
Q_TRANS_OBS_CELLS_END
```

7. Run the model
8. Once the model has completed. The project folder will contain two files with the extension *_qtx.txt, and *_qty.txt. These files contain the total sediment transport rates in the x and y directions in units of kg/m/s.
9. Plot the sediment transport roses in Matlab
10. Exporting a Georeferenced image in SMS

If the grid is telescoping, it may be difficult to plot the grid in Matlab. There are ways of plotting a telescoping grid in Matlab, but for the purposes of visualization, the easiest approach is to export a georeferenced image from SMS and plot the image in Matlab. This is done by

1. Click on *File* | *Save as…* (see Figure below)

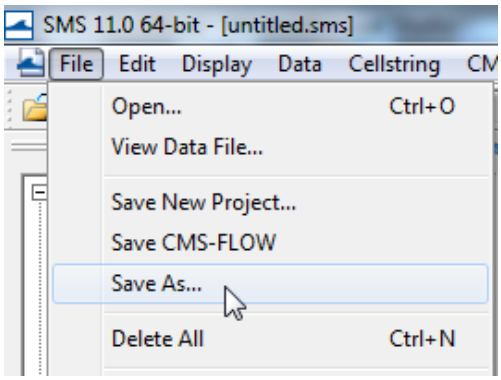


Figure 7-22. Opening the SMS Save As window.

2. Next to *Save as type*, select JPEG Image File.
3. Navigate to the correct directory
4. Enter a file name
5. Click *Save* (see figure below).

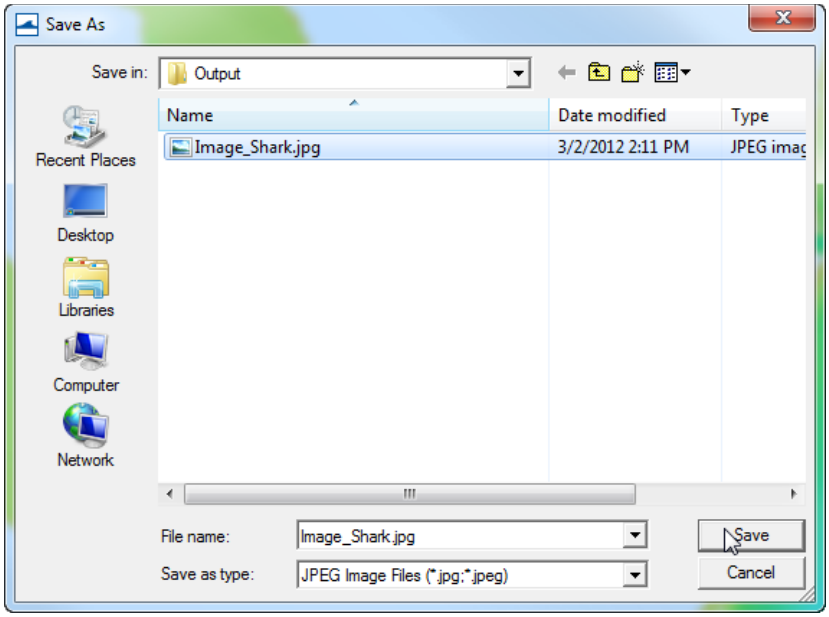


Figure 7-23. SMS Save As window.

An image file will be saved along with a projection and world file. The projection file contains the horizontal projection information and the world file contains the coordinates of the image origin, the rotation angles (zero for SMS), and pixel size. The image file can easily be loaded and plotted in SMS.

Editing and running the Matlab script

The Matlab script below is just an example to show the basic steps in plotting the sediment transport roses in Matlab. For each application, the file names, directories, station ID's, and settings will need to be modified.

```
%Plots Sediment Transport Roses
clear all
%1. Obtain Observation Point coordinates
id = [22943 28784 30926 33484 34145]; Station id's
tel = read_cmstel('..\Hands-on\Flow_Shark.tel');
xsta = tel.x(id); ysta = tel.y(id);
cosang = cos(tel.angle*pi/180);
sinang = sin(tel.angle*pi/180);
%2. Read transport vectors
qtx = load('..\Hands-on\Flow_Shark_qtx.txt');
qty = load('..\Hands-on\Flow_Shark_qty.txt');
%3. Read and plot background image
A = imread('..\Hands-on\Image_Shark2.jpg');
wld = load('..\Hands-on\Image_Shark2.wld');
xi = wld(5) + (0:size(A,2)-1)*wld(1); %SMS images are NOT rotated
yi = wld(6) + (0:size(A,1)-1)*wld(4);
```

```
imagesc(xi,yi,A)
set(gca,'ydir','normal','Nextplot','add')
%4. Plot roses, Rotate transports from local grid angle to world
for k=1:size(qtx,2)-1
    qte = qtx(:,k+1)*cosang - qty(:,k+1)*sinang; %East
    qtn = qtx(:,k+1)*sinang + qty(:,k+1)*cosang; %North
    [sumx,sumy] = sumdirbin(qte,qtn);
    x = ones(length(sumx),1)*xsta(k);
    y = ones(length(sumx),1)*ysta(k);
    h = vecplot(x,y,sumx,sumy,'marker','^',...
        'lengthscale',10,'linewidth',2,...
        'MarkerFaceColor','k','MarkerSize',3);
end
axis([1.9018    1.9257    1.4979    1.5132]*1e5)
xlabel('X, State Plane, m'), ylabel('Y, State Plane, m')
return
```

The subroutine `sumdirbin` sums the transport rates over directional bins. Note that to integrate over time the values should also be multiplied by the output interval. However, because the vectors are only for plotting purposes, this step is not necessary here. The matlab script for summing the transports in directional bins is provided below

```
function [binx,biny] = sumdirbin(u,v,binsize)
% [binx,biny] = sumdirbin(u,v,varargin)
% Sums the vector u,v in directional bins
% written by Alex Sanchez, USACE
if nargin<3
    binsize = 22.5; %deg, default bin size
end
u = u(:); v = v(:);
mag = sqrt(u.^2+v.^2);
binlim = (-binsize/2:binsize:360-binsize/2)'; %limits
bincen = (0:binsize:360-binsize)';            %centers
ndir = length(binlim);
angle = atan2(v,u)*180/pi;
angle = mod(angle,360)+0.00001;
angle(angle==0) = 0.00001;
binmag = zeros(ndir-1,1);
for k=1:ndir-1
    ind = (angle>binlim(k) & angle<=binlim(k+1));
    binmag(k) = sum(mag(ind));
end
binx = binmag.*cos(bincen.*pi/180);
biny = binmag.*sin(bincen.*pi/180);
return
```

The resulting Matlab plot is shown in the figure below.

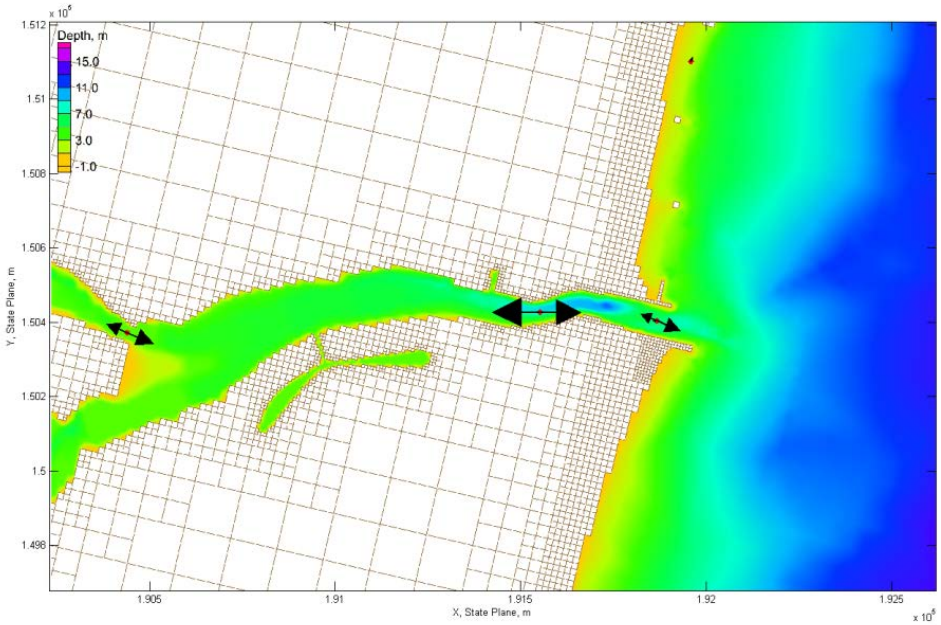


Figure 7-24. Total-load Sediment transport rates for Shark River Inlet.

Calculating Sediment Fluxes across Observational Arcs

In the CMS, sediment transport can be calculated across lines or polygons defined by feature arcs and polygons in a post-processing procedure. The integration across an arc may be written as

$$\bar{Q}_t = \oint_L \frac{1}{T} \int_t (\vec{q}_t \cdot \vec{n}) dt dL \qquad (7.1)$$

where $\vec{q}_t$ is the total-load sediment transport rate vector, t is time, L is the arc length, and $\vec{n}$ is the unit vector normal to the arc line. Note that the order of the above integrals is irrelevant; however it is more efficient and accurate to perform the temporal integration first. CMS has the option to output sediment transport statistics which includes the net total-load sediment transport rate which is basically the temporal integration shown in the equation above. For further details on the sediment statistics see section Simulation Statistics. To use the sediment statistics in calculating fluxes across observational arcs:

1. Set the Sediment Transport Simulation Statistics (see section Simulation Statistics for more details).
2. Run the CMS with sediment transport.
3. Load the XMDF Global Solution File into SMS (drag-and-drop is the easiest way).

The data tree will have several sediment transport statistics including the Gross and Net Total-load sediment transport rates in units kg/m/s. Figure 7-25 shows an example of the data tree with the sediment simulation statistics for Shark River inlet.

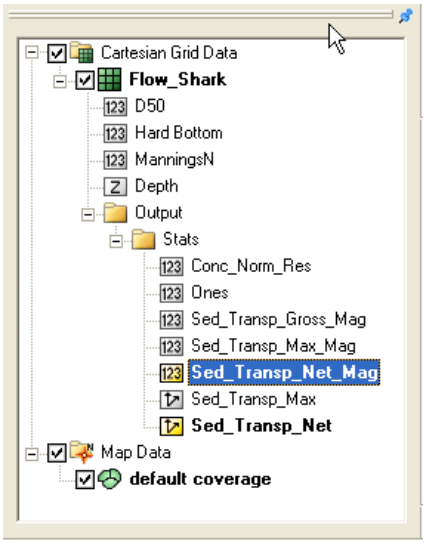


Figure 7-25. Data tree showing the sediment and hydrodynamic simulation statistics.

The total-load sediment transport rate is output into the Sed_Transp vector and Sed_Transp_Net scalar datasets. Figure XX shows an example of the Net Total-load Sediment Transport vectors and magnitude after a short 24-hr simulation.

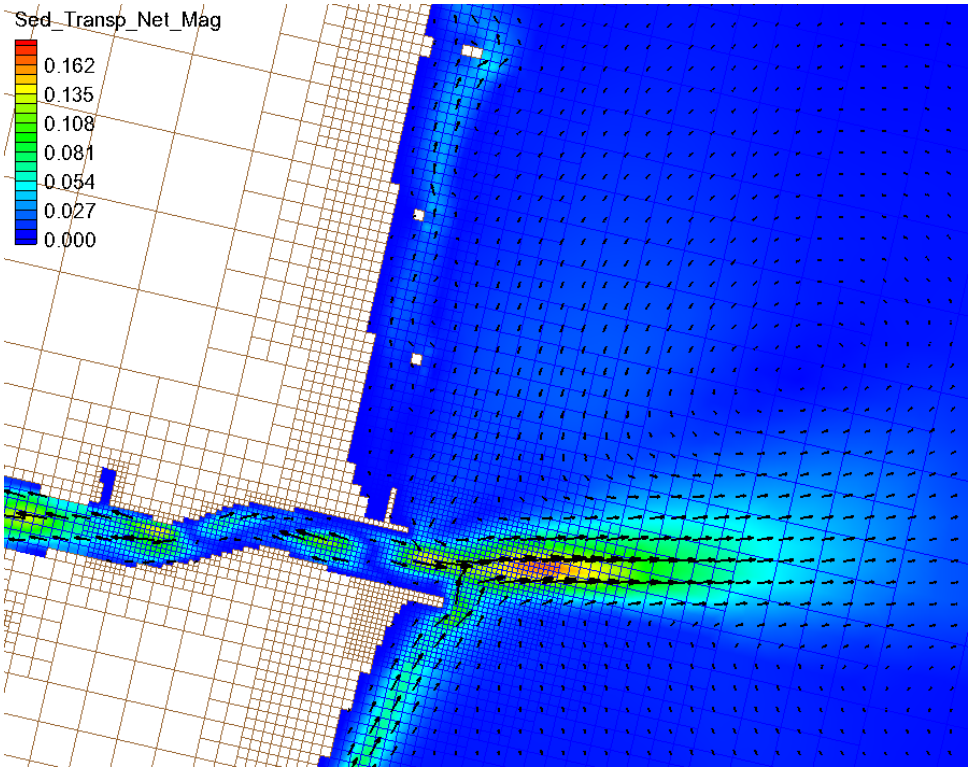


Figure 7-26. Example of the Net Total-load Sediment Transport Rate vectors and magnitudes for Shark River inlet after a short 24-hr simulation.

The datasets are output at start and end of the statistics period. The statistics at the initial time are always zero and are only necessary for SMS to be able to perform the line integral which is part of the Time Series plot (described further below).

1. Either:
    a. Create a new map coverage of Observation type or
    b. Set the current map coverage to Observation type.

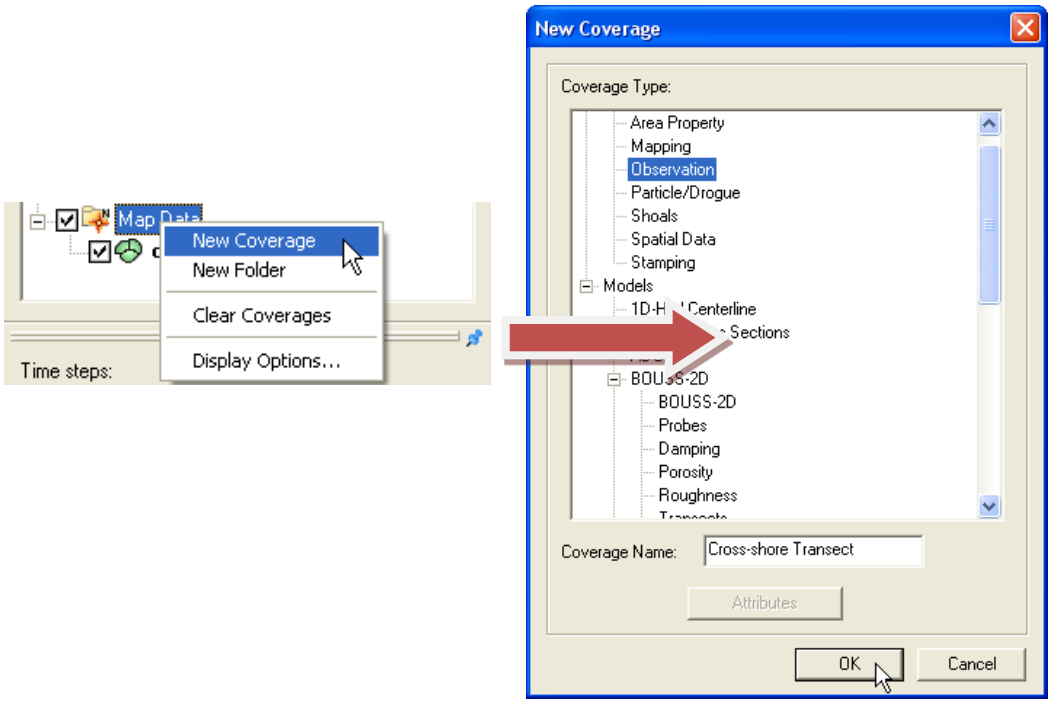


Figure 7-27. Creating a new Map Coverage of type Obesrvation.

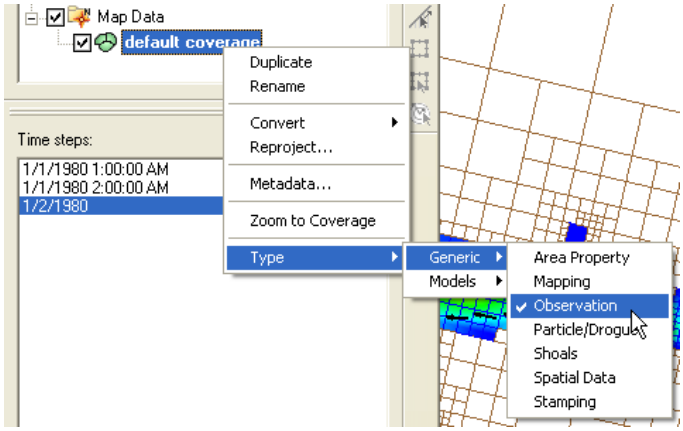


Figure 7-28. Setting the current Map Coverage to an Observational type.

2. Create an observational arc by selecting the Create Feature Arc Tool , and click on the points that define the Observational arc (see Figure below). The example shown below is intended for longshore sediment transport. When calculating longshore sediment transport, the offshore end of the transects should extend beyond the breaker and closure depth. It is best that the ends of the obser-

vational arc do not touch inactive (land) cells as this may cause interpolation problems in SMS.

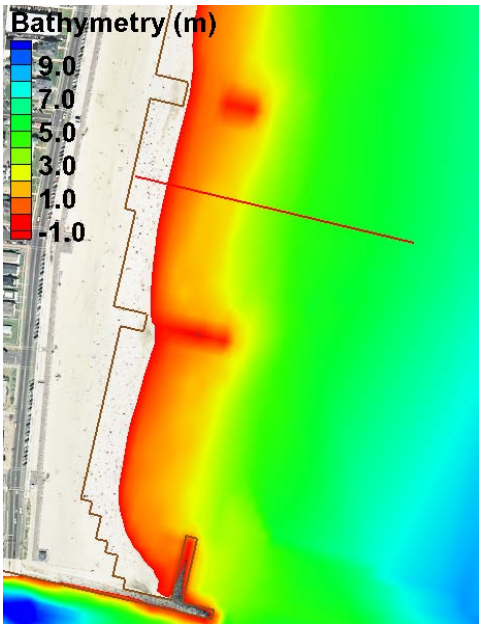


Figure 7-29. Example of a Feature Arc for calculating sediment transport.

3. If necessary, use the Select Feature Point tool to adjust the end points of the arc (the arc can be deleted by selecting with the Select Feature Arc tool and pressing the delete key).
4. The positive direction of the sediment transport is defined by the Feature Arcs direction. To view the arc direction, click on the Select Feature Arc tool and right-click on the Feature Arc once. The arrows displayed at the beginning and end of the transect indicate the direction of the arc which defines the positive for all fluxes or vectors calculated across it.

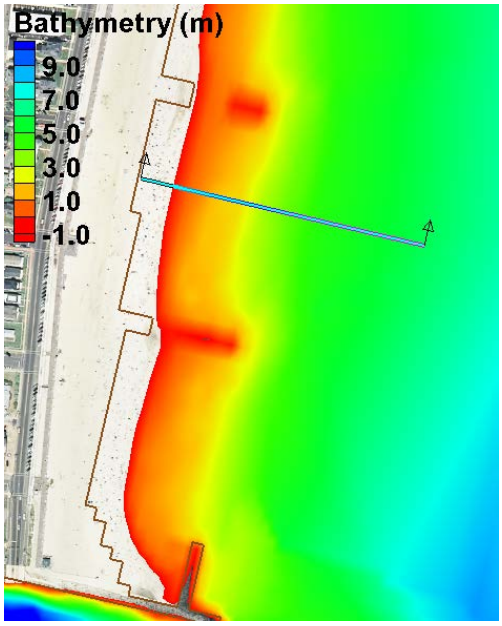


Figure 7-30. Highlighted Feature Arc with the positive direction shown by the arrows.

5. If necessary, the arc direction may be changed by clicking on the Select Feature Arc tool, and selecting the feature arc by right-clicking it once and selecting the option: Reverse Arc Direction (see figure below). The same applies for feature polygons.

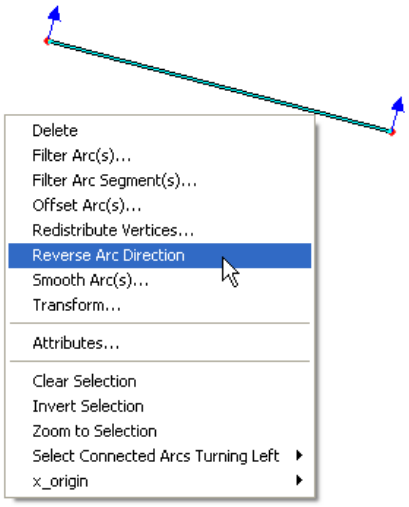


Figure 7-31. Reversing the arc direction.

The net sediment transport dataset is integrated along the arc (line integral) to compute the sediment transport across the Feature Arc. This procedure is done in the SMS Plot Wizard. The sediment transport vector field is interpolated to the points where the arc intercepts cell faces and is then integrated across the Feature Arc. Follow each step outlined below:

6. Open the *Plot Wizard* by clicking on the plot wizard icon , or clicking on the *Display* menu and selecting Plot Wizard.
7. Select *Time series* as the *Plot Type* and then click *Next*.
8. In the *Plot Wizard – Step 2 of 2* window, select *Flux* as the *Function type*.
9. Check the appropriate feature arc checkbox(es).
10. Select the ones dataset for the scalar dataset.
11. Select the *Sed_Tranp_Net* as the vector dataset.
12. Make sure the start and end times are correct.
13. Select *Finish*.

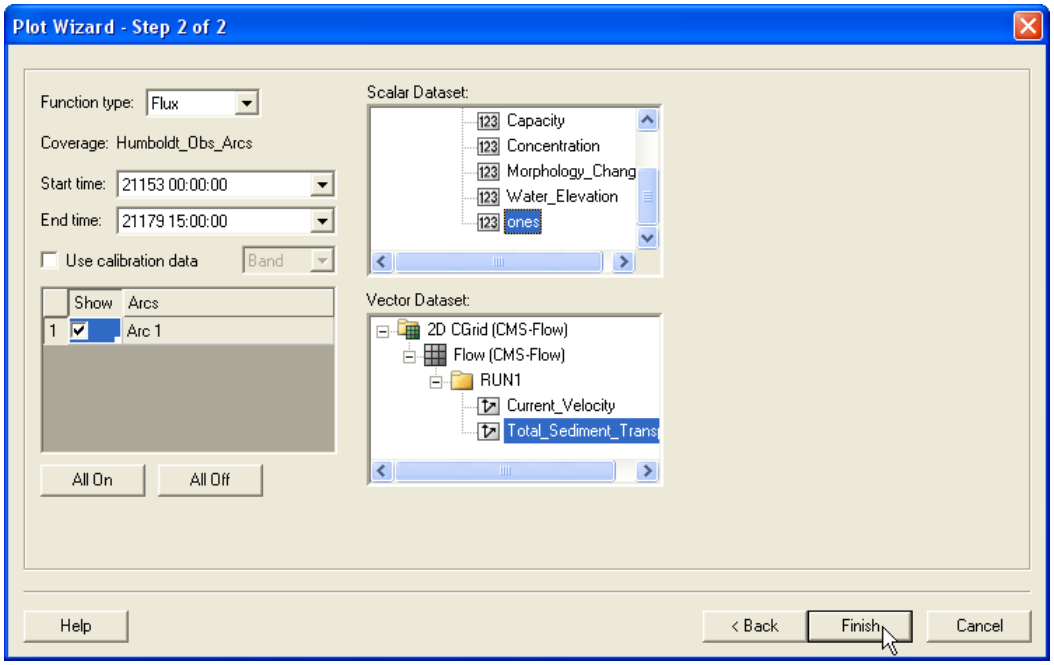


Figure 7-32. Selecting the correct Start and End Times, Feature Arcs, and Scalar and Vector Datasets in Step 2 of 2 of the Plot Wizard.

A plot will appear in SMS similar to the example shown below

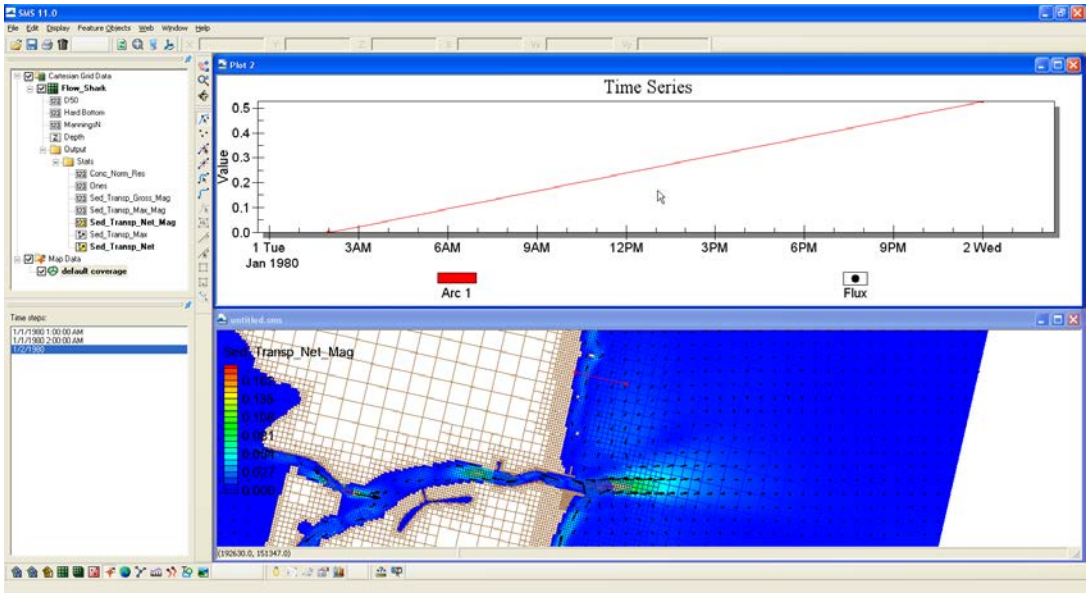


Figure 7-33. Example plot of an integrated net sediment transport rate across a feature arc.

To view the final value of the net sediment transport rate across the feature arc, left-click on the plot, and select View Values... (see figure below).

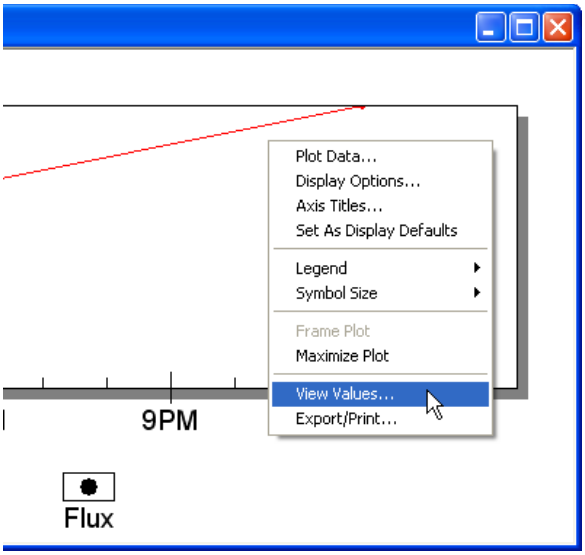


Figure 7-34. Selecting View Values... after left-clicking on the time series plot.

An SMS window will appear with the title View Values. The net sediment transport rate across the feature will be shown under the second row under the Value column. The units of the transport will be kg/s.

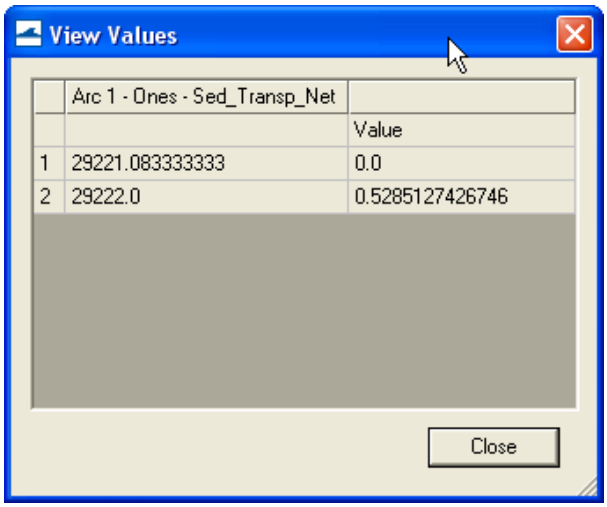


Figure 7-35. Net sediment transport rate integrated across

Morphology

The results of the CMS calculated morphology change can be visualized several ways, from three dimensional or planar view of the bathymetry, planar view of the volumetric erosion and accretion, and with 1-D cross-sections of the time series. All vector and scalar information can be extracted from points and arcs (and polygons) in SMS. This section will cover the methodology used to post process time series morphology change data.

To plot channel infilling in SMS in a graphical format, the results in the solution file need to be changed from depths (the depth below the datum in which CMS calculates) to elevations. SMS can plot multiple cross-sections, or arcs, against each other, or through time.

Channel Infilling by cross-sections

1. Load *Jan09tel.cmcards* (and solution file, if it does not load automatically).

2. Right-click on the *Time Series* window on the left and select "Time Settings", and change the time reference from *Relative* to *Absolute*.

3. In the Cartesian Grid Module, click *Data*, *Data Calculator*, and the Data Calculator should be selected in the Tools section as shown in Figure 7-36.

4. Select the *Bathymetry* scalar ( ) dataset; under *Time Steps* check the *Use all time steps* on, and double-click the *Bathymetry*.

5. The line under Calculator will display "e:all", add the multiplication symbol (*) and negative 1 (-1) as shown in Figure 7-18.

6. Change the *Output data set name* as shown in Figure 7-18 so as to distinguish this dataset from the original *Bathymetry* output, click *Compute*, and after the scalar set appears in the SMS window, click *Done*. (Note: values for the contours may need to be adjusted to view elevation range.)

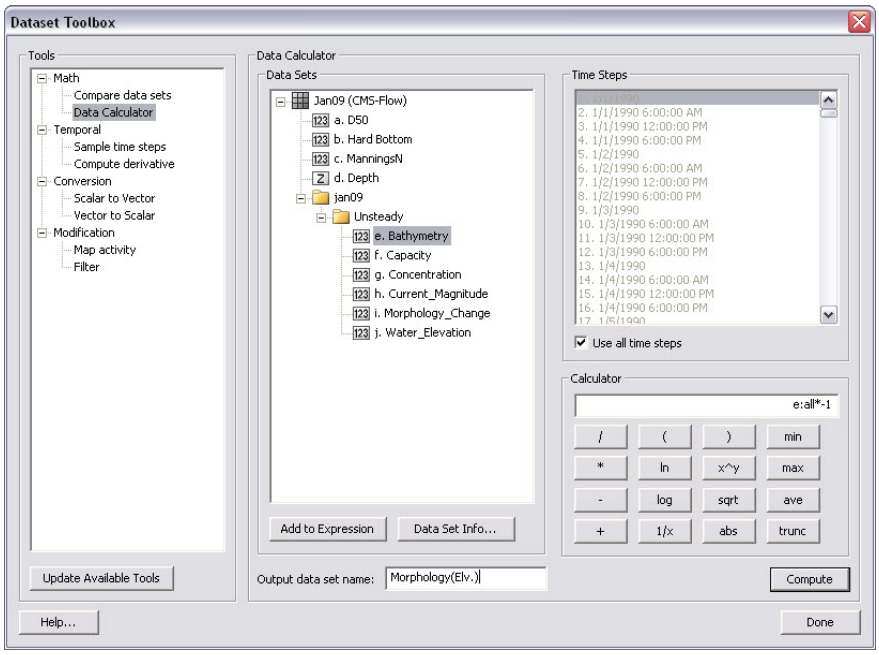


Figure 7-36. Data Calculator Options in the Dataset Toolbox.

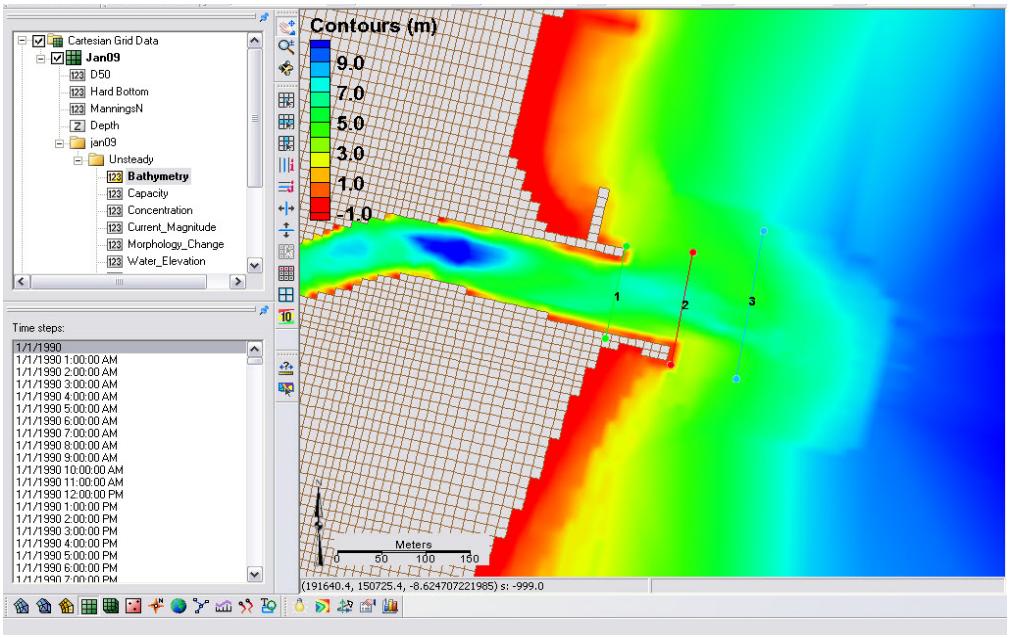


Figure 7-37. Display of calculated bathymetry now converted to elevations. Note the legend has been modified to display the new contours.

7. Create feature arcs for extracting the time series data by switching to the *Map Module* button, or Map Data

, rt-click on *default coverage*, and select *Type, Generic, Observation*

8. Click the *Create a Feature Arc* button to generate the arcs from which cross sections will be extracted, and click on one side of the channel and double-click on the other side to close the arc

9. Click the *Select a Feature Arc* button and select the generated arc (Note that there is a direction associated with the arc, which determines the sideview of the cross section)

10. Right-click an arc and click on *Attributes*, and set the color and name(s) of each arc which are listed together in the *Observation Coverage* options (Figure 7-38), click *OK*.

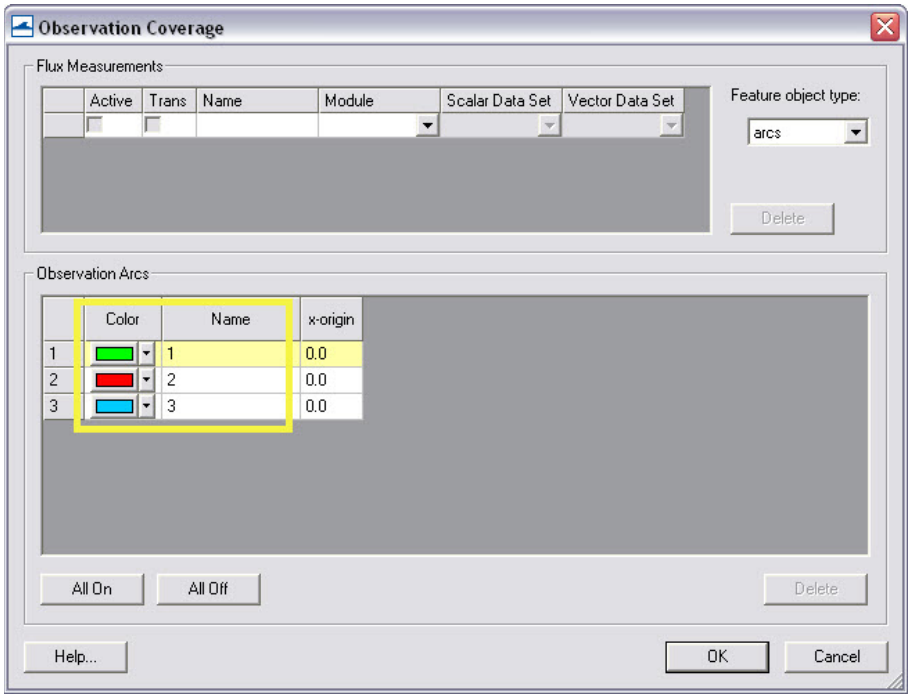


Figure 7-38. Observation Coverage attributes

11. To plot the profiles in SMS, Select *Display*, *Plot Wizard*, and *Observation Profile*, and select *Next*.

12. Following Figure 7-39, under *Coverage* check one arc (though multiple can be plotted), under *Data set* select *Specified data set(s)* and select the generated *Morphology* (Elev.) set only, and under *Time step* select *Specified time step(s)* and check on several times spaced apart by at least several days to months, click *Finish*.

13. Right-click on the plot (Figure 7-40) and select *view values..*, and a table will appear with all the plotted distance and elevations given in the graph, (Note this ASCII table can be copied and pasted into excel).

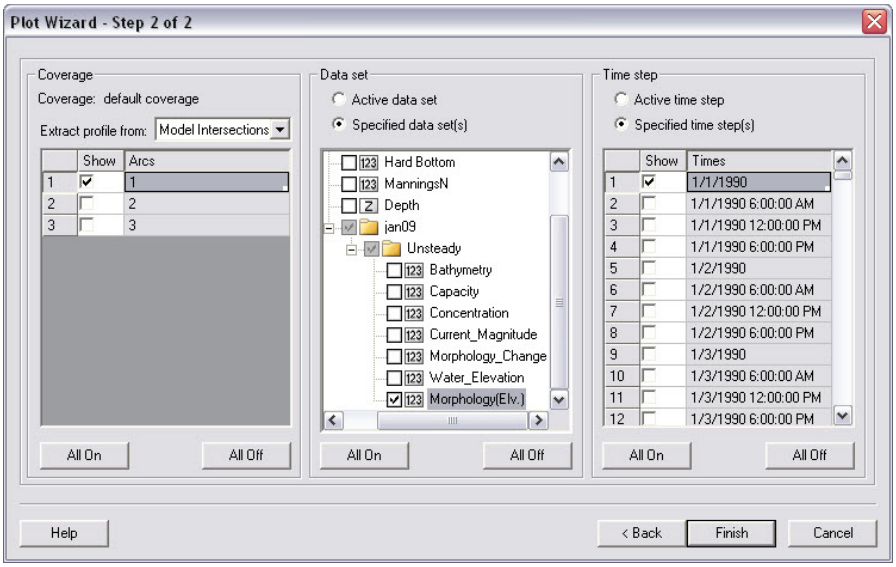


Figure 7-39. *Plot Wizard* Step 2.

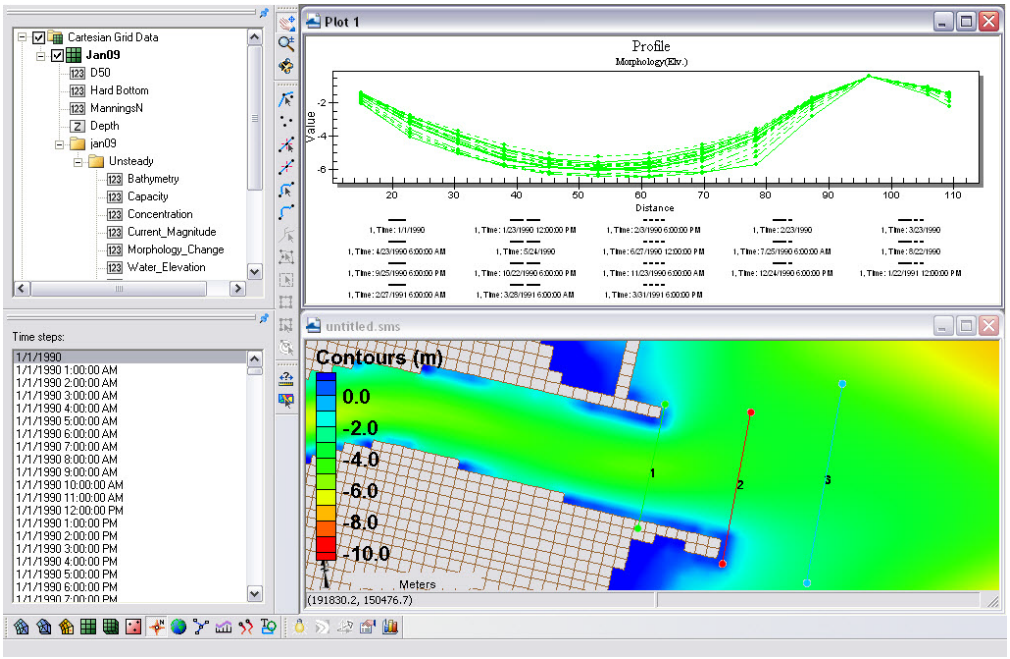


Figure 7-40. Plotted cross-section.

14. To export the data in different formats, right-click on the plot, and select *Export/Print* for several options to export the graph or data from the graph (Figure 7-41)

15. To add more profiles or time-series, right-click on the plot, and select *Plot data.*. to bring up the original options from Figure 7-40.

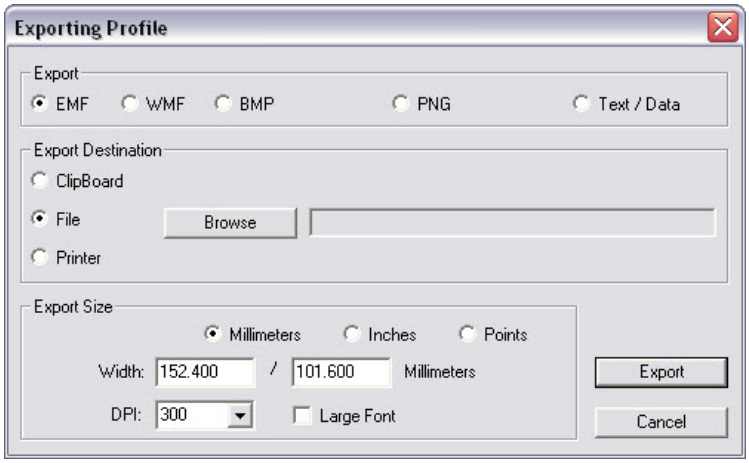


Figure 7-41. Data Export Options.

Analysis of cross-sectional area and sediment volume changes are the most common approaches to quantifying morphologic change. Another method of quantifying change in a spatial framework is through comparison of the planform aerial change. This method is useful in delineating active zones of transport, typically based off of the depth of closure, and can define the upper and lower limits of transport. These results can be used to interpret bypassing areas or zones, however, determining the bypassing pathways should not be based off of this analysis alone.

Planform area changes

1. To filter the morphology dataset for the time periods of interest, i.e. every month, open the *Data Calculator* and select *Sample time steps* under the *Temporal* section (Figure 7-42).

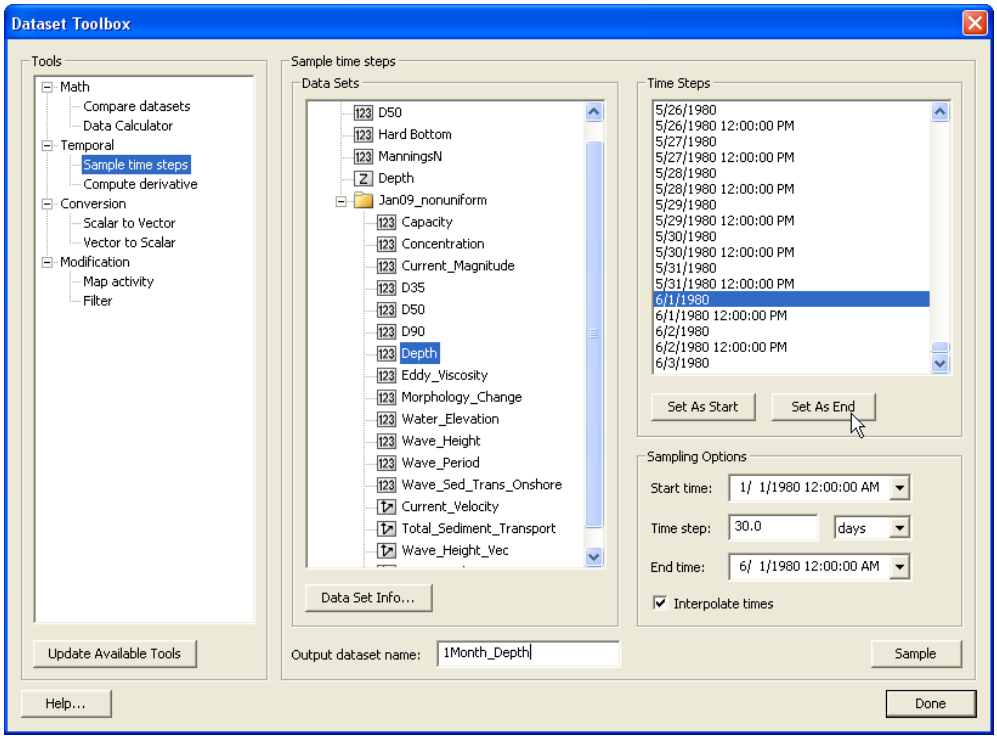


Figure 7-42. Sampling time steps from a solution dataset.

2. Select *Depth* under *Data Sets*, change the *Output dataset name*, and set the *start time* and *end time* to Jan 1 and Jun 1, respectively, with an interval of 30 days. Select *Sample*. This should produce a *Depth* time step approximately every month.

3. In order to map an elevation, the grid must be converted to a scatterset.

    - Rather than take the time required to convert the entire solution during this workshop, we have provided this file in the folder (File: *1Month_Depth*). (To convert the grid solution file to a scatterset, select *Data|Map to Scatter*, and SMS will convert all of your time series datasets to the scatter module (Figure 7-43). This can take anywhere from minutes to hours, depending on the size of the file and speed of the processor. Open the provided scatterset.

    - Or, delete the Jan09_nonuniform solution datatree so that the main CMS-Flow datasets and the *1Month_Depth* dataset are the only datasets left. Then select *Data|Grid-> Scatter*, and SMS will convert only those datasets.

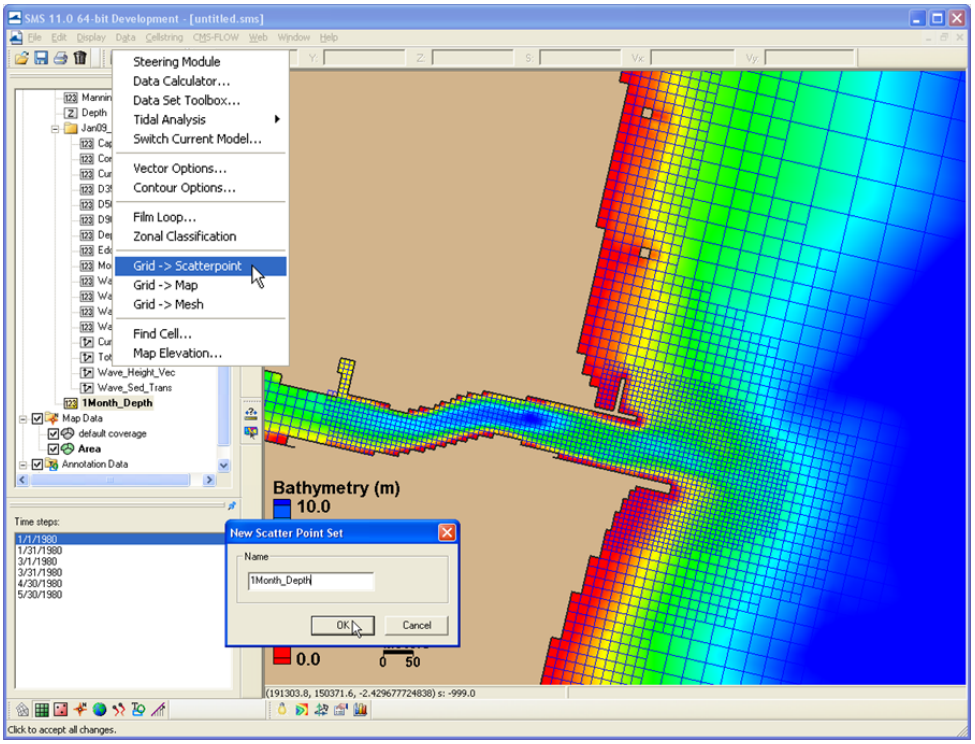


Figure 7-43. Converting a solution dataset to a scatter dataset.

4. Create a new map coverage (by right clicking *Map Data* in the data tree and selecting *New Coverage*) for mapping out specified contours over the time series depth scatterset.

5. Select the *1Month_Depth* Scatterset (*Scatter Module* should be on) and the first time stamp (1 Jan), and go to *Data|Scatter Contour -> Feature*. Choose an elevation to map (6.0 m), a spacing of the points in the resultant arc, and select the Map Coverage ('*Area*' in Figure 7-44) that the arcs are created in.

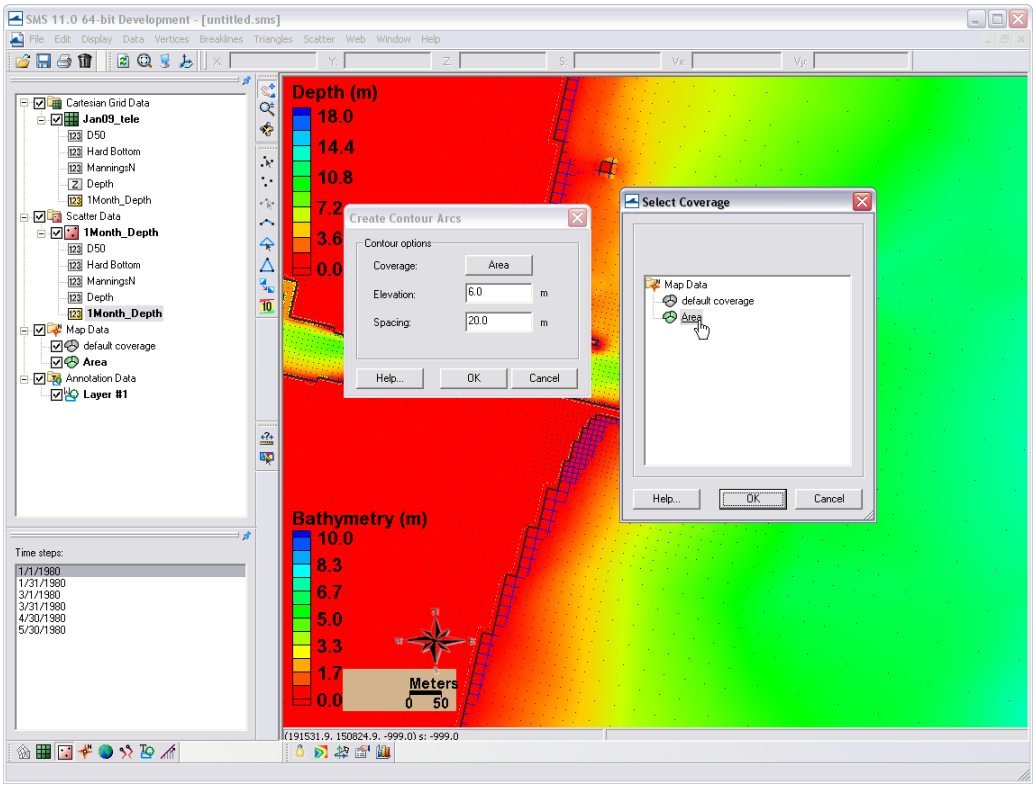


Figure 7-44. Mapping a scatterset contour to the map module.

6. Create another new Map Coverage, select the final timestamp (30 May), and go to *Data|Scatter Contour -> Feature* and select the same elevation (6.0 m) and point spacing. Switch between the two coverages to view the location of each contour (Figure 7-45). Note that the arcs hold a z-elevation (displayed in Z in the bar above) and are partially visible under the color-filled scatter data.

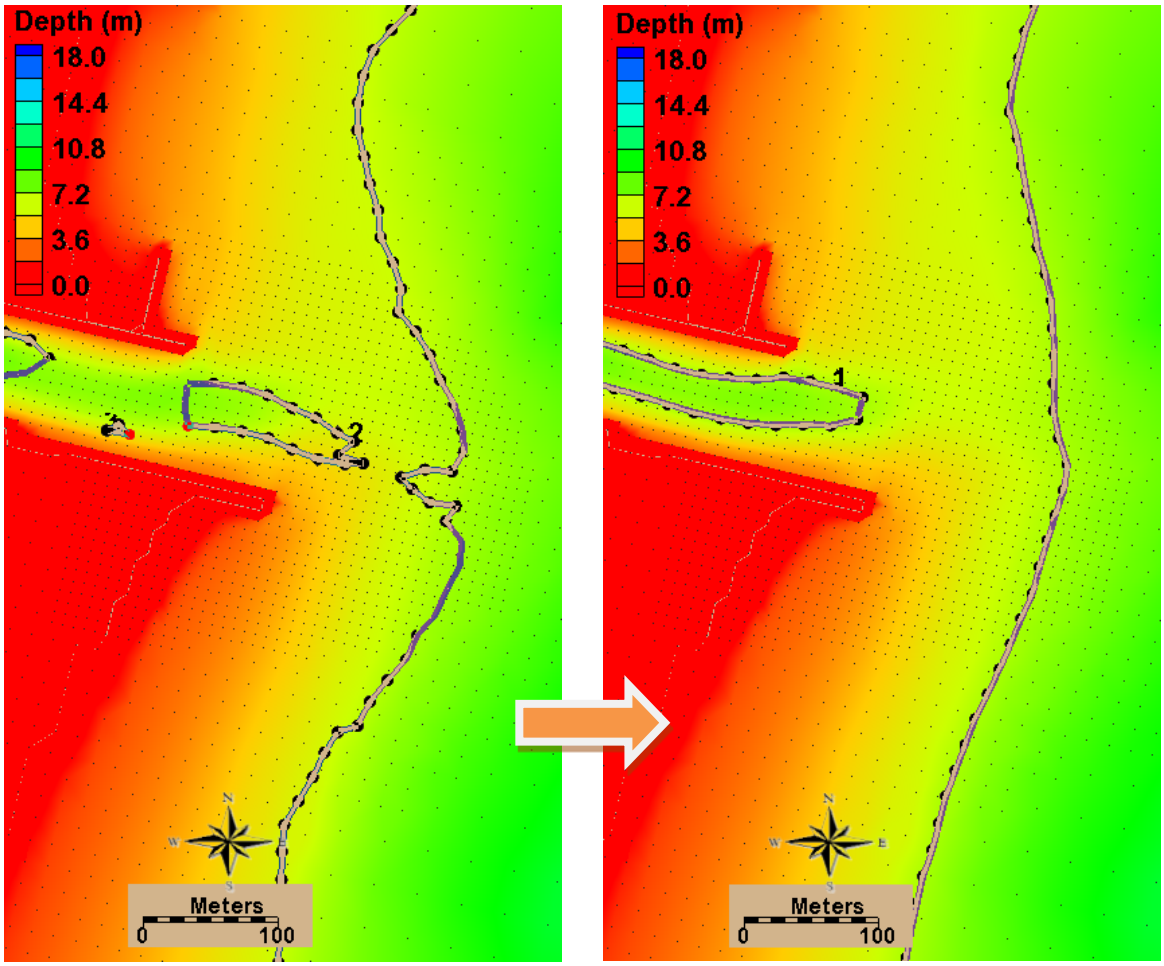


Figure 7-45. Mapped coverage of the 6-m contour at Months 1 and 6 (left and right, respectively).

7. To calculate planform area, each contour in both maps must be closed off with arcs to create a polygon. Create a polygon in the closed arcs by selecting *Feature Objects*|*Build Polygons*. An example is shown in Figure 7-46. Note that the area of the selected polygon is given at the bottom of the screen in the units the solution file is given in, which is always metric for the CMS.

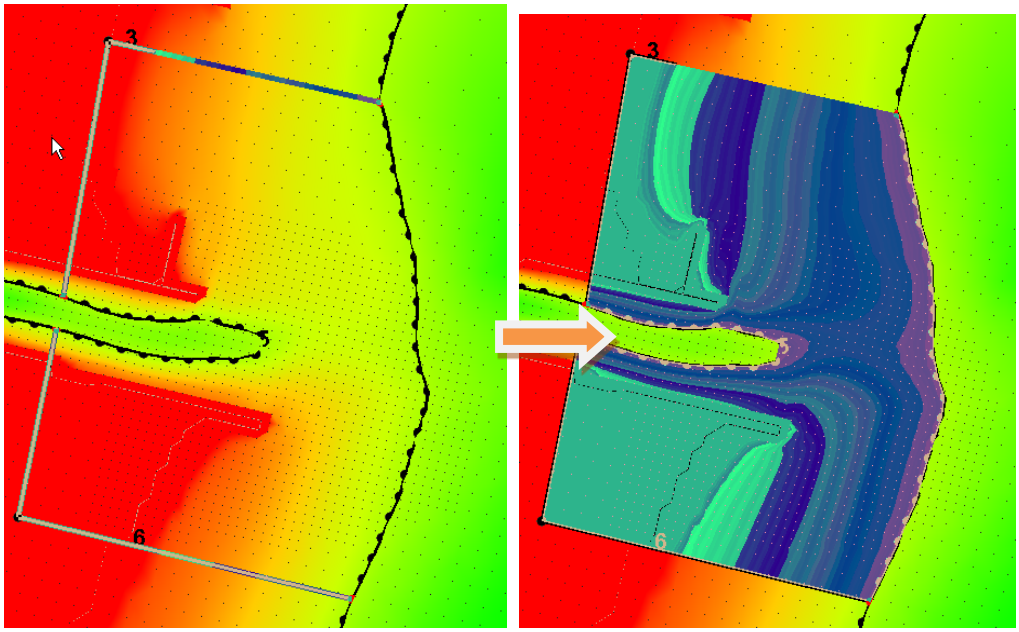


Figure 7-46. Left: Two arcs drawn to connect the 6m contour back to a common point to close off the polygon. Right: Highlighted polygon after created.

Creating polygons between time series allows for calculation of planform area change at a morphologic spatial scale through time. The difference shown in the arcs is substantial, and can be quantified with the use of arcs and polygons. It is important to be careful with delineating the morphologic form of interest. If the full area is being calculated for a feature (an ebb shoal), be sure to be discrete in the size of these aerial calculations because they may skew the final calculation of the morphologic feature of interest.

Similarly, volume change can be elicited from polygons created with Feature Arcs. The next section describes how to calculate volume change over an arc.

Calculating volume change from polygons.

1. Create a polygon similarly to the above exercise that created arcs to complete a polygon in the planform area calculations. Choose an area around the ebb shoal, or within the channel, and draw a single arc connecting around the feature. Be sure to close the end of the polygon by letting the mouse automatically hover over the final point (the pointer will relocate when placed near another feature point).
2. Create a polygon in the closed arcs by selecting *Feature Objects|Build Polygons*. An example is shown in Figure 7-46.
3. To select the cells within the polygon, next, with the polygon selected, go to *Feature Objects|Select/Delete Data*, and select *select*, *Cartisian Grid Cells*, and *Inside Polygon* (Figure 7-47).

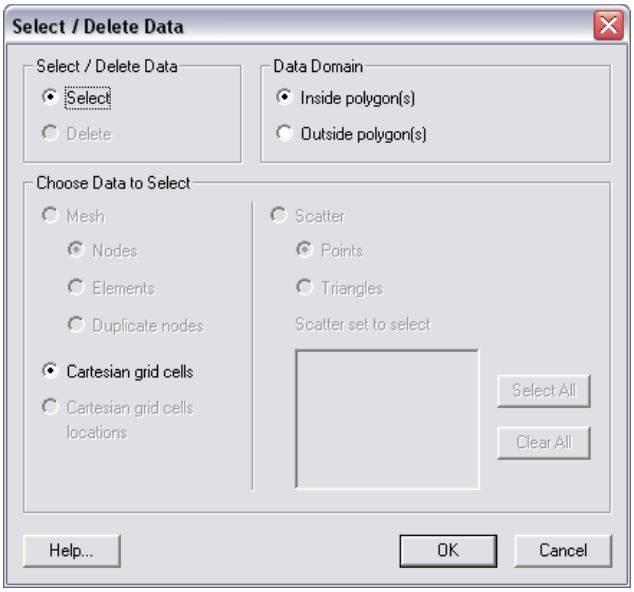


Figure 7-47. Options for selecting and deleting data from Mesh, Cartesian Grids, and Scattersets using polygons.

Note, again, that the volume of the selected polygon is given at the bottom of the screen in the units the solution file is given in, which is always metric (m^3) for the CMS. The volume given in the lower right corner of the window (Figure 7-48) are a positive, negative, and total volume representative of the integrated positive, negative, and combined volumes from each cell. Therefore, the numbers represent the volume based off of the dataset's datum.

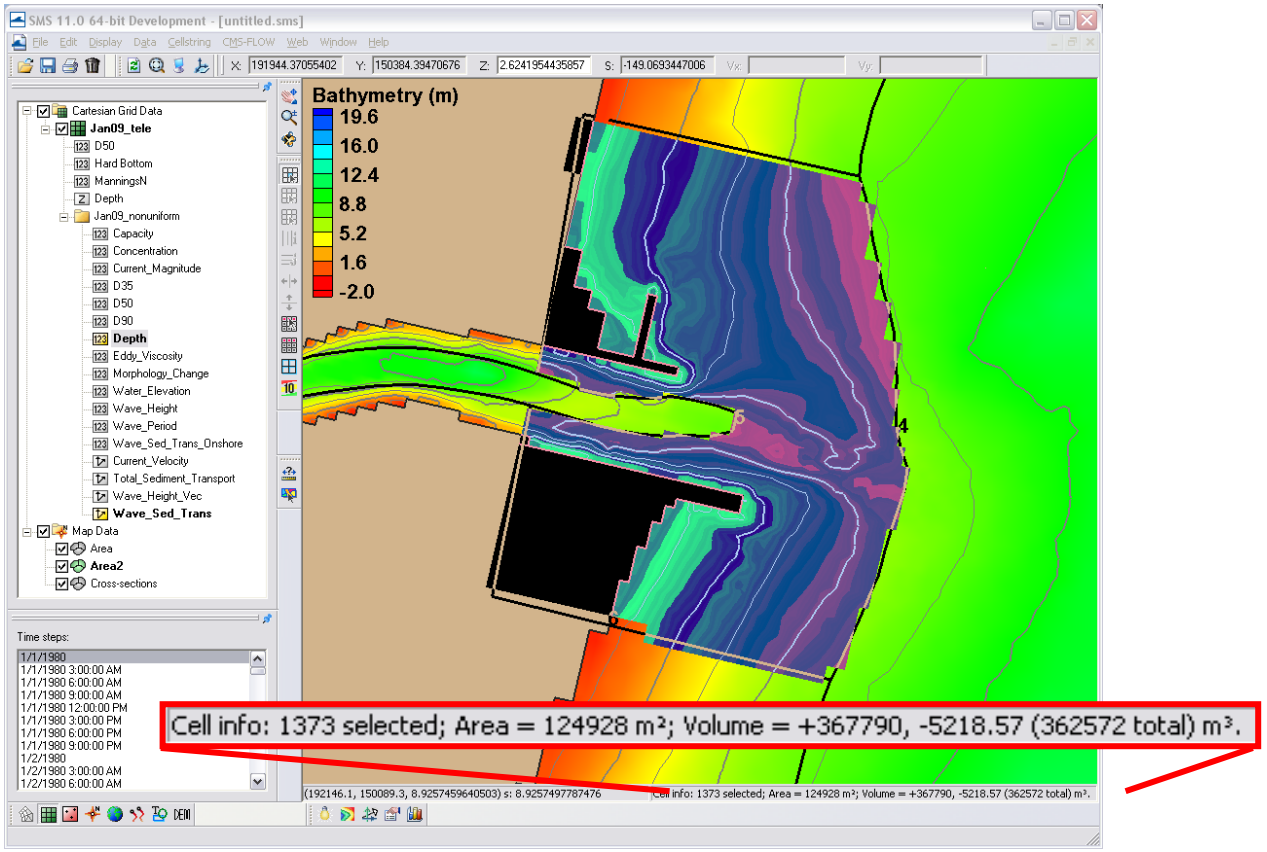


Figure 7-48. Selected cells from a polygon. Volume information given in lower, right corner of screen.

If depths are selected in the solution datatree as the measurement for volume calculation (one of the direct outputs in the CMS solution file), volume is positive and represents the space that the water occupies. To get the volume of a morphologic feature, a base level to calculate above must be chosen. Navigation depths or depth of closures are often used as a base level for a quick estimation of volume. The area of the cells is also included in basic cell info at the bottom of the screen, as illustrated in Figure 7-48. Multiplying this area by the base level (with respect to the model datum) will produce a base volume of which the calculated volumes can be subtracted from.

8 Support Policy

Technical support for CMS users within the USACE is provided through an annual maintenance program subscription. USACE users are encouraged to contact the CMS technical team before beginning a new project in order to discuss the feasibility of the using CMS, especially if the project requires advanced CMS features such as multiple grain sizes and hydraulic structures. Extended support can be arranged under a separate reimbursable project agreement.

Users may contact the CMS development team by email at:

chl.cms@usace.army.mil

or through the CMS moderated subscription list. Information on how to join and use the subscription list is available at:

http://cirp.usace.army.mil/other/subscription.php

Support for the SMS interface is provided directly by Aquaveo. Information on technical support for the SMS interface can be found at:

http://www.aquaveo.com/technical-support

Support to users outside of the USACE is limited reporting of suspected bugs or errors. The CMS is being developed and improved continuously. The latest version of the CMS executables a copy of the CMS License Agreement may be found at:

http://cirpwiki.info/wiki/CMS_Releases

and

http://cirp.usace.army.mil/products/cms.php

9 Summary

The Coastal Modeling System (CMS) User Manual serves as a reference guide to the applied theory, numerical methodology, and the application of the modeling system. The CMS is an integrated wave, current, sediment transport and morphology change model available in the Surface-water Modeling System (SMS). The first chapter on theoretical background summarizes the governing hydrodynamic and sediment transport theory and empirical equations. The Numerical Methods chapter describes the numerical solvers and schemes used in the implicit CMS-Flow. Chapter 4, the User Guide, gives step by step guidance on how to setup and run the CMS and how to analyze the modeling results as part of the calibration process.

The CMS was developed under the Coastal Inlets Research Program (CIRP), an Operations & Maintenance Navigation research program at the Coastal and Hydraulics Laboratory, Engineer Research and Development Center, U.S. Army Corps of Engineers. Information presented herein was prepared for use in SMS 11.0 and higher, although many features may exist in earlier versions of SMS. The latest guidance, documentations and downloads are available at: http://cirp.usace.army.mil/ and from the CIRP wiki: http://cirp.usace.army.mil/wiki/Main_Page.

10 List of Symbols

Variable	Meaning	Units (SI)
f_c	Coriolis parameter	1/s
h	Time -averaged total water depth $h = \zeta + \eta$	m
$\overline{\eta}$	Time-averaged water surface elevation above the still water line	m
ζ	Still water depth	m
S^m	Source term due to precipitation, evaporation and sub-grid structures	m/s
V_i	Total flux velocity defined as $V_i = U_i + U_{wi}$	m/a
U_i	Time- and depth-averaged current velocity vector (i.e. Eulerian velocity)	m/s
U_i^S	Depth-averaged Stokes velocity	m/s
g	Gravitational constant (9.806 m/s^2)	m/s^2
p_{atm}	Atmospheric pressure	Pa
ρ	Water density (~1025 kg/m^3)	kg/m^3
ν_t	Turbulent eddy viscosity	m^2/s
τ_{si}	Wind surface stress vector	N/m^2
τ_{wi}	Wave stress vector	N/m^2
τ_{bi}	Combined wave-current mean bed shear stress vector	N/m^2
C_{salt}	Depth-average salinity concentration	ppt
u_w	Peak bottom orbital velocity based on the significant wave height and linear wave theory	m/s
u_{rms}	Peak bottom wave orbital velocity based on the root-mean-squared wave height	m/s
ρ_a	Air density at sea level	kg/m^3
c_D	Wind drag coefficient	-
W_i	10-m wind speed vector	m/s
W	10-m wind velocity magnitude $= \sqrt{W_i W_i}$	m/s
q_{b*}	Equilibrium bed load transport	m^2/s
q_{s*}	Equilibrium suspended load transport	m^2/s
q_{t*}	Equilibrium total load transport	m^2/s

d_{50}	median grain size	m
s	sediment specific gravity or relative density	-
Θ_c	Shields parameters due to currents	-
$\Theta_{cw,m}$	Mean Shields parameters due to waves and currents	-
Θ_{cw}	Maximum Shields parameters due to waves and currents	-
Θ_{cr}	Critical Shields parameter.	-
d_*	Dimensionless grain size based on the median grain size	-
d_{*k}	Dimensionless grain size based on the k^{th} size class	-

11 References

Andrews, D.G., and McIntyre, M.E. 1978a. An exact theory of nonlinear waves on a Lagrangian mean flow. Journal of Fluid Mechanics. 89, 609-646.

Aquaveo. 2010. SMS: XY Series Files (*.xys). http://www.xmswiki.com/xms/SMS:XY_Series_Files_(*.xys).

Armanini, A., and di Silvio, G. 1986. "Discussion on the paper 'A depth-averaged model for suspended sediment transport', " by Galappattti, G., and Vreugdenhil, C.B. Journal of Hydraulic Research. 24(5), 437-441.

Batten, B.K., and N.C. Kraus. 2006. "Evaluation of Downdrift Shore Erosion, Mattituck Inlet, New York: Section 111 Study," *Technical Report ERDC/CHL-TR-06-1*, US Army Engineer Research and Development Center, Coastal and Hydraulics Laboratory, Vicksburg, Mississippi

Beck, T.B., and N.C. Kraus. 2010. Shark River Inlet, New Jersey, Entrance Shoaling: Report 2, Analysis with Coastal Modeling System. Technical Report ERDC/CHL-TR-10-4, U.S. Army Engineer Research and Development Center, Coastal and Hydraulics Laboratory, Vicksburg, MS.

Benson, M.A., and Dalrymple, T. 1967. General field and office procedures for indirect discharge measurements: U.S. Geological Survey Techniques of Water-Resources Investigations book 3, Chap. Al, 30 p.

Butler, C.D., Richards, D.R., Wallace, R.M., Jones, N.L., and Jones, R. 2007. "eXtensible Model Data Format (XMDF)," *Coastal and Hydraulics Laboratory Special Report ERDC SR-07-1*, U.S. Army Engineer Research and Development Center, Vicksburg, MS.

Buttolph, A.M., Reed, C.W., Kraus, N.C., Ono, N., Larson, M., Camenen, B., Hanson, H.,Wamsley, T., and Zundel, A. K. 2006. Two-dimensional depth-averaged circulation model CMS-M2D: Version 3.0, Report 2: Sediment transport and morphology change. *Tech. Rep. ERDC/CHL TR-06-9*, U.S. Army Engineer Research and Development Center, Coastal and Hydraulic Engineering, Vicksburg, MS.

Bye, J.A.T. 1985. Large-Scale Momentum Exchange in the Coupled Atmosphere–Ocean, Coupled Ocean–Atmosphere Models," Elsevier Science Publishers, Amsterdam, pp. 51–61.

Byrnes, M.R., Griffee, S.F., and Osler. M.S. 2010. Channel Dredging and Geomorphic Response at and Adjacent to Mobile Pass, Alabama. *Technical Report ERDC/CHL-TR-10-8*, U.S. Army Engineer Re-

search and Development Center, Coastal and Hydraulics Laboratory, Vicksburg, MS. http://cirp.usace.army.mil/pubs/html/10-Byrnes_TR-10-8.html, accessed June 2011

Camenen, B., and Larson, M. 2005. A general formula for non-cohesive bed load sediment transport. *Estuarine, Coastal and Shelf Science*, 63, 249–260.

Camenen, B., and Larson, M. 2007. A unified sediment transport formulation for coastal inlet application. *Technical report ERDC/CHL CR-07-1*, US Army Engineer Research and Development Center, Vicksburg, MS.

Camenen, B., and Larson, M. 2008. A general formula for noncohesive suspended sediment transport. *Journal of Coastal Research*, 24 (3), 615–627.

Cebeci, T., and Bradshaw, P. 1977. Momentum transfer in boundary layers. Hemisphere, Washington D.C.

CEM (2002). Coastal Engineering Manual. Engineer Manual 1110-2-1100, U.S. Army Corps of Engineers, Washington, D.C. (in 6 volumes).

Choi, S.K., Nam, H.Y., and Cho, M. 1995. A comparison of higher-order bounded convection schemes. *Computational Methods in Applied Mechanics and Engineering*, 121, 281-301.

Chawla A., and J. T. Kirby. 2002. Monochromatic and random wave breaking at blocking points. *Journal of Geophysical Research* 107(C7), 10.1029/2001JC001042.

Coriolis, G.G. 1835. Sur les équations du mouvement relatif des systèmes de corps. *J. de l'Ecole royale polytechnique*, 15, 144–154.

Dawe, J.T., and Thompson, L. 2006. Effect of ocean surface currents on wind stress, heat flux, and wind power input to the ocean. *Geophysical Research Letters*. 33, L09604.

Davies, A.G., Soulsby, R.L., and King, H.L. 1988. A numerical model of the combined wave and current bottom boundary layer, *Journal of Geophysical Research*, 93(C1), 491-508.

Demirbilek, Z. and Rosati, J.D. 2011. Verification and Validation of the Coastal Modeling System: Report I, Executive Summary. *Tech. Report ERDC/CHL-TR-11-10*, U.S. Army Engineer Research and Development Center, Coastal and Hydraulics Laboratory, Vicksburg, MS.

Falconer, R.A. 1980. Modeling of planform influence on circulation in harbors. *Proceedings Coastal Engineering Conference* '17. ASCE, 2,726-2, 744.

Ferziger, J. H., and Peric, M. (1997). Computational Methods for Fluid Dynamics, Springer-Verlag, Berlin/New York, 226 p.

Folk R.L, and Ward, W.C. 1957. Brazos River bar: a study in the significance of grain size parameters. *Journal of Sedimentary Petrology*, 27, 3–26.

Fredsoe, J. 1984. Turbulent boundary layer in wave-current motion. *Journal of Hydraulic Engineering*, ASCE, 110, 1103-1120.

Gallappatti, G., and Vreugdenhil, C.B. 1985. A depth-integrated model for suspended sediment transport. Journal of Hydraulic Research, 23(4), 359-377.

Gauckler, P. 1867. Etudes Théoriques et Pratiques sur l'Ecoulement et le Mouvement des Eaux. Comptes Rendues de l'Académie des Sciences, Paris, France, 64, 818–822.

Graf, W.H., and Altinakar, M. 1998. Fluvial Hydraulics. Wiley & Sons Ltd., 681 pp.

Grant, W.D., and Madsen, O.S. 1979. Combined wave and current interaction with a rough bottom, *Journal of Geophysical Research*, 86(C4), 1797-1808.

Harlow, F.H. and Welch, J.E. 1965. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *Physics of Fluids*. 8, 2182.

Hirano, M. 1971. River bed degradation with armouring, Trans. of Jap. Soc. Civ. Eng., 3(2), 194-195.

Hsu, S.A. 1988. Coastal meteorology. Academic Press, San Diego, CA.

Huynh-Thanh, S., and Temperville, A. 1991. A numerical model of the rough turbulent boundary layer in combined wave and current interaction," in Sand Transport in Rivers, Estuaries and the Sea, eds. R.L. Soulsby and R. Bettess, pp.93-100. Balkema, Rotterdam.

Jasak H., Weller, H.G., And Gosman, A.D. 1999. High Resolution Nvd Differencing Scheme For Arbitrarily Unstructured Meshes. *International Journal* of *Numerical Methods for Fluids*, 31, 431–449.

Johnson, I.G., 1966. Wave Boundary Layer and Function Factors. Proceedings 10th Coastal Engineering Conference. ASCE, pp. 127-148.

Jonsson, I.G. 1990. Wave-current interactions. Chapter 7, *The Sea*. B. Le Mehaute and D. Hanes (ed.). New York, NY: John Wiley and Sons, 65-120.

Karim M.F., and Kennedy J.F. 1982. IALLUVIAL: A Computer-Based Flow- and Sediment-Routing for Alluvial Stream and Its Application to the Missouri River. Iowa Institute of Hydraulic Research Report No. 250, University of Iowa; Iowa City, Iowa.

Kraus, N.C., Lin, L., Batten, B.K., and Brown, G.L. 2006. Matagorda Ship Channel, Texas. jetty stability study. *Coastal and Hydraulics La-*

boratory Technical Report ERDC/CHL TR-06-7. Vicksburg, MS: U.S. Army Engineer Research and Development Center.

Lai, Y.G. 2010. Two-dimensional depth-averaged flow modeling with an un-structured hybrid mesh. *Journal of Hydraulic Engineering*, 136, 12-23.

Lamb, H., 1932. *Hydrodynamics*. 6th ed. New York: Dover Publications.

Larson, M., and Kraus, N.C. 2002. *NMLONG: Numerical model for simulating longshore current*. Coastal and Hydraulics Laboratory Technical Report ERDC/CHL TR-02-22. Vicksburg, MS: U.S. Army Engineer Research and Development Center.

Le Provost, C., Genco, M.L., Lyard, F., Vincent, P., and Canceil, P. 1994. Spectroscopy of the ocean tides from a finite element hydrodynamic model. Journal of Geophysical Research, 99(C12), 24777–24797.

Li, H., Brown, M.E., Smith, T.D., and Podoski, J.H. 2009. Evaluation of proposed channel on circulation and morphology change at Kawaihae Harbor and Pelekane Bay, Island of Hawaii, HI. *Tech. Report ERDC/CHL TR-09-19*. Vicksburg, MS: U.S. Army Engineer Research and Development Center.

Li, H., Reed, C.W., and Brown, M.E. 2012. Salinity calculations in the Coastal Modeling System. *Coastal and Hydraulics Engineering Technical Note ERDC/CHL CHETN-IV-80*. Vicksburg, MS: U.S. Army Engineer Research and Development Center.

Lin, B.N. 1984. Current study of unsteady transport of sediment in China. *Proceedings Japan-China Bilateral Seminar on River Hydraulics and Engineering Experience*, July, Tokyo-Kyoto-Saporo, Japan, 337-342.

Lin, L., Demirbilek, Z., Mase, H., Zheng, J., and Yamada, F. 2008. CMS-Wave: a nearshore spectral wave processes model for coastal inlets and navigation projects. *Tech. Report ERDC/CHL TR-08-13*. Vicksburg, MS: U.S. Army Engineer Research and Development Center.

Lin, L., Demirbilek, Z., and Mase, H. 2011a. Recent capabilities of CMS-Wave: A coastal wave model for inlets and navigation projects. *Journal of Coastal Research*, Special Issue 59, 7-14.

Lin, L., Demirbilek, Z., Thomas, R., and Rosati III, J. 2011b. Verification and Validation of the Coastal Modeling System, Report 2: CMS-Wave, *Technical Report ERDC/CHL-TR-11-10*, US Army Engineer Research and Development Center, Coastal and Hydraulics Laboratory, Vicksburg, Mississippi.

Lin, L., Watts, I., and Demirbilek, Z. 2012. CMS-Wave Model: Part 3. Grid Nesting and Application Example for Rhode Island South Shore Regional Sediment Management Study (In Press)

Longuet-Higgins, M.S., and Stewart, R.W. 1961. The changes in amplitude of short gravity waves on steady non-uniform currents. *Journal of Fluid Mechanics*, 10(4), 529-549.

Lyard, F., Lefevre, F., Letellier, T., and Francis, O. 2006. Modelling the global ocean tides: modern insights from FES2004. Ocean Dynamics, 56, 394–415.

MacDonald, N.J., Davies, M.H., Zundel, A.K., Howlett, J.D., Lackey, T. C., Demirbilek, Z., and Gailani, J.Z. 2006. PTM: Particle Tracking Model; Report 1: Model theory, implementation, and example applications. Coastal and Hydraulics Laboratory Technical Report ERDC/CHL-TR-06-20. Vicksburg, MS: U.S. Army Engineer Research and Development Center.

Madsen, O.S., 1994. Spectral wave–current bottom boundary layer flows. Proceedings of the 24th Conference on Coastal Engeering, ASCE, Kobe, Japan, 1, 384–398.

Mase, H., Oki, K., Hedges, T.S., and Li, H.J. 2005. Extended energy-balance-equation wave model for multidirectional random wave transformation. Ocean Engineering, 32(8-9), 961-985.

Mei, C. 1983. The Applied Dynamics of Ocean Surface Waves, John Wiley, New York.

Militello, A., Reed, C.W., Zundel, A.K., and Kraus, N.C. 2004. Two-dimensional depth-averaged circulation model CMS-M2D: Version 2.0, Report 1, Technical documentation and user's guide. Coastal and Hydraulics Laboratory Technical Report ERDC/CHL TR-04-02. Vicksburg, MS: U.S. Army Engineer Research and Development Center.

Mukai, A.Y., Westerlink, J.J., Luettich, R.A.Jr., and Mark, D. 2002. East-Coast 2001, A tidal constituent database for Western North Atlantic, Gulf of Mexico and Caribbean Sea, Coastal and Hydraulics Laboratory Technical Report ERDC/CHL TR-02-04. Vicksburg, MS: U.S. Army Engineer Research and Development Center.

Myrhaug, D., Holmedal, L.E., Simons, R.R., and MacIver, R.D., 2001. Bottom friction in random waves plus current flow. Coastal Engineering, 43, 75-92.

Nicholson, J., and O'Connor, B.A. 1986. Cohesive sediment transport model. *Journal of Hydraulic Engineering*, 112(7): 621-640.

Nielsen, P. 1992. Coastal bottom boundary layers and sediment transport. World Scientific, Singapore, 324 pp.

Pacanowski, R.C. 1987. Effect of equatorial currents on surface wind stress. *Journal of Physical Oceanography*. 17, 833–838.

Parker, G., Kilingeman, P.C., and McLean, D.G. 1982. Bed load and size distribution in paved gravel-bed streams," Journal of the Hydraulics Division, ASCE, 108(4), 544-571.

Patankar, S. V. 1980. Numerical heat transfer and fluid flow, Hemisphere, New York.

Phillips, O.M. 1977. The dynamics of the upper ocean. (2nd ed.). Cambridge University Press.

Powell, M.D., Vickery, P.J., and Reinhold, T.A. 2003. Reduced drag coefficient for high wind speeds in tropical cyclones. Nature, 422, 279-283.

Reed, C.W., and Lin, L. 2011. "Analysis of Packery Channel Public Access Boat Ramp Shoreline Failure," Journal of Coastal Research Special Edition, Coastal Education and Research Foundation, Inc., Special Issue, 59, 150-155.

Reed, C.W., Brown, M.E., Sánchez, A., Wu, W., and Buttolph, A.M. 2011. "The Coastal Modeling System Flow Model (CMS-Flow): Past and Present,"Journal of Coastal Research, Special Edition, 59, 1-6.

Rhie, T.M. and Chow, A. 1983. Numerical study of the turbulent flow past an isolated airfoil with trailing-edge separation. AIAA J., 21, 1525–1532.

Rosati, J.R., Frey, A.E., Brown, M.E., and Lin, L. 2011. Analysis of Dredged Material Placement Alternatives for Bottleneck Removal, Matagorda Ship Channel, Texas," ERDC/CHL-TR-11-2, U.S. Army Engineer Research and Development Center, Coastal and Hydraulics Laboratory, Vicksburg, MS. http://cirp.usace.army.mil/pubs/html/11-Rosati-Frey-TR-11-2.html, accessed 7 June 2011.

Ruessink, B.G., Miles, J.R., Feddersen, F., Guza, R.T. and Elgar, S., 2001. Modeling the alongshore current on barred beaches. Journal of Geophysical Research, 106(C10): 22,451-22,464.

Saad, Y. 1993. A flexible inner-outer preconditioned GMRES algorithm. SIAM Journal Scientific Computing, 14, 461–469.

Saad, Y. 1994. ILUT: a dual threshold incomplete ILU factorization. Numerical Linear Algebra with Applications, 1, 387-402.

Saad, Y. and Schultz, M.H., 1986. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. SIAM Journal of Scientific and Statistical, Computing, 7, 856-869.

Saad, Y. 1996. Iterative methods for sparse linear systems. PWS Publishing Company, 528 pp.

Sakai, S., Kobayashi, N., and Koike, K. 1989. Wave breaking criterion with opposing current on sloping bottom: an extension of Goda's breaker index. *Annual Journal of Coastal Engineering* 36:56-59, JSCE (in Japanese).

Sánchez, A., and Wu, W. 2011a. A non-equilibrium sediment transport model for coastal inlets and navigation channels. *Journal of Coastal Research*, Special Issue 59, 39-48.

Sánchez, A., and Wu, W. 2011b. Nonuniform sediment transport modeling and Grays Harbor, WA. *Proceedings Coastal Sediments '11*, [In Press].

Sánchez, A., Wu, W. Rosati, J.D., Demirbilek, Z. Li, L., Rosati, J., Thomas, R., Reed, C., Watts, I., and Brown, M. 2011a. Validation of the Coastal Modeling System: Report III, Hydrodynamics. *ERDC/CHL-TR-11-10*, US Army Engineer Research and Development Center, Coastal and Hydraulics Laboratory, Vicksburg, MS.

Sánchez, A., Wu, W. Rosati, J.D., Demirbilek, Z. Li, L., Rosati, J., Thomas, R., Reed, C., Watts, I., and Brown, M. 2011b. Validation of the Coastal Modeling System: Report IV, Sediment Transport and Morphology Change. ERDC/CHL-TR-11-10, US Army Engineer Research and Development Center, Coastal and Hydraulics Laboratory, Vicksburg, MS.

Shepard, D. 1968. A two-dimensional interpolation function for irregularly-spaced data. Proceedings of the 1968 ACM National Conference. pp. 517–524.

Shore protection manual 1984. 4th ed., 2 Vol., U.S. Army Engineer Waterways Experiment Station, U.S. Government Printing Office.

Smagorinsky, J. 1963. General circulation experiments with the primitive equations, Monthly Weather Review, 93(3), 99-164.

Soulsby, R.L. 1983. The bottom boundary layer of shelf seas. In Physical Oceanography of Coastal and Shelf Seas, ed. B. Johns, 189-266. Elsevier, Amsterdam.

Soulsby, R.L. 1987. Calculating bottom orbital velocity beneath waves. Coastal Engineering 11, 371–380.

Soulsby, R.L. 1995. Bed shear-stresses due to combined waves and currents. in Advanced in Coastal Morphodynamics, ed M.J.F Stive, H.J. de Vriend, J. Fredsoe, L. Hamm, R.L. Soulsby, C. Teisson, and J.C. Winterwerp, Delft Hydraulics, Netherlands. 4-20 to 4-23 pp.

Soulsby, R.L. 1997. Dynamics of marine sands. Thomas Telford, London.

Soulsby, R.L., and Whitehouse, R.J.S.W. 1997. Threshold of sediment motion in coastal environments. Proceedings of the Pacific Coasts and Ports '97 Conference, Christchurch, 1, pp. 149154. University of Canterbury, New Zealand.

Spargo, E.A., Westerlink, J.J., Luetick, R.A.Jr., and Mark, D. 2004. ENPAC 2003: A tidal constituent database for Eastern North Pacific Ocean, *Tech. Report ERDC/CHL-TR-04-12*, US Army Engineer

Research and Development Center, Coastal and Hydraulics Laboratory, Vicksburg, MS.

Stive, M.J.F. and De Vriend, H.J. 1994. Shear stresses and mean flow in shoaling and breaking waves. ASCE, New York, pp. 594-608.

Stone, H.L. 1968. Iterative Solution of Implicit Approximations of Multidimensional Partial Differential Equations, SIAM Journal of Numerical Analysis, 5(3), 530–538.

Svendsen, I.A. 2006. Introduction to nearshore hydrodynamics. World Scientific.

Swart, D.H., 1976. Predictive equations regarding transports. Proc. 15th Coastal Engng. Conf., Honolulu. Coastal ASCE.

van Doormal, J.P., and Raithby, G.D. 1984. Enhancements of the SIMPLE method for predicting incompressible fluid flows. Numerical Heat Transfer, 7, 147–163.

van Rijn, L.C. 1984. Sediment transport, part I: bed load transport. Journal of Hydraulic Engineering. ASCE, 110(10), 1431-1456.

van Rijn, L.C. 1989. Handbook: Sediment transport by currents and waves. Delft Hydraulics, Delft, The Netherlands.

van Rijn, L.C. 1993. Principles of sediment transport in rivers, estuaries, and coastal seas,

van Rijn, L.C. 1998. Principles of Coastal Morphology. AquaPublications, Amsterdam, The Netherlands.

van Rijn, L.C. 2007a. Unified View of Sediment Transport by Currents and Waves. I: Initiation of Motion, Bed Roughness, and Bed-load Transport. *Journal of Hydraulic Engineering*, 133(6), 649-667.

van Rijn, L.C. 2007b. Unified View of Sediment Transport by Currents and Waves. II: Suspended Transport, *Journal of Hydraulic Engineering*, 133(6), 668-689.

Walstra, D.J.R., Roelvink, J.A., and Groeneweg, J. 2000. Calculation of wave-driven currents in a 3D mean flow model, 27th International Conference on Coastal Engineering, Sydney, 1050-1063.

Wang, P., Beck, T.M., and Roberts T.M. 2011. Modeling Regional-Scale Sediment Transport and Medium-term Morphology Change at a Dual Inlet System Examined with the Coastal Modeling System (CMS): A Case Study at Johns Pass and Blind Pass, West-central Florida. *Journal of Coastal Research*, Special Issue 59, pp. 49-60.

Watanabe, A. 1987. 3-dimensional numerical model of beach evolution. *Proceedings Coastal Sediments '87*, 802-817.

Wu, W. 1991. The study and application of 1-D, horizontal 2-D and their nesting mathematical models for sediment transport. PhD disserta-

tion,Wuhan Univ. of Hydraulic and Electric Engineering, Wuhan, China.

Wu, W. 2004. Depth-averaged 2-D numerical modeling of unsteady flow and nonuniform sediment transport in open channels. Journal of Hydraulic Engineering, ASCE, 135(10) 1013-1024.

Wu, W. 2007. Computational River Dynamics. Taylor & Francis, 494 p. Wu, W., Sánchez, A., Zhang, M. 2010. An implicit 2-D depth-averaged finite volume model of flow and sediment transport in coastal waters. *Proceeding of the International Conference on Coastal Engineering*, North America, 1 Feb 2011, Available at: http://journals.tdl.org/ICCE/article/view/1431. Date accessed: June 12.

Wu, W., Wang, S.S.Y., and Jia, Y. 2000. Nonuniform sediment transport in alluvial rivers. *Journal of Hydraulic Research*, IAHR, 38(6), 427-434.

Wu, W., Altinakar, M., and Wang, S.S.Y. 2006. Depth-averaged analysis of hysterersis between flow and sediment transport under unsteady conditions. *International Journal of Sediment Research*, 21(2), 101-112.

Wu, W., Sánchez, A., and Mingliang, Z. 2010. An implicit 2-D depth-averaged finite-volume model of flow and sediment transport in coastal waters. *Proceeding of the International Conference on Coastal Engineering*, [In Press]

Wu, W., Sánchez, A., and Mingliang, Z. 2011. An implicit 2-D shallow water flow model on an unstructured quadtree rectangular grid," *Journal of Coastal Research*, [In Press]

Zarillo, G.A., Brehin F. G. A. 2007. Hydrodynamic and Morphologic Modeling at Sebastian Inlet, FL. *Proceedings of Coastal Sediments '07*, New Orleans, LA, ASCE Press, 1297-1311

Zheng, J.,H. Mase, Z. Demirbilek, and L. Lin. 2008. Implementation and evaluation of alternative wave breaking formulas in a coastal spectral wave model. *Ocean Engineering*

Zhu, J. 1991. A low-diffusive and oscillation-free convection scheme. *Communications in Applied Numerical Methods*, 7, 225-232.

Zwart, P.J., Raithby, G. D., Raw, M.J. 1998. An integrated space-time finite volume method for moving boundary problems. *Numerical Heat Transfer*, B34, 257.

Zundel, A.K., 2006. Surface-water Modeling System reference manual – Version 9.0. Brigham Young University Environmental Modeling Research Laboratory, Provo, UT.

12 Appendix A: Description of Input Files

CMS-Flow

Telescoping Grid File (*.tel)

The telescoping Grid File is saved in SMS 11.0 when saving a telescoping grid. The first line contains a header which says "CMS-Telescoping". The second line contains four elements corresponding to the (1) grid orientation *theta*, (2) grid origin in the x-direction x_0, (3) grid origin in the y-direction y_0, and (4) total number of cells N (including inactive cells). Lines 3 through N+3 contain the following column data:

Table A1. Description of column data in the CMS-Flow Telescoping Grid File.

Column	Symbol	Variable
1	*i*	Sequential index
2	*x*	Cell-centered x-coordinate of cell *i* in m
3	*y*	Cell-centered y-coordinate of cell *i* in m
4	Δx	Grid size in x-direction of cell *i* in m
5	Δy	Grid size in *y*-direction of cell *i* in m
6	*iN1*	Index of first neighboring cell to the North direction
7	*iN2*	Index of second neighboring cell to the North direction
8	*iE1*	Index of first neighboring cell to the East direction
9	*iE2*	Index of second neighboring cell to the East direction
10	*iS1*	Index of first neighboring cell to the South direction
11	*iS2*	Index of second neighboring cell to the South direction
12	*iW1*	Index of first neighboring cell to the West direction
13	*iW2*	Index of second neighboring cell to the West direction
14	*Depth*	Still water depth in m. Positive values indicate wet cells and dry cells indicate dry cells.

Notes:

- Directions of neighboring cells are relative to the local grid axis (i.e. positive in x = East, negative in x = West, positive in y = North, and negative in y = South).
- Indexes equal to 0 indicate the absence of a neighboring cell.
- If a cell is assigned as an inactive cell, -999
- Because the file is ASCII, the file size can be relatively large and difficult to view in WordPad or Notepad. To view the telescoping grid file, it is recommended to use a more advanced text editor such as UltraEdit or Textpad.

An example of the first 6 lines of a CMS-Flow telescoping grid file is provided below.

```
CMS-Telescoping
0  -2.32  -0.01  13522
1  0.085 0.235  0.17 0.094  17 0 2 0 169 168 0 0 0.188531
2  0.255 0.235  0.17 0.094  18 0 3 0 171 170 1 0 0.188543
3  0.425 0.235  0.17 0.094  19 0 4 0 173 172 2 0 -999
4  0.595 0.235  0.17 0.094  20 0 5 0 175 174 3 0 0.188566
```

A simple Matlab script is provided below to read the *.tel file.

```
function out = read_cmstel(telfile)
% out = read_cmstel(telfile)
% Reads a CMS telescoping grid file
% and output all variables to a structure array
% written by Alex Sanchez, USACE
fid = fopen(telfile,'r');
fgets(fid); %Skip header line
data=fscanf(fid,'%f %f %f %d',4);
out.ncells = data(4);
out.x0 = data(2); out.y0 = data(3); out.angle = data(1);
data=fscanf(fid,'%d %f %f %f %f %d %d %d %d %d %d %d %d %f',...
    [14,out.ncells])';
fclose(fid);
out.id = data(:,1); out.x = data(:,2); out.y = data(:,3);
out.dx = data(:,4); out.dy = data(:,5);
out.iloc = data(:,6:13);
out.depth = data(:,14);
return
```

Card (Control) File

Control files are used for specifying input parameters, options, and boundary conditions, initial conditions and forcing. In CMS-Flow the Control File is the *.cmcards (card) file. The ASCII file uses a simple card format to specify model input in free format. Data Input Files depend on the specific model setup, and may include but are not limited to spatially variable

Grid File (*_grid.h5)

The SMS interface can save a scatter set in a binary format called the eXtensible Model Data Format (XMDF). The XMDF format (Butler et al. 2007) stores data in a much smaller file and decreases the time to load large data files, such as SHOALS or LIDAR surveys. Operation in XMDF format rather than ASCII format greatly reduces time of input and output operations.

Model Parameters File (*_mp.h5)

TBC

SMS XY Series File (*.xys)

TBC

SMS Time Series Data File (*.tsd)

TBC

Note:

- It is recommended to use ANSI file encoding for all ASCII files. CMS has problems reading Unicode or other encodings.

CMS-Wave

CMS-Wave Spectral File (*.eng)

TBC

CMS-Wave Model Parameters File (*.std)

Users can use SMS11 or higher versions, or simply edit the existing model control file *.std, to specify/select these advanced features.

The *.std has a maximum of 24 parameters - the first 15 parameters are more the basic ones as described in the CMS-Wave Technical Report (CHL-TR-08-13) while the remaining 9 parameters are relatively new for advanced CMS-Wave features.

Number	Variable	Argument Type	Options/Range	Description
1	iprp	integer	**0 - waves and wind input in *.eng 1 - waves only, neglect wind input in *.eng -1 - fast mode 2 - forced grid internal rotation 3 - without lateral energy flux**	Wave propagation mode.
2	icur	integer	**0 - no current input 1 - with current input *.cur 2 -with *.cur, use only the 1st set current data**	Current interaction.
3	ibk	integer	**0 - no wave breaking output 1 - output breaking indices 2 - output energy dissipation rate**	Wave breaking output option.
4	irs	integer	**0 - no wave radiation stress calculation or output 1 - calculate and output radiation stresses 2 - calculate and output radiation stresses plus setup/max-water-level**	Radiation stress and runup options.
5	kout	integer	**>= 0**	Number of special wave output location, output spectrum in *.obs and parameters in selhts.out.
6	ibnd	integer	**0 - no input a parent spectrum *.nst 1 - read *.nst, averaging input spectrum 2 - read *.nst, spatially variable spectrum input**	Nesting option.
7	iwet	integer	**0 - allow wet/dry, default 1 - without wet/dry -1 allow wet/dry, output swell and local**	Wetting and drying options.

			sea files -2 - output combined steering wav files -3 - output swell, local sea, and combined wav files	
8	ibf	integer	**0 - no bottom friction calc 1 - constant Darcy-Weisbach coef, c_f 2 -read variable c_f file, *.fric 3 - constant Mannings n 4 - read variable Mannings n file, *.fric**	Bottom friction option.
9	iark	integer	**0 - without forward reflection 1 - with forward reflection**	Forward reflection option.
10	iarkr	integer	**0 - without backward reflection, 1 - with backward reflection**	Backward reflection option.
11	akap	real	**0.0<=akap<=4.0**	Diffraction intensity coefficient.
12	bf	real	**>=0**	constant bottom friction coef c_f or n (typical value is 0.005 for c_f and 0.025 for Mannings n).
13	ark	real	**0.0<=ark<=1.0**	Constant forward reflection coef, global specification (0 for zero reflection, 1 for full reflection).
14	arkr	real	**0.0<=arkr<=1.0**	Constant backward reflection coef, global specification (0 for zero reflection, 1 for full reflection).
15	iwvbk	integer	**0 - Goda-extended 1 - Miche-extended 2 - Battjes and Janssen 3 - Chawla and Kirby)**	Option for the primary wave breaking formula.
16	nonln	integer	**0 - none, default 1 - nonlinear wave-wave interaction**	Nonlinear wave-wave interaction.
17	igrav	integer	**0 - none, default 1 - infra-gravity wave enter inlets**	Infragravity waves option.
18	irunup	integer	**0 - none, default 1 - automatic, runup relative to absolute datum 2 - automatic, runup relative to updated MWL**	Runup option.
19	imud	integer	**0 - none default 1 - Mud dissipation on**	Mud dissipation option. The kinematic viscosity is specified in mud.dat in units of m^2/sec.
20	iwnd	integer	**0 - none, default 1 - Spatially variable wind on**	Spatially variable wind field option.

				The winds are specified in wind.dat in units of m/s and in the reference frame of the CMS-Wave grid.
21	isolv	integer	**0 - GSR solver, default** **1 - ADI**	Matrix solver for CMS-Wave.
22	ixmdf	integer	**0 - output ascii, default** **1 - output xmdf** **2 - input & output xmdf**	XMDF input and output options.
23	iproc	integer	**>=0**	Number of threads for parallel computing. Optimum number is approximately equal to the total row number divided by 300. Only for isolv = 0.
24	iview	integer	**0 - half-plane, default** **1 - full-plane**	Half-plane/full-plane option. Users can provide additional input wave spectrum file wave.spc (same format as the *.eng) along the opposite side boundary an imaginary origin for wave.spc at the opposite corner; users can rotate the CMS-Wave grid by 180 deg in SMS to generate this wave.spc.

13 Appendix B: Description of Output Files

XMDF Solution Files (Binary Format)

By default, CMS global solution output files are written in a binary format called XMDF which is based on the HDF5 format which was developed to allow for portability between Windows and non-windows platforms. At present, the SMS reads and writes XMDF files. Additional binary formats are under review to be implemented in CMS for compatibility with other numerical models.

To make post-processing a little easier, CMS developers have written a Matlab code, shown below, that reads in the CMS XMDF solution file.

```
function varargout = read_cmsh5sol(filename,varargin)
% sol = read_cmsh5out(filename,...'dataset1','dataset2',...)
%
%DESCRIPTION:
% Reads a CMS solution file and creates
% a structure variable containing the solution
% datasets values and times
%
%INPUT:
% filename - input file name including full path
% varargin - dataset names
%
%OUTPUT:
% sol - structure variable containing dataset values and times
% varargout - variable length datasets corresponding to varargin
%
%USAGE:
% filename = 'test_sol.h5';
% sol = read_cmsh5sol(filename);
% wse = read_cmsh5sol(filename,'Water_Elevation');
% [wse,uv] = read_cmsh5sol(filename,'Water_Elevation',...
%            Current_Velocity');
%
% written by Alex Sanchez, USACE-ERDC-CHL
info = hdf5info(filename);
s = info.GroupHierarchy.Groups.Groups;
sol = struct();
for i=1:length(s);
    field = s(i).Name;
    ind = findstr(field,'/');
    field = field(ind(end)+1:end);
    if nargin>1 %select only few datasets
        for k=1:nargin-1
            if strcmpi(field,varargin{k})
                varargout{k}.Times = ...
                    double(hdf5read(s(i).Datasets(3)));
                varargout{k}.Values = ...
```

```
                    double(hdf5read(s(i).Datasets(4)));
                continue
            end
        end
    else %write all datasets
        field = regexprep(field,'(', '_');
        field = regexprep(field,')', '');
        field = regexprep(field,' ', '');
        try
            sol = setfield(sol,field,'Times',...
                double(hdf5read(s(i).Datasets(3))));
            sol = setfield(sol,field,'Values',...
                double(hdf5read(s(i).Datasets(4))));
        catch
            try
                for j=1:length(s(i).Groups)
                    field = s(i).Groups(j).Name;
                    ind = findstr(field,'/');
                    field = field(ind(end)+1:end);
                    field = regexprep(field,'(', '_');
                    field = regexprep(field,')', '');
                    field = regexprep(field,' ', '');
                    val = double(hdf5read( ...
                        s(i).Groups(j).Datasets(3)));
                    sol = setfield(sol,field,'Times',val);
                    val = double(hdf5read( ...
                        s(i).Groups(j).Datasets(4)));
                    sol = setfield(sol,field,'Values',val);
                end
            catch
                warning(['Unable to read ',field])
            end
        end
    end
end
if nargin>1 && nargout==1
    for k=1:nargin-1
        sol.(varargin{k}) = varargout{k};
    end
elseif nargin==1
    simlabel = info.GroupHierarchy.Groups.Name(2:end);
    sol.Simulation_Label = simlabel;
    varargout{1} = sol;
end
return
```

SMS Super Files (ASCII Format)

Another option in CMS is to write out solutions in an ASCII format. CMS developers decided to use a format previously implemented in the SMS for writing large ASCII datasets. This format requires three files to be present for each solution dataset: a super file, a scatter point file, and the scatter data file.

The super file simply contains a link to the scatter point file which stores the XY coordinates of each solution point. This file has a ".sup" file extension.

The scatter point file contains the name associated with this output file, the total number of solution points and a list of the XY coordinates of each solution point. This file has a ".xy" file extension.

The scatter data file typically contains information such as the name associated with this output file (same as the name given in 2. Above), reference time, time units, and then scalar or vector records for each output interval as specified in the parameter file. This file has a ".dat" file extension.

To activate ASCII output for solutions, the user must define the "GLOBAL_SUPER_FILES ON" card in the CMS parameter file.

Time Series (Observation Point) Files

As an alternative to frequent output intervals in the global solutions, which are often quite large after completion, CMS has the option to output information for individual locations inside the domain into ASCII files. These are referred to as observation points. Because information is stored for a very small selection of points, users can define this output to be at much smaller time intervals without resulting in very large files.

Individual scalar files are written for each type of information that is to be written. Presently, these types are: current velocity U and V components, water surface elevation, flow rates in X and Y direction, sediment transport rates in X and Y direction, suspended load sediment concentration, salinity concentration, and bed composition. Each file may contain output for one or more cells as defined in the SMS observation cell selection procedure. A flagged value of '-999.0000' is given for any computa-

tional cell that is considered dry at each output time. A partial example of the file format is shown below:

```
% time(hrs)    22883    60601    62753
     0.0000    0.0000 -999.0000    0.0000
     0.5000    0.0078 -999.0000    0.0077
     1.0000    0.0500 -999.0000    0.0495
…
  2062.5000    1.1605    1.1216    1.1614
  2063.0000    1.3580    1.3414    1.3580
  2063.5000    1.4783    1.4835    1.4788
  2064.0000    1.5100    1.5406    1.5174
```

In each file, there is a header row which gives a description of the contents of the following rows. In the example above, the first column contains “time in hours”, followed by three columns giving the cell numbers as saved by the SMS interface. There will be one row per output time.

14 Appendix C: Matlab Scripting

Scipting refers to the automation of running multiple CMS runs with different parameters, without manually having to create and edit each alternative. The scripting process can include the following steps:

1. Setting up alternatives
2. Creating batch file
3. Plotting and analyzing results

Scripting can be done using a variety of software programs. The examples shown here were written in Matlab because it is widely used, easy to read and convenient for plotting and analyzing results.

Setting Up Alternatives

In this example, 4 cases or alternatives (Figure 5.11.1) are set up using the Matlab script below. The script copies the base setup files into subfolders and then modifies specific CMS-Flow cards in the *.cmcards file. The settings for each case are setup using a structure variable with field names corresponding to each CMS-Flow card (e.g. `TIME_SERIES_INCREMENT`). Separating each case into its own subfolder keeps the input and output separate and also allows for the different cases to be run at the same time.

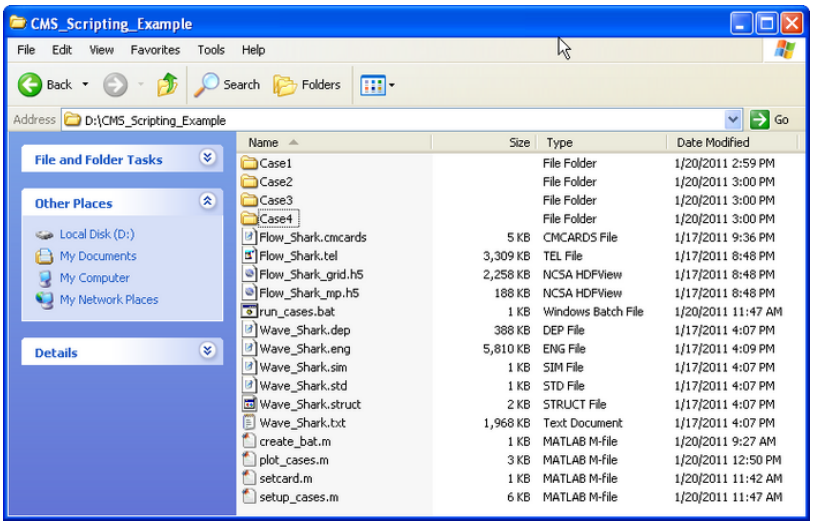


Figure D- 1. Example of scripting showing the files used.

Example: Setting up the cases in Matlab

```
% Matlab Script: setup_cases.m
clear all
flow = 'Flow_Shark';
wave = 'Wave_Shark';
ncases = 4; %Number of cases or alternatives
r(1).MANNINGS_N_DATASET = '"Manning_Alt1.h5"
"Flow_Shark/Datasets/ManningsN"';
r(1).WAVE_CURRENT_MEAN_STRESS = 'W09';
r(1).TIME_SERIES_INCREMENT = 1800;
r(2).MANNINGS_N_DATASET = '"Manning_Alt1.h5"
"Flow_Shark/Datasets/ManningsN"';
r(2).WAVE_CURRENT_MEAN_STRESS = 'DATA2';
r(2).TIME_SERIES_INCREMENT = 900;
r(3).MANNINGS_N_DATASET = '"Manning_Alt2.h5"
"Flow_Shark/Datasets/ManningsN"';
r(3).WAVE_CURRENT_MEAN_STRESS = 'W09';
r(3).TIME_SERIES_INCREMENT = 900;
r(4).MANNINGS_N_DATASET = '"Manning_Alt2.h5"
"Flow_Shark/Datasets/ManningsN"';
r(4).WAVE_CURRENT_MEAN_STRESS = 'DATA2';
r(4).TIME_SERIES_INCREMENT = 600;
for i=1:ncases
  d = ['Case',int2str(i)];
  if ~exist(d,'dir')
    mkdir(d)
  end
  copyfile([wave,'.*'],d)
  copyfile([flow,'.*'],d)
  copyfile([flow,'_mp.h5'],d);
  copyfile([flow,'_grid.h5'],d)
  cards = fieldnames(r(i));
  file = ['.\',d,'\Flow_Shark.cmcards'];
  fork=1:length(cards)
    setcard(file,cards{k},r(i).(cards{k}));
  end
end
return
```

The script above requires the subroutine below.

```
function setcard(cmcardsfile,card,value)
% setcard(file,card,value)
% Overwrites or appends a CMS-Flow card
% in the *.cmcards file
copyfile(cmcardsfile,'temp')
fid=fopen('temp','r');
fid2=fopen(cmcardsfile,'w');
nc=length(card);
ok = false(1);
if ~ischar(value)
  value = num2str(value);
end
while 1
  tline = fgets(fid);
  if ~ischar(tline), break, end
  if strncmp(card,tline,nc)
```

```
    fprintf(fid2,'%s       %s %s' ,card,value,tline(end));
    ok = true(1);
    continue
  end
  nline = length(tline);
  if(~ok && strcmp(tline(1:min(nline,14)),'END_PARAMETERS'))
    fprintf(fid2,'%s       %s %s',card,value,tline(end));
    fprintf(fid2,'%s' ,tline);
    break
  end
  fprintf(fid2,'%s' ,tline);
end
fclose(fid);
fclose(fid2);
delete('temp')
return
```

Creating a Batch File

Although it is possible to launch CMS from Matlab a batch file is preferable to use a batch file because it allows running all of the cases without opening Matlab.

```
% Matlab Script: create_bat.m
cmsexe = 'cms2d_v4r23_x64p.exe'; %CMS-Flow executable
batfile = 'run_cases.bat'; %Output batch file
numgroup = 2; %Number of cases to run in a group
fid = fopen(batfile,'w');
for i=1:ncases
  cmcards = ['.\Case',int2str(i),'\',flow,'.cmcards']; %cmcards file
  if mod(i,numgroup)~=0
    fprintf(fid,'START       %s %s %s',cmsexe,cmcards,char(10));
  else
   fprintf(fid,'START /WAIT %s %s %s',cmsexe,cmcards,char(10));
  end
end
fclose(fid);
return
```

The following text shows what the resulting batch file (*.bat) looks like

```
START  cms2d_v4b42_x64p.exe .\Case1\Flow_Shark.cmcards
START /WAIT cms2d_v4b42_x64p.exe .\Case2\Flow_Shark.cmcards
START cms2d_v4b42_x64p.exe .\Case3\Flow_Shark.cmcards
START /WAIT cms2d_v4b42_x64p.exe .\Case4\Flow_Shark.cmcards
```

To run the batch file, simply double click on the file and each case will launch separately in its own MS-DOS window.

Plotting

The following example reads the Observation Point time series output file (*_eta.txt) and plots the 3rd column corresponding to the second observation point.

```
% Matlab Script: plot_cases.m
close all
eta = cell(ncases,1);
for  i=1:ncases
  etafile = ['.\Case' ,int2str(i),'\',flow,'_eta.txt' ]; %Water elevation
  eta{i} = load(etafile);
end
figure
hold on
for i=1:ncases
  h = plot(eta{i}(:,1),eta{i}(:,3),'-');  %3 is the index is the observa-
tion point index
end
ylabel('Water elevation, m')
xlabel('Elapsed Time, hr')
return
```

15 Appendix D: Determining the sediment size Classes

In order to determine the appropriate grain sizes for a simulation it is useful to be able to determine ahead of time the size class fractions using a log-normal distribution for different median grain sizes and sorting. The Matlab example below determines the grain size distribution given the smallest and largest grain sizes, number of sediment sizes, median grain size and geometric standard deviation. The figure shows the computed grain size distribution.

```
clear all; close all
%--- Start Input ---
d1 = 0.234;  %mm, smallest grain size
dn = 2;      %mm, largest grain size
nsed = 5;    %number of grain sizes
d50 = 0.4;   %mm, median grain size
sg = 1.5;   %mm, geometric standard deviation
%--- End Input ---
%Characteristic diameters
d = exp(log(d1) + log(dn/d1)*((1:nsed)-1.0)/(nsed-1));
%Limits or bounds
dlim = zeros(1,nsed+1);
dlim(2:nsed)=sqrt(d(2:nsed).*d(1:nsed-1));
dlim(1)=d(1)*d(1)/dlim(2);
dlim(nsed+1)=d(nsed)*d(nsed)/dlim(nsed);
%Fractions
p = diff(dlim).*lognpdf(d,log(d50),log(sg));
p = p/sum(p);
%Plotting
figure
hold on
for k=1:nsed
  fill([dlim(k) dlim(k+1) dlim(k+1) dlim(k)],...
    [0 0 p(k) p(k)]*100,0.5*[1,1,1])
end
ylabel('Fraction, %')
xlabel('Grain size, mm')
set(gca,'box','On','TickDir','out',...
  'XMinorTick','OFF','YMinorTick','OFF')
[d',p']
return
```

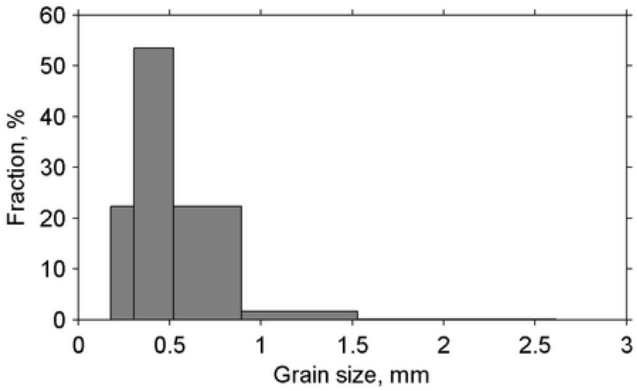


Figure E- 1. Computed grain size distribution from the Matlab script above.

16 Appendix E: Goodness of Fit Statistics

Brier Skill Score

The Brier Skill Score (BSS) is defined as

$$\mathrm{BSS} = 1 - \frac{\left\langle (x_m - x_c)^2 \right\rangle}{\left\langle (x_m - x_0)^2 \right\rangle} \tag{A1}$$

where the angled brackets indicate averaging, subscripts m, c, and 0 indicate measured, calculated, and initial values, respectively. The BSS ranges between negative infinity and one. A BSS value of 1 indicates a perfect agreement between measured and calculated values. Scores equal to or less than 0 indicates that the mean observed value is as or more accurate than the calculated values. The following quantifications are used for describing the BSS values: 0.8<BSS<1.0 = excellent, 0.6<BSS<0.8 = good, 0.3<BSS<0.6 = reasonable, 0<BSS<0.3 = poor, BSS<0 = bad.

Room Mean Squared Error

The Root Mean Squared Error (RMSE) is defined as

$$\mathrm{RMSE} = \sqrt{\left\langle (x_c - x_m)^2 \right\rangle} \tag{A2}$$

The RMSE has the same units as the measured data. Lower values of RMSE indicate a better match between measured and computed values.

Normalized Root Mean Squared Error

The Normalized Root Mean Squared Error (NRMSE) is

$$\mathrm{NRMSE} = \frac{\mathrm{RMSE}}{\mathrm{range}(x_m)} \tag{A3}$$

The NRMSE is often expressed in units of percent. The measured data range $\mathrm{range}(x_m)$ can be estimated as $\max(x_m) - \min(x_m)$. Lower values of NRMSE indicate a better agreement between measured and computed values.

Mean Absolute Error

The Mean Absolute Error (MAE) is defined as

$$\mathrm{MAE} = \left\langle \left| x_c - x_m \right| \right\rangle \tag{A4}$$

Normalized Mean Absolute Error

Similarly, the Normalized Mean Absolute Error (NMAE) is given by

$$\mathrm{NMAE} = \frac{\mathrm{MAE}}{\mathrm{range}(x_m)} \tag{A5}$$

The NMAE is often expressed in units of percent. Smaller values of NMAE indicate a better agreement between measured and calculated values.

Correlation Coefficient

Correlation is a measure of the strength and direction of a linear relationship between two variables. The correlation coefficient *R* is defined as

$$R = \frac{\left\langle x_m x_c \right\rangle - \left\langle x_m \right\rangle \left\langle x_c \right\rangle}{\sqrt{\left\langle x_m^2 \right\rangle - \left\langle x_m \right\rangle^2} \sqrt{\left\langle x_c^2 \right\rangle - \left\langle x_c \right\rangle^2}} \tag{A5}$$

A correlation of 1 indicates a perfect one-to-one linear relationship and -1 indicates a negative relationship. The square of the correlation coefficient describes how much of the variance between two variables is described by a linear fit. The interpretation of the correlation coefficient depends on the context and purposes. For the present work, the following qualifications are used: $0.7<R^2<1$ = strong, $0.4<R^2<0.7$ = medium, $0.2<R^2<0.4$ = small, and $R^2<0.2$ = none.

Bias

The Bias is defined as

$$\mathrm{Bias} = \left\langle x_c - x_m \right\rangle \tag{A6}$$

Positive values indicate over-prediction and negative values indicate under-prediction.

17 Appendix F: Piecewise Lagrangian Polynomial Interpolation

Piecewise polynomials in Lagrange form are given by

$$L(x) = \sum_{j=1}^{n+1} y_j l_j(x) \tag{17-1}$$

where

y_j = interpolation data values corresponding to x_j

n = order or the interpolation polynomial

$l_j(x)$ = Lagrange basis polynomials given by

$$l_j(x) = \prod_{\substack{1<k\le n+1 \\ k\neq j}} \frac{x - x_k}{x_j - x_k} \tag{17-2}$$

The Lagrange basis polynomials are such that

$$l_{j\neq i}(x_i) = 0,\ l_{j=i}(x_i) = 1,\ \therefore L(x_i) = y_i \tag{17-3}$$

One advantage of using Lagrange polynomials is that the interpolation weights (Lagrange basis functions l_j) are not a function of y_j. This property is useful when many interpolations are needed for the same x_j but different y_j such as in the case of interpolating spatial datasets in time.

18 Appendix G: Inverse Distance Interpolation

The inverse-distance interpolation also referred to as Shepard interpolation is given by (Shepard 1968)

$$\phi(\vec{x}) = \sum_{i=1}^{N} w_i \phi_i \tag{18-1}$$

where the interpolation weights are given by

$$w_i = \frac{d_i^{-p_s}}{\sum_j d_j^{-p_s}} \tag{18-2}$$

where

p_s = real and positive power parameter [-]

d = distance between the known points $\vec{x}_i$ and the unknown interpolation points $\vec{x}$ equal to the Euclidean norm $d = \left\| \vec{x} - \vec{x}_i \right\|$.

In this interpolation, the weight of each point decreases with distance from the interpolated point. One advantage of the inverse-distance interpolation is the interpolation weights are independent of the interpolation function, and therefore only need to be calculated once and can be saved for computational efficiency.

19 Appendix H: Providing Sea Buoy Data to CMS-Wave

Directional spectral data collected by NDBC or CDIP buoys can be processed as alternative source for wave input to CMS-Wave. Two examples are given below using CDIP 154 and NDBC 44025 standard spectral files for December 2009.

- NDBC buoy data – run **ndbc-spectra.exe** (FORTRAN) to read the NDBC standard directional wave file and generate the CMS-Wave input spectral *.eng.

    1. Download the NDBC standard monthly directional wave spectral file from http://www.nodc.noaa.gov/BUOY/buoy.html (e.g., 44025_200912) - see Figs 2.3.1 to 2.3.4 for accessing NDBC spectral data from the Web.

    2. In the DOS window, run **ndbc-spectra.exe**
    3. Responding to the on-screen input, type the NDBC spectral filename
    4. Type the starting timestamp (default value is 0) for saving output files
    5. Type ending timestamp (default is 99999999) for saving output files
    6. Type the time interval (hr) for saving output data
    7. Type 2 to save the CMS-Wave *.eng and *.txt files
    8. Type the CMS-Wave input spectrum filename (*.eng)
    9. Type the local shoreline orientation (the CMS-Wave grid y axis) in clockwise polar coordinates (deg, positive from North covering the sea, e.g., 180 deg for St Mary's Entrance, FL/GA, or 360 deg - the wave grid orientation angle in *.sim)
    10. Type the NDBC buoy location water depth (m) and then the CMS-Wave seaward boundary mean water depth (m), e.g. Buoy 44025 has a nominal depth of 36.3 m relative to Mean Sea Level
    11. Type 1 to include wind or 0 to skip the wind input information
    12. Type 1 or 2 or 3 for different choice of calculated frequency bins to complete the run – see Fig 2.3.5 for running **ndbc-spectra.exe** in DOS.

The output files include *.txt, *.eng, *.out (time series of wave parameters at the buoy), and *.dat (time series of shoreward wave parameters at the CMS-Wave offshore boundary).

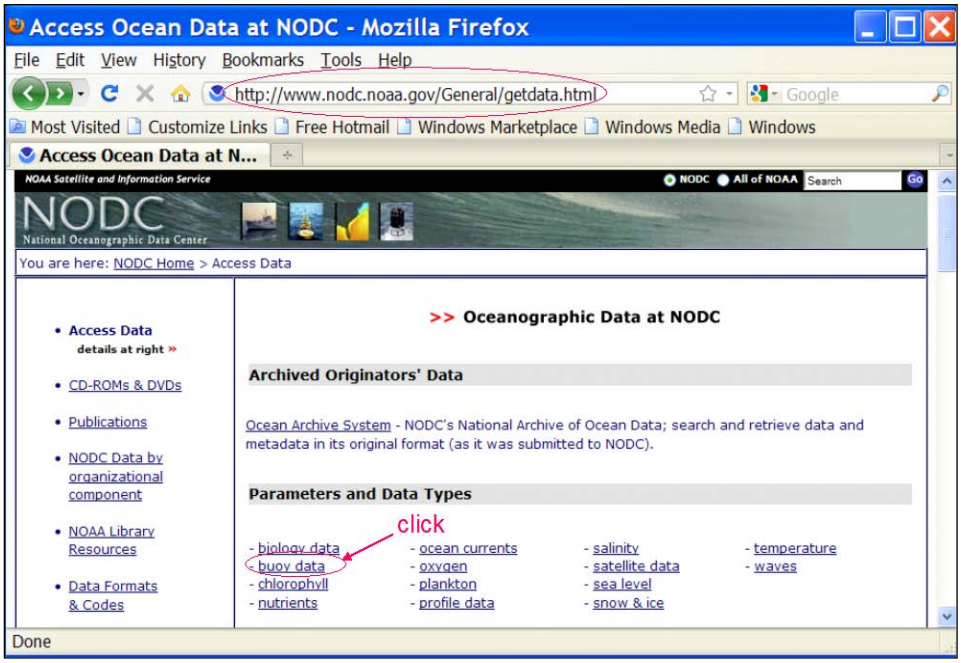


Figure G- 1. NODC buoy data access website.

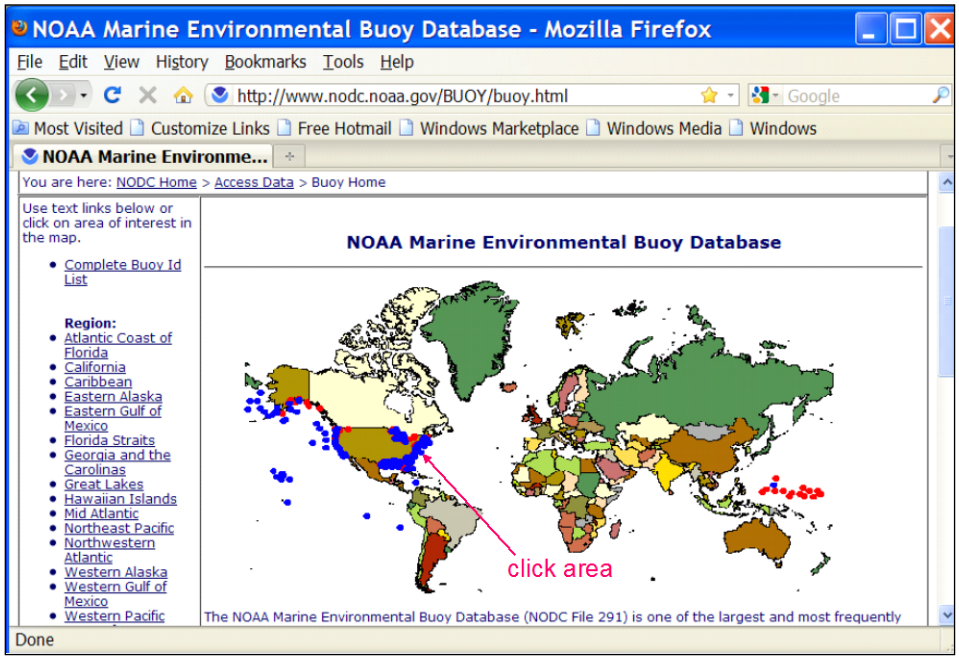


Figure G- 2. NODC buoy data access world map.

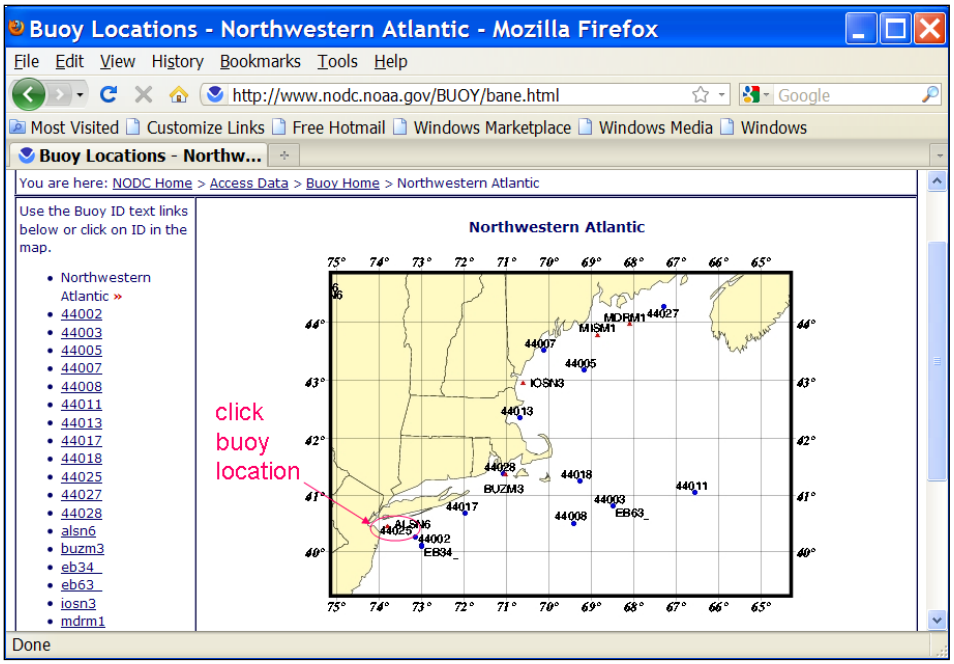


Figure G- 3. NODC buoy data access regional map.

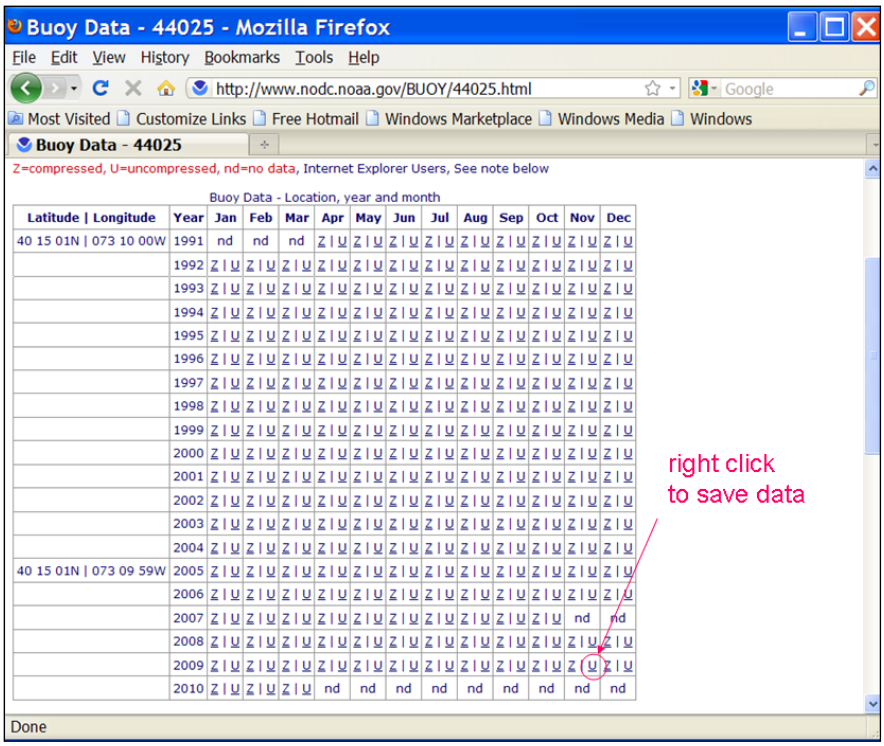


Figure G- 4. NDBC buoy spectral data download web page.

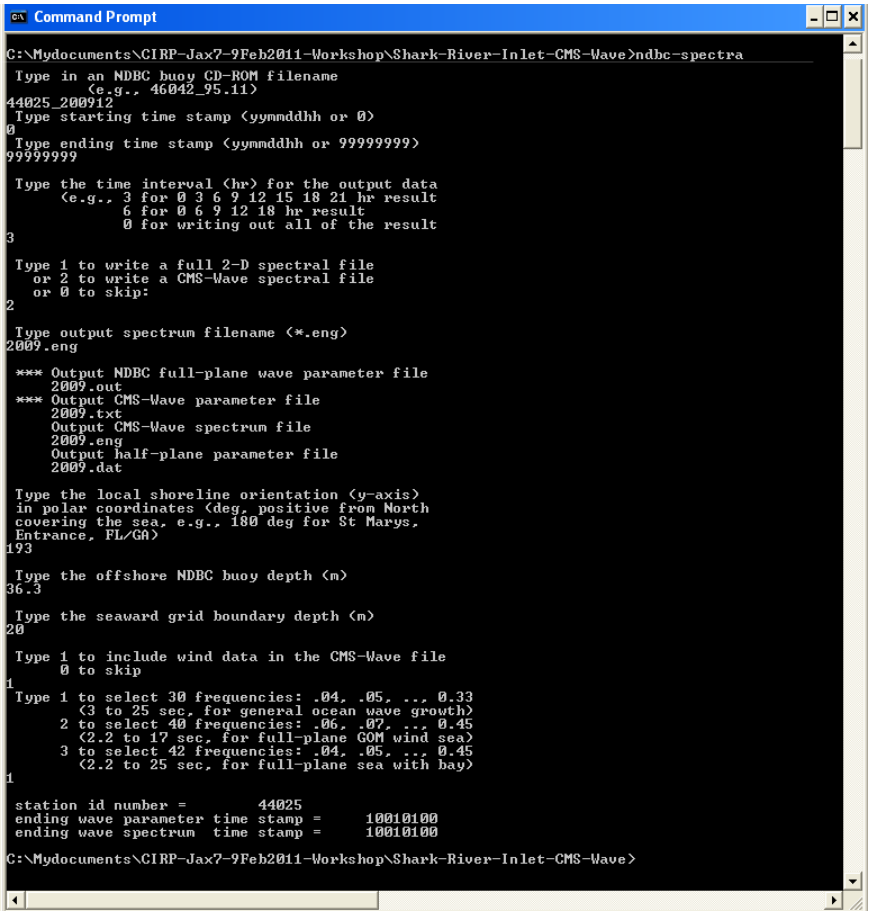


Figure G- 5. Run ndbc-spectra.exe in DOS

- CDIP buoy data - run **cdip-spectra.exe** (also FORTRAN code) to read the CDIP standard directional wave file and generate the CMS-Wave input *.eng file. Download the CDIP wave file from http://cdip.ucsd.edu/?nav=historic&sub=data (e.g., sp154-200912) – see Figure G- 6 to Figure G- 8.

  Run **cdip-spectra.exe** in the DOS window similar to **ndbc-spectra.exe** – see Figure G- 9. Because CDIP spectral file already contains the buoy location depth information, **cdip-spectra.exe** will not prompt for this depth input. For processing either NDBC or CDIP data, users shall check and manually fill any data gaps in *.eng and *.txt files (using the first available spectral data from the neighboring time interval).

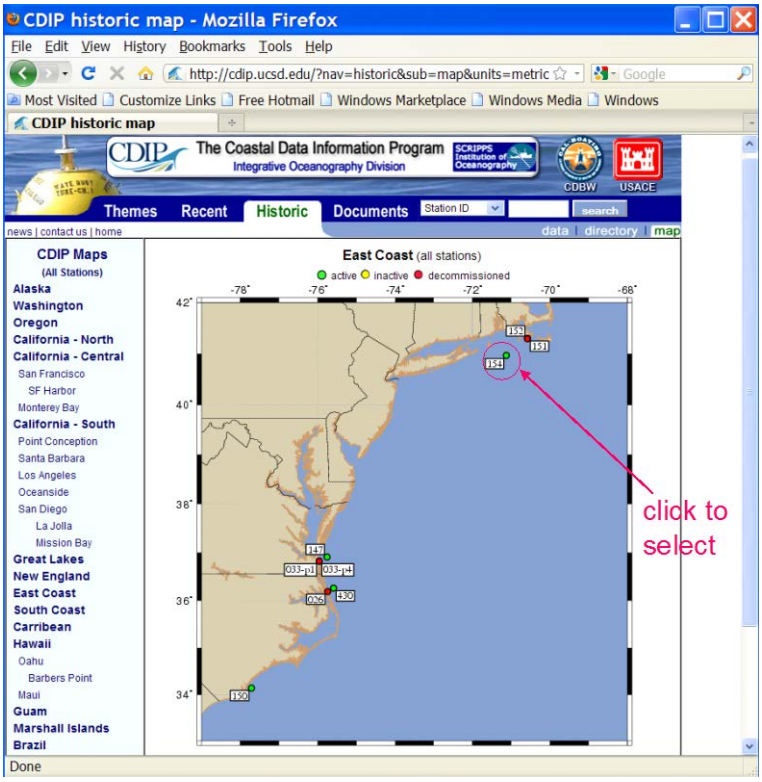


Figure G- 6. CDIP buoy data access web page.

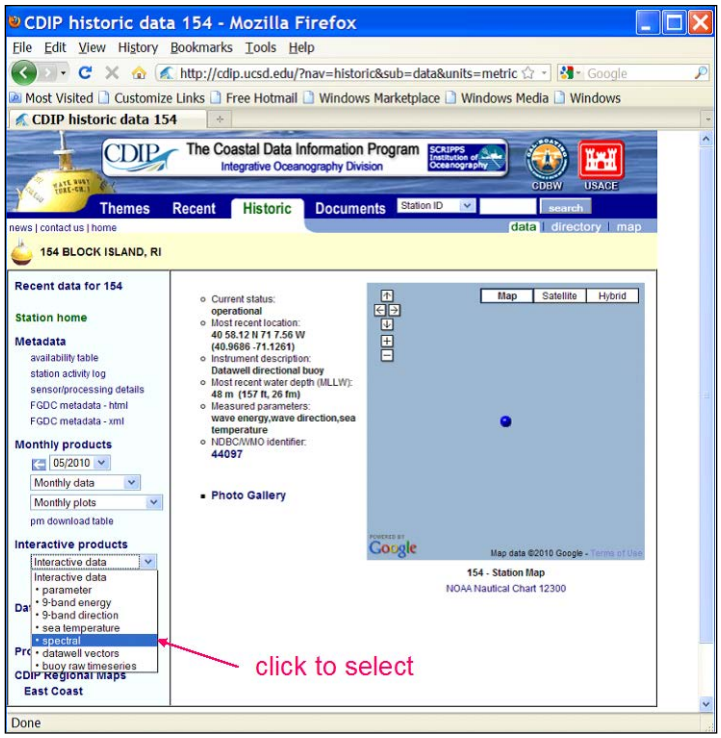


Figure G- 7. CDIP buoy data access web page.

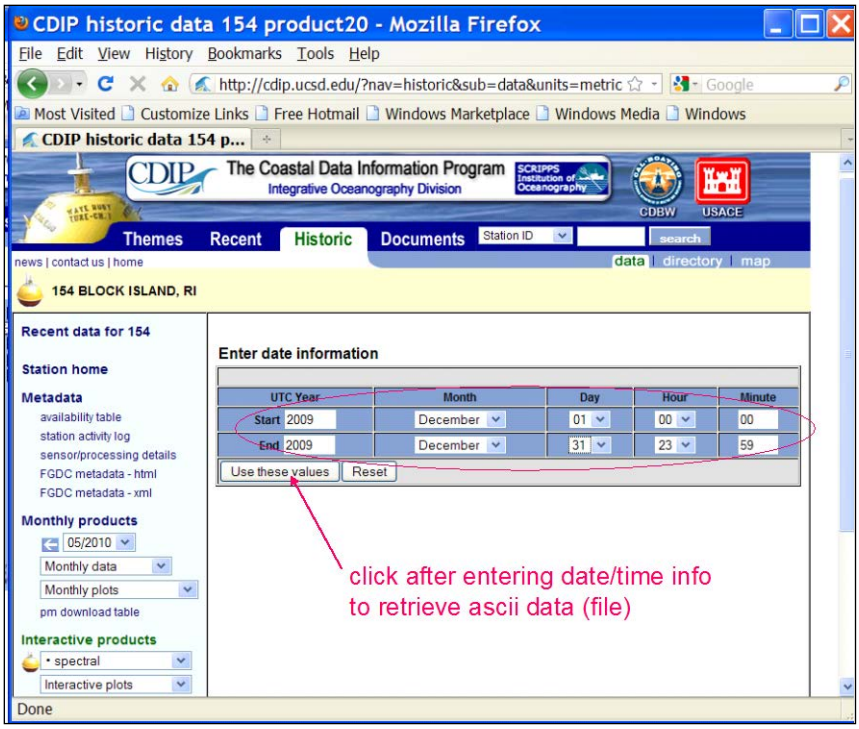


Figure G- 8. CDIP buoy spectral data download web page.

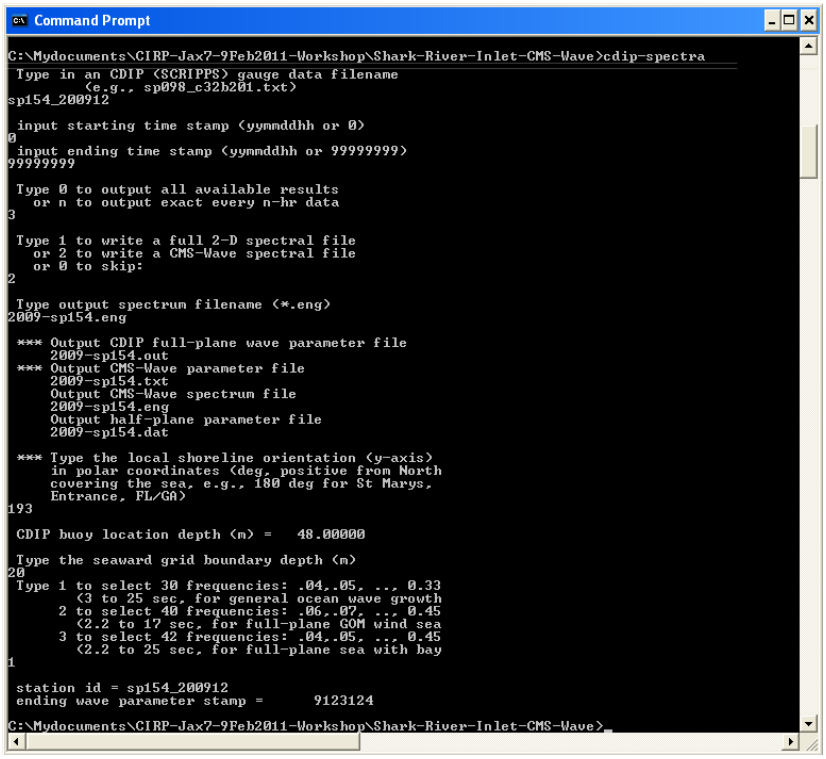


Figure G- 9. Run cdip-spectra.exe in DOS.

20 Appendix I: UTM and State Plane Zones

The main purpose of the State Plane coordinate system was to designe a conformal mapping system which minizes local distortion to less than 1/10,000. In order to achieve this accuracy, some states were divided into smaller zones called FIPS (Federal Information Processing Standard) zone. The table below lists the US zone names FIPS zone numbers, and UTM zones

Zone Name	FIPS Zone	UTM Zone
ALASKA ZONE 1	5001	8 & 9
ALASKA ZONE 2	5002	7
ALASKA ZONE 3	5003	6
ALASKA ZONE 4	5004	5 & 6
ALASKA ZONE 5	5005	5
ALASKA ZONE 6	5006	4
ALASKA ZONE 7	5007	3 & 4
ALASKA ZONE 8	5008	3
ALASKA ZONE 9	5009	
ALASKA ZONE 10	5010	1 & 4
ALABAMA EAST ZONE	101	16
ALABAMA WEST ZONE	102	16
ARIZONA EAST ZONE	201	12
ARIZONA CENTRAL ZONE	202	12
ARIZONA WEST ZONE	203	12
ARKANSAS NORTH ZONE	301	15
ARKANSAS SOUTH ZONE	302	15
CALIFORNIA ZONE I	401	10
CALIFORNIA ZONE II	402	10 & 11
CALIFORNIA ZONE III	403	10 & 11
CALIFORNIA ZONE IV	404	10 & 11
CALIFORNIA ZONE V	405	10 & 11
CALIFORNIA ZONE VI	406	11
CALIFORNIA ZONE VII	407	
COLORADO NORTH ZONE	501	12 & 13
COLORADO CENTRAL ZONE	502	12 & 13
COLORADO SOUTH ZONE	503	12 & 13
CONNECTICUT	600	18 & 19

DELAWARE	700	18
FLORIDA EAST ZONE	901	17
FLORIDA WEST ZONE	902	17
FLORIDA NORTH ZONE	903	16 & 17
GEORGIA EAST ZONE	1001	17
GEORGIA WEST ZONE	1002	16 & 17
HAWAII ZONE 1	5101	5
HAWAII ZONE 2	5102	4
HAWAII ZONE 3	5103	4
HAWAII ZONE 4	5104	4
HAWAII ZONE 5	5105	4
IDAHO EAST ZONE	1101	12
IDAHO CENTRAL ZONE	1102	11 & 12
IDAHO WEST ZONE	1103	11
ILLINOIS EAST ZONE	1201	16
ILLINOIS WEST ZONE	1202	15 & 16
INDIANA EAST ZONE	1301	16
INDIANA WEST ZONE	1302	16
IOWA NORTH ZONE	1401	14 & 15
IOWA SOUTH ZONE	1402	15
KANSAS NORTH ZONE	1501	14 & 15
KANSAS SOUTH ZONE	1502	14 & 15
KENTUCKY NORTH ZONE	1601	16 & 17
KENTUCKY SOUTH ZONE	1602	16 & 17
LOUISIANA NORTH ZONE	1701	15
LOUISIANA SOUTH ZONE	1702	15 & 16
MAINE EAST ZONE	1801	19
MAINE WEST ZONE	1802	19
MARYLAND	1900	17 & 18
MASSACHUSETTS MAINLAND	2001	18 & 19
MASSACHUSETTS ISLAND	2002	19
MICHIGAN NORTH ZONE	2111	16
MICHIGAN CENTRAL ZONE	2112	16 & 17
MICHIGAN SOUTH ZONE	2113	16 & 17
MINNESOTA NORTH ZONE	2201	14 & 15
MINNESOTA CENTRAL ZONE	2202	14 & 15
MINNESOTA SOUTH ZONE	2203	14 & 15
MISSISSIPPI EAST ZONE	2301	16
MISSISSIPPI WEST ZONE	2302	15 & 16
MISSOURI EAST ZONE	2401	15 & 16
MISSOURI CENTRAL ZONE	2402	15

MISSOURI WEST ZONE	2403	15
MONTANA	2500	11 & 12 & 13
MONTANA NORTH ZONE (NAD27)	2501	11 & 12 & 13
MONTANA CENTRAL ZONE (NAD27)	2502	11 & 12 & 13
MONTANA SOUTH ZONE (NAD27)	2503	11 & 12 & 13
NEBRASKA	2600	13 & 14 & 15
NEBRASKA NORTH ZONE (NAD27)	2601	13 & 14 & 15
NEBRASKA SOUTH ZONE (NAD27)	2602	13 & 14 & 15
NEVADA EAST ZONE	2701	11
NEVADA CENTRAL ZONE	2702	11
NEVADA WEST ZONE	2703	11
NEW HAMPSHIRE	2800	18 & 19
NEW JERSEY	2900	18
NEW MEXICO EAST ZONE	3001	13
NEW MEXICO CENTRAL ZONE	3002	13
NEW MEXICO WEST ZONE	3003	12 & 13
NEW YORK EAST ZONE	3101	18
NEW YORK CENTRAL ZONE	3102	18
NEW YORK WEST ZONE	3103	17 & 18
NEW YORK LONG ISLAND ZONE	3104	18
NORTH DAKOTA NORTH ZONE	3301	13 & 14
NORTH DAKOTA SOUTH ZONE	3302	13 & 14
OHIO NORTH ZONE	3401	16 & 17
OHIO SOUTH ZONE	3402	16 & 17
OKLAHOMA NORTH ZONE	3501	13 & 14 & 15
OKLAHOMA SOUTH ZONE	3502	14 & 15
OREGON NORTH ZONE	3601	10 & 11
OREGON SOUTH ZONE	3602	10 & 11
PENNSYLVANIA NORTH ZONE	3701	17 & 18
PENNSYLVANIA SOUTH ZONE	3702	17 & 18
RHODE ISLAND	3800	19
SOUTH CAROLINA	3900	17
SOUTH CAROLINA NORTH ZONE (NAD27)	3901	17
SOUTH CAROLINA SOUTH ZONE (NAD27)	3902	17
SOUTH DAKOTA NORTH ZONE	4001	13 & 14
SOUTH DAKOTA SOUTH ZONE	4002	13 & 14
TENNESSEE	4100	16 & 17
TEXAS NORTH ZONE	4201	13 & 14
TEXAS NORTH CENTRAL ZONE	4202	13 & 14 & 15
TEXAS CENTRAL ZONE	4203	13 & 14 & 15
TEXAS SOUTH CENTRAL ZONE	4204	13 & 14 & 15

UTAH NORTH ZONE	4301	12
UTAH CENTRAL ZONE	4302	12
UTAH SOUTH ZONE	4303	12
VERMONT	4400	18 & 19
VIRGINIA NORTH ZONE	4501	17 & 18
VIRGINIA SOUTH ZONE	4502	17 & 18
WASHINGTON NORTH ZONE	4601	10 & 11
WASHINGTON SOUTH ZONE	4602	10 & 11
WEST VIRGINIA NORTH ZONE	4701	17 & 18
WEST VIRGINIA SOUTH ZONE	4702	17
WISCONSIN NORTH ZONE	4801	15 & 16
WISCONSIN CENTRAL ZONE	4802	15 & 16
WISCONSIN SOUTH ZONE	4803	15 & 16
WYOMING EAST ZONE	4901	13
WYOMING EAST CENTRAL ZONE	4902	13
WYOMING WEST CENTRAL ZONE	4903	12
WYOMING WEST ZONE	4904	12

A description of the State Plane zones including maps can be viewed at http://www.ems-i.com/smshelp/General_Tools/Coordinates/StatePlane_system.htm.

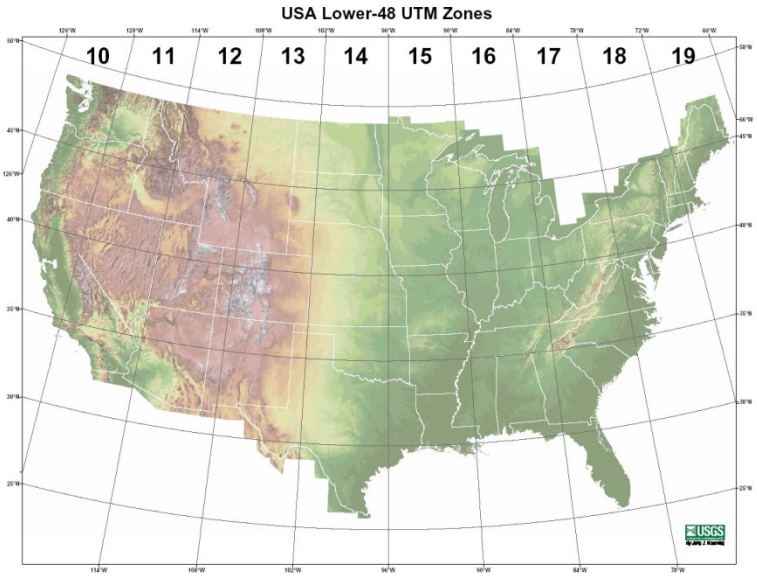


Figure 20-1. UTM zones for the Lower 48 states (from USGS).

21 Appendix J: Glossary

A

Abrasion: The mechanical wearing away by rock material transported by wind or water

Active Bed: The portion of the bed sediment that is being transported.

Active Layer: The top surface layer of sediment in which sediments are subject to exchange with those moving with the flow (entrainment or deposition).

Accretion: The accumulation of (beach) sediment, deposited by natural fluid flow processes.

Advection: Physical process by means which a substance (i.e. sediment, salinity) or conserved property (energy or enthalpy) is transported by a fluids bulk motion.

Aggradation: The geologic process by which various parts of the surface of the earth are raised in elevation or built up by the deposition of material transported by water or wind.

Amplitude: Half of the peak-to-trough range (or height) of a wave.

Angle of Repose: Angle between the horizontal and the maximum slope that a soil assumes through natural processes.

Armoring: The formation of a resistant layer of relatively large particles resulting from removal of finer particles by erosion.

B

Bathymetry: The measurement of depths of water in oceans, seas and lakes; also the information derived from such measurements.

Bay: A recess or inlet in the shore of a sea or lake between two capes or headlands, not as large as a gulf but larger than a cove.

Bed: The bottom of a watercourse, or any body of water.

Bed load: Sediment transport mode in which individual particles either roll or slide along the bed as a shallow, mobile layer a few particles in height above the bed.

Bedform: Any deviation from a flat bed that is readily detectable by eye and higher than the largest sediment size present in the parent bed material; generated on the bed of an alluvial channel by the flow.

Breaker Index: Maximum ratio of wave height to water depth in the surf zone, typically 0.78 for spilling waves, ranging from about 0.6 to 1.5.

Bulk Density: Bed density equal to the total mass sand and water per unit volume of the bed.

Bypasssing Sand: Hydraulic or mechanical movement of sand from the accreting updrift side to the eroding downdrift side of an inlet or harbor entrance. The hydraulic movement may include natural as well as movement caused by man.

C

Channel: (1) A natural or artificial waterway of perceptible extent which either periodically or continuously contains moving water, or which forms a connecting link between two bodies of water. (2) The part of a body of water deep enough to be used for navigation through an area otherwise too shallow for navigation. (3) The deepest portion of a stream, bay, or strait through which the main volume of current of water flows. (4) An open conduit for water either naturally or artificially created, but does not include artificially created irrigation, return flow or stockwatering channels.

Chart Datum: The plane or level to which soundings, tidal levels or water depths are referenced, usually low water datum. See also Datum.

Constituent Tide: See Tidal Constituent.

Convection: Physical process by means which a substance (i.e. sediment, salinity) or conserved property (energy or enthalpy) is transported by advection and diffusion.

Conveyance: A measure of the flow carrying capacity of a channel section. Flow is directly proportional to conveyance for steady flow. From Manning's equation, the proportionality factor is the square root of the energy slope.

Coordinate System: A set of rules for specifying how coordinates are to be assigned to points.

Coriolis Force: Force or Pseudo-force due to the Earth's rotationwhich causes moving bodies to be deflected to the right in the Northern Hemisphere and to the left in the Southern Hemisphere. The force is proportional to the speed and latitude of the moving object. It is zero at the equator and maximum at the poles.

Cross-shore: Perpendicular to the shoreline.

D

Datum: A datum is a point, line or surface used as a reference as in surveying, mapping, geology or numerical modeling.

Depth: Vertical distance from still-water level (or datum as specified) to the bottom.

Dry Density: Bed density equal to the mass of (dry) sand per unit volume of the bed.

E

Ebb: Period when tide level is falling; often taken to mean the ebb current which occurs during this period.

Elevation: The distance of a point above a specified surface of constant potential; the distance is measured along the direction of gravity between the point and the surface.

Equilibrium Argument: The theoretical phase of the equilibrium tide. It is usually represented by the expression (V + u), in which V is a uniformly changing angular quantity involving multiples of the hour angle of the mean Sun, the mean longitudes of the Moon and Sun, and the mean longitude of lunar or solar perigee; and u is a slowly changing angle depending upon the longitude of the Moon's node. When pertaining to an initial instant of time, such as the beginning of a series of observations, it is expressed by (V0 + u).

Equilibrium Tide: Hypothetical tide due to the tide-producing forces under the equilibrium theory. Also known as gravitational tide.

F

Fetch length: (1) The horizontal distance (in the direction of the wind) over which a wind generates waves or creates wind setup. (2) The horizontal distance along open water over which the wind blows and generates waves.

G

Gradation: (ASTM D 653) The proportions by mass of a soil or fragmented rock distributed in specified particle-size ranges.

Graded Bed: An arrangement of particle sizes within a single bed, with coarse grains at the bottom of the bed and progressively finer grains toward the top of the bed.

Grain Density: See Sediment Density.

Greenwich Mean Time or **GMT**: Mean solar time at the meridian of Greenwich, England. It has been used as a basis for standard time throughout the world. Also called Zulu time. (See also Coordinated Universal Time or UTC).

Grid: Network of points covering the space or time-space domain of a numerical model. The points may be regularly or irregularly spaced.

H

Harmonic Analysis: The mathematical process by which the observed tide or tidal current at any place is separated into basic harmonic constituents.

Hydrodynamic: Relates to the specific scientific principles that deal with the motion of fluids and the forces acting on solid bodies immersed in fluids, and in motion relative to them.

I

Infragravity wave: Long waves with periods of 30 seconds to several minutes.

Initial Conditions: The values of water levels, velocities, concentrations, etc., that are specified everywhere in the grid or mesh at the beginning of a model run. For iterative solutions, the initial conditions represent the first estimate of the variables the model is trying to compute.

Interpolation: Estimation of an intermediate value of one variable (dependent) as a function of a second variable (independent) when values of the dependent variable corresponding to several discrete values of the independent variable are known.

J

Jetty: (1) On open seacoasts, a structure extending into a body of water, and designed to prevent shoaling of a channel by littoral materials, and to direct and confine the stream or tidal flow. Jetties are built at the mouth of a river or tidal inlet to help deepen and stabilize a channel. (2) A shore-perpendicular structure built to stabilize an inlet and prevent the inlet channel from filling with sediment.

K

Kinematic Viscosity: The dynamic viscosity divided by the fluid density.

L

Longshore: Parallel to and near the shoreline; Alongshore.

LHW: Lower High Water.

LLW: Lower Low water

M

MHHW: Mean Higher High Water. The average of the higher high water height of each tidal day observed over the National Tidal Datum Epoch.

MHW: Mean High Water. The average of all the high water heights observed over the National Tidal Datum Epoch.

MSL: Mean Sea Level. The arithmetic mean of hourly heights observed over the National Tidal Datum Epoch.

MLW: Mean Low Water. The average of all the low water heights observed over the National Tidal Datum Epoch.

MLLW: Mean Lower Low Water. The average of the lower low water height of each tidal day observed over the National Tidal Datum Epoch

N

National Geodetic Vertical Datum: The vertical-control datum used by the National Geodetic Survey since 1980. This definition is considered to be misleading because it implies the adoption of an NGVD as adjusted in 1980. Accordingly, this expression should be disregarded. Consultation with the National Geodetic Survey (N/CG172, NOAA/NOS) resulted in the confirmation that NGVD of 1929 still applies. (See also National Geodetic Vertical Datum of 1929.)

Nodal Factor: A factor depending upon the longitude of the Moon's node which, when applied to the mean coefficient of a tidal constituent, will adapt the same to a particular year for which predictions are to be made.

O

Overdepth: (1) (EM 1110-2-1003) Additional depth below the required section (or template) specified in a dredging contract. This additional depth is permitted (but not required) because of inaccuracies in the dredging process. (2) The distance between the theoretical maintenance depth and the actual dredging depth. (3) The amount of extra dredging depth that is allowed and paid for, if dredged, in excess of prescribed contract depth

P

Particle Size: A linear dimension, usually designated as "diameter," used to characterize the size of a particle. The dimension may be determined by any of several different techniques, including sedimentation sieving, micrometric measurement, or direct measurement.

Phi Grade Scale: A logarithmic transformation of the Wentworth grade scale for size classifications of sediment grains based on the negative logarithm to the base 2 of the particle diameter: = -log2d.

Porosity: Percentage of the total volume of a soil sample not occupied by solid particles but by air and water, n = Vv/VT × 100. (ASTM D 653) the ratio, usually expressed as a percentage, of (1) the volume of voids of a given soil or rock mass, to (2) the total volume of the soil or rock mass.

Progradation: The building forward or outward toward the sea of a shoreline or coastline (as with a beach, delta, or fan) by nearshore deposition of river-borne sediments or by continuous accumulation of beach material thrown up by waves or moved by longshore drifting.

Q

Quality Assurance or **QA**: The total integrated program for assuring the reliability of data. A system for integrating the quality planning, quality control, quality assessment, and quality improvement efforts to meet user requirements and defined standards of quality with a stated level of confidence.

Quality Control or **QA**: The overall system of technical activities for obtaining prescribed standards of performance in the monitoring and measurement process to meet user requirements.

R

Rectification: The process of producing, from a tilted or oblique photograph, a photograph from which displacement caused by tilt has been removed.

Reconnaissance Survey: (EM 1110-2-1003) A general minimum-effort survey performed to determine the general project or channel conditions. The surveys may be controlled or uncontrolled. The survey may serve as a before dredging survey.

Reflection of Water Waves: The process by which the energy of the wave is returned seaward.

Refraction of Water Waves: (1) The process by which the direction of a wave moving in shallow water at an angle to the contours is changed. The part of the wave advancing in shallower water moves more slowly than that part still advancing in deeper water, causing the wave crest to bend toward alignment with the underwater contours. (2) The bending of wave crests by currents.

S

Salinity: Number of grams of salt per thousand grams of sea water, usually expressed in parts per thousand (symbol: ‰) or in Practical Salinity Units (PSU).

Scatter Plot: A two-dimensional plot showing the joint distribution of two variables within a data sample.

Sediment Density: Density of the sediment grains or solids per unit volume of the solids. Also known as grain density.

Sediment Transport Formula: A formula or algorithm for calculating sediment transport rate given the hydraulics and bed material characteristics at a point or cross section.

Sediment Transport Paths: The routes along which net sediment movement occurs.

Spinup Period: The time it takes for the simulation errors due to the initial condition to disappear.

T

Tidal Datum: A tidal datum is a standard elevation defined by a certain phase of the tide. Tidal datums are used as references to measure local water levels.

Tidal Constituent: a harmonic element consisting of an amplitude, frequency, and phase, representing a periodic variation in the corresponding tide or tidal currents due to the relative positions of the Earth, Moon, and Sun. Also known as Constituent Tide.

U

Universal Time Coordinated (UTC): Coordinates time scale maintained by the Bureau International des Poids et Measures (BIPM). Generally equivalent to Greenwich Mean Time (GMT). Also referred to simply as Universal Time, Z Time, or Zulu Time.

Universal Transverse Mercator (UTM): A worldwide metric military coordinate system rarely used for civil works applications (EM 1110-2-1003).

V

Validation: In computer modeling and simulation, validation is the process of determining the degree or accuracy to which a model or simulation is an accurate representation of a real world process from the perspective of the intended uses of the model or simulation.

Verification: In computer modeling and simulation, verification is the process of determining the accuracy of which the governing equations of a specific model or simulation are being solved.

W

Wash Load: Part of the suspended load with particle sizes smaller than found in the bed; it is in near-permanent suspension and transported without deposition; the amount of wash load transported through a reach does not depend on the transport capacity of the flow; the load is expressed in mass or volume per unit of time.

Water Surface Elevation: A measure of the free water surface with respect to a given datum. See also - Water Elevation.

Weir Jetty: An updrift jetty with a low section or weir over which littoral drift moves into a predredged deposition basin which is then dredged periodically.

Winnow: Natural removal of sediments through suspension and erosion of fine particles by water flow.

X

Y

Z