

Preface

Asmita is a Matlab™ App to examine the long-term morphological response of tidal inlets and estuaries. Rather than modelling the flows and sediment transport in detail, the model considers the sediment exchange between morphological elements (channel, tidal flat, etc) and determines the exchanges needed based on the forcing conditions and a specified set of equilibrium conditions for each type of element. The model output provides the change in element volume over time and can also examine changes in tidal prism, biomass production (if saltmarshes are included)

Requirements

The model is written in Matlab™ and provided as Open Source code (issued under a GNU General Public License) and runs under v2018b or later. Asmita uses the *multoolbox* and *dstoolbox*.

Routing within the model makes use of graph and directed graph functions which were introduced in Matlab 2015b. There are however several versions of these functions available from the Mathworks File Exchange if there is a need to use an earlier version of Matlab™.

Resources

The Asmita App and two toolboxes (*multoolbox* and *dstoolbox*) can be downloaded from www.coastalsea.uk.

Cite as:

Townend, I.H., 2021, Asmita manual, CoastalSEA, UK, pp69, www.coastalsea.uk.

Bibliography

Key papers setting out the theoretical background to the model include:

Stive, M.J.F., Capobianco, M., Wang, Z.B., Ruol, P., Buijsman, M.C., 1998. Morphodynamics of a tidal lagoon and adjacent coast. In: J. Dronkers, M.B.A.M. Scheffers (Eds.). *Physics of Estuaries and Coastal Seas: 8th International Biennial Conference on Physics of Estuaries and Coastal Seas*, 1996. A Balkema, pp. 397-407.

van Goor, M.A., Zitman, T.J., Wang, Z.B., Stive, M.J.F., 2003. Impact of sea-level rise on the morphological equilibrium state of tidal inlets. *Marine Geology*, 202, 211-227.

Kragtwijk, N.G., Stive, M.J.F., Wang, Z.B., Zitman, T.J., 2004. Morphological response of tidal basins to human interventions. *Coastal Engineering*, 51, 207-221.

Townend, I.H., Wang, Z.B., Stive, M.J.E., Zhou, Z., 2016a. Development and extension of an aggregated scale model: Part 1 – Background to ASMITA. *China Ocean Engineering*, 30(4), 482-504.

Townend, I.H., Wang, Z.B., Stive, M.J.E., Zhou, Z., 2016b. Development and extension of an aggregated scale model: Part 2 – Extensions to ASMITA. *China Ocean Engineering*, 30(5), 651-670.

Further sources of information are detailed in the References towards the end of the Manual.

Acknowledgements

The concept of an **A**ggregated **S**cale **M**orphological **I**nteraction between **T**idal basin and **A**djacent coast (ASMITA) was first proposed by Marcel Stive and co-workers at TU Delft. The first Matlab version of ASMITA was developed by Tjerk Zitman at TU Delft. Work by Zheng Bing Wang and colleagues also at TU Delft further developed the model and set out the basis of model parameterisation. As part of the Estuaries Research Program Phase 2 (ERP2), funded by the Environment Agency and Department for Environment, Food and Rural Affairs (part of project FD2107) this research code was documented, tested and a manual prepared by Ian Townend and Adrian Wright at ABPmer. Version 1.3 was released as Open Source software available as part of the

Estuary Guide web site (www.estuary-guide.net). Further developments were undertaken to add additional elements and processes by Ian Townend, Jeremy Spearman, Kate Rossington and Michiel Knappen at HR Wallingford. This manual is for a version developed by Ian Townend at CoastalSEA in 2016 and ported to install as a Matlab™ App using the *multitoolbox* in 2021.

Revision history

Version	Date	Changes
3.5	Jan 2024	Added additional checks on model stability. Revised conditions for limiting erosion and fixing interventions. Improved stability when area or vertical exchange goes to zero (e.g., to simulate closure and opening of marshes). Updated function for simple_tide now used. Utility to update models to latest version of code added. Tool to plot multiple elements from PlotUI added. Added matrix condition checks and option to display a summary plot during run time. Added check that toolboxes are installed. Implemented checks when using Response tab.
3.4	Sept 2023	Interventions can now be set to be erodible or non-erodible. Equilibrium for beach type elements can now use drift rate as an alternative to prism or tidal range to define the equilibrium volume. Added option to plot differences of variable as a Scaling option. Water level offset for tidal range cycles modified to ensure variation gives correct mean tide. Saltmarsh algorithm improved to be stable when switching to zero depth and to bare flat depth and find the correct root for the equilibrium depth. Bug fixes for (i) time series drift from a single source, (ii) advection tabs used before models runs and (iii) handling of data time and duration options (duration used if start time =0). Table formatting with large numerical values improved. Pan and zoom button handling improved.
3.3	July 2023	Added functions and scripts to run Asmita from the command line or in batch mode. Added Unit testing for set of models that test the range of functionality available. Added option to Time plots to use a pivot year to define the start point (y-value and time) for all lines included on plot. Modified handling of saltmarsh equilibrium when water depth is zero and removed scaling to initial values for saltmarsh and storage elements. Added option to use time-varying horizontal and vertical exchanges. Added moving surface area to output variables (to view interventions).
3.2	June 2023	Added Help options for Documentation (online), Manual and Papers that detail theoretical background.
3.1	Oct 2022	Added option to examine sea level rise over the Holocene (i.e., for periods prior to 0 A.D) see Section 6.2. Various error traps added to improve editing and modifying of model components. Error in reloading past cases corrected.
3.0	Oct 2021	ASMITA ported to run as an App using the modelui toolbox. Added option to include along channel changes in water level due to river-tide interactions using the CSTmodel. Graphics modified to make use of new interface.



2.0	Nov 2020	ASMITA OO ported to work as model ASMITA within the ModelUI framework.
1.5	Oct.2020	Added inlet model. See Section 3.5. Added option to define element specific equilibrium volume using tidal range rather than tidal prism. See Sections 5.4 and 6.1. Updated Unit Testing tools.
1.4	Nov.2016	Added tidal damping for use in multi-channel element models Added pan and zoom options to graph tabs Fixed bug in upstream prism when channels are in random order Element Types extended to allow deltas and beach/spit complexes to be modelled. See Section 5.1. Added 'Animate' tab to plot GUI, to provide animations of how some variable changes across the network. Option to load interventions from a file. See Section 4.4 Option to load element parameters from a file. See Section 4.4 Option to include timeseries definition of river and drift sources. See Sections 4.4 and 5.3. Deletion of River and Drift input sources corrected.
1.3	Sep.2016	First release to accompany papers published in China Ocean Engineering



Contents

1	Introduction	1
2	Getting started	2
2.1	Configuration.....	2
2.1.1	Installing the toolboxes	2
2.1.2	Installing the App	2
2.2	Opening ASMITA.....	2
2.3	Typical workflow to set-up new model.....	3
3	Sample models	5
3.1	Humber Model (simple 3 element model: delta-channel-flat)	5
3.2	Venice model (multiple inlets and saltmarsh)	8
3.3	Ameland model (mixed sediments – experimental code)	9
3.4	Severn model (tidal pumping).....	10
3.5	Inlet model.....	10
4	Application Menus	12
4.1	File.....	12
4.2	Tools.....	12
4.3	Project.....	12
4.4	Setup.....	13
4.4.1	Estuary.....	13
4.4.2	Element.....	15
4.4.3	Saltmarsh.....	17
4.4.4	River advection.....	19
4.4.5	Drift advection.....	20
4.4.6	Interventions	21
4.4.7	Hydraulics	22
4.4.8	Run parameters.....	22
4.5	Run	24
4.6	Analysis.....	25
4.6.1	Plotting	25
4.6.2	Statistics.....	27
4.7	Help	30
4.8	Tabs	30
4.9	UI Data Selection	31
4.10	Model outputs.....	33
4.11	Interpretation when using water and sediment volumes ($\pm n$).....	33
4.11.1	Elements using water volumes	34



4.11.2	Elements using sediment volumes.....	34
4.11.3	All Elements output.....	34
4.12	Accessing data from the Command Window	35
5	Supporting Information for Model Setup	36
5.1	Model elements	36
5.1.1	File format for Element Parameters	38
5.2	Dispersion.....	38
5.2.1	Exchange between channel elements (and channel and delta)	38
5.2.2	Exchange between channel and flats	38
5.3	Advection	39
5.3.1	River timeseries file format.....	40
5.3.2	Drift timeseries file format	40
5.4	Specification of Equilibrium	41
5.5	Interventions.....	42
5.5.1	File format for loading interventions.....	43
5.5.2	Fixed Elements and Interventions	43
5.5.3	Managed realignment, reclamation and storage schemes.....	45
5.6	Saltmarsh.....	46
5.7	Storage.....	47
5.8	Hydraulics	47
5.9	Tidal Pumping	47
5.10	Morphological Response.....	48
5.11	Derive Output.....	48
5.11.1	Calling an external function	49
5.11.2	Pre-defined functions	49
6	User functions.....	52
6.1	User Prism Coefficients function – userprismcoeffs.m.....	52
6.2	UserSLRrate	52
6.3	Command Line option.....	52
6.4	Asmita_model and AsmitaModel.....	54
7	Program Structure.....	55
7.1	Classes and functions	55
7.2	Unit Testing.....	57
8	Bibliography.....	59
	Appendix A – Function call sequence in AsmitaModel.....	61
	Appendix B – Erratum	62
	Appendix C – Matrix Condition Checks	63



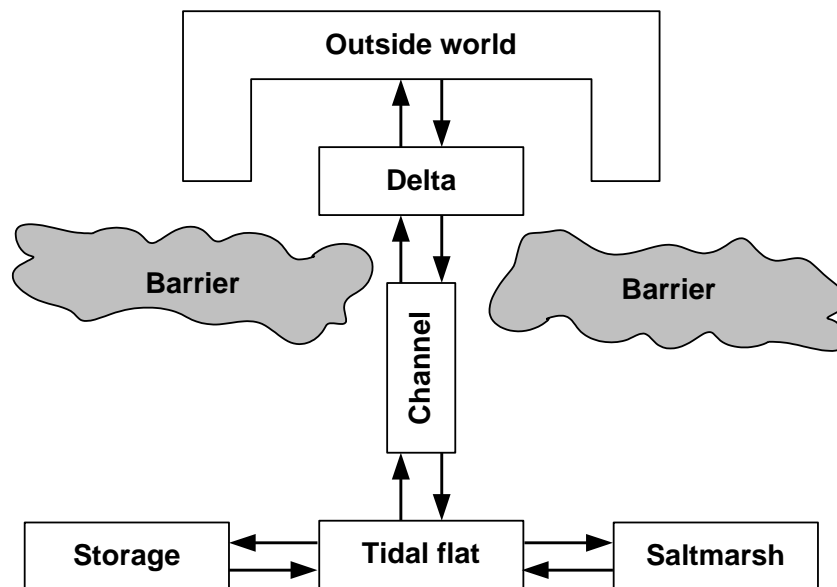
1 Introduction

Aggregated **S**cale **M**orphological **I**nteraction between a **T**idal inlet and the **A**djacent coast (**ASMITA**).

This manual has been written as a guide to using the Matlab ASMITA model and illustrates the concepts and methodology used in long-term morphological modelling.

The idea behind ASMITA is that a tidal basin, or estuary can be sub-dividing into several elements (Figure 1). Each element has an equilibrium condition when unperturbed and following some form of perturbation (e.g., sea level rise, addition or removal of volume or area) the model seeks a new equilibrium. For example, if under initial conditions the accommodation space is zero (no net change in the transport of sediment), with increased mean sea level the accommodation space becomes positive and sediment may be imported, the opposite is true when sea level falls.

Figure 1 - Schematisation of an estuary or inlet into a number of elements as used in ASMITA



The key concepts of the ASMITA model are:

- The estuary is schematized into a number of geomorphologic elements;
- The state of each element is described by its volume and plan area (water or sediment);
- Integrated parameters of hydrodynamics (tidal prism, or tidal range) are used;
- Empirical relationships define the morphological equilibrium for each element;
- Deviation from the morphological equilibrium causes sediment demand; and
- A gradient in sediment demand drives sediment transport, and thereby morphological change.

The theoretical background to the model is explained in various published papers as detailed in the Bibliography in the Preface and in the References.

2 Getting started

2.1 Configuration

Asmita is installed as an App and requires muitoolbox and dstoolbox to be installed. The download for each of these includes the code, documentation and example files. The files required are:

dstoolbox: `dstoolbox.mltbx`

muitoolbox: `muitoolbox.mltbx`

The App file: `Asmita.mlappinstall`

2.1.1 Installing the toolboxes

The two toolboxes can be installed using the *Add-Ons > Manage Add-Ons* option on the Home tab of Matlab™. Alternatively, right-click the mouse on the ‘mltbx’ files and select install. All the folder paths are initialised upon installation and the location of the code is also handled by Matlab™. The location of the code can be accessed using the options in the *Manage Add-Ons* UI.

2.1.2 Installing the App

The App is installed using the Install Apps button on the APPS tab in Matlab™. Alternatively, right-click the mouse on the ‘mlappinstall’ file and select install. Again, all the folder paths are initialised upon installation and the location of the code is handled by Matlab™.

Once installed, the model can be run from the APPS tab. This sets the App environment paths, after which the App can be run from the Command Window using:

```
>> Asmita;
```

The App environment paths can be saved using the Set Path option on the Matlab™ Home tab.

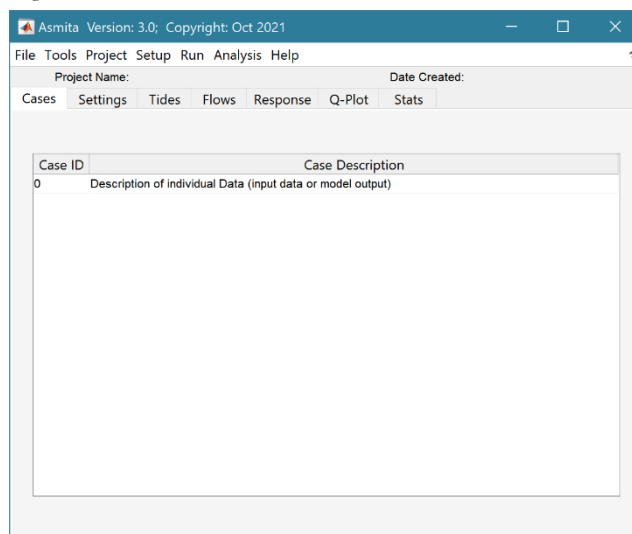
Documentation can be viewed from App Help menu, or the Supplemental Software in the Matlab™ documentation. The location of the code can be accessed by hovering over the App icon and then finding the link in the pop-up window.

2.2 Opening ASMITA

A graphical user interface (GUI) is used to set-up, run scenarios, plot results and export model output.

A splash screen crediting the developers appears for a few seconds before being replaced by the ASMITA interface (Figure 2).

Figure 2 - The ASMITA GUI



Version number

Drop-down menus

Project title and date

Display Tabs

Description of model runs (Cases)

2.3 Typical workflow to set-up new model

The following steps explain how to set-up a model using the various tools provided. Text in *Red italic* refers to drop down menus. Text in *Green italic* refers to Tab titles.

File>New: define project name and date.

Setup>Estuary>System Parameters: define equilibrium concentration.

Setup>Estuary>Water Levels: define tidal range, slr and any variation in tidal range (e.g nodal tide).

Setup>Element>Define Elements: specify how many elements of each type are needed.

Setup>Element>Element Parameters: define volume, area etc for each element.

Setup>Run Parameters>Time Step: define time step and length of run.

Setup>Run Parameters>Equilibrium Coefficients: select equilibrium prism coefficient set from list (coefficients are defined in userprismcoeffs.m – see Section 6.1 for further details).

Setup>Estuary>Dispersion: define the horizontal exchange between elements. The direction from-to working landwards from the most seaward element is important as this is used to calculate the tidal prism in each reach.

River flow, Interventions, Littoral Drift, Saltmarsh, Tidal Pumping and various constraints can be added as required. However, the above is the minimum required for the model to run.

To examine what has been set-up the Tabs provide a summary of what is currently defined. Note: these only update when clicked on using a mouse and the values cannot be edited from the Tabs. A full description of the Tabs is given in Section 4.8.

Cases: lists the cases that have been run with a case id and description.

Settings tab details the various model parameters currently defined.

Tides tab provides various graphical display of the model tidal forcing conditions.

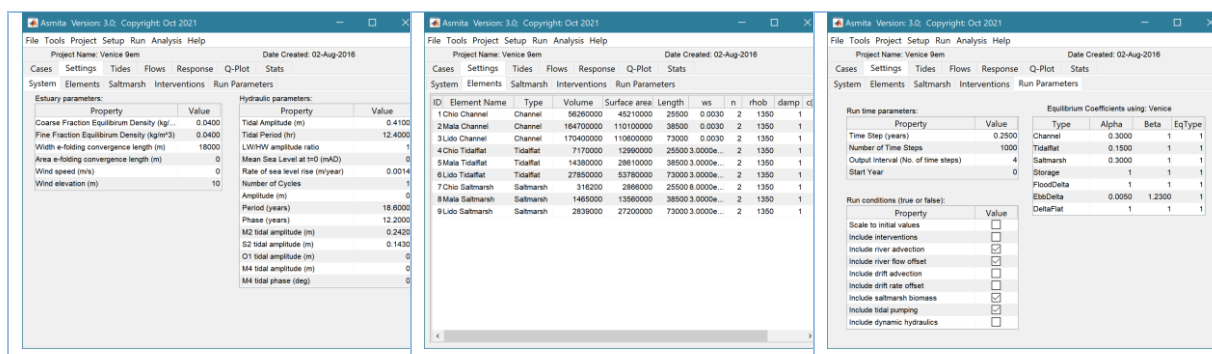
Flows tab provides various graphical display of the model advection forcing conditions.

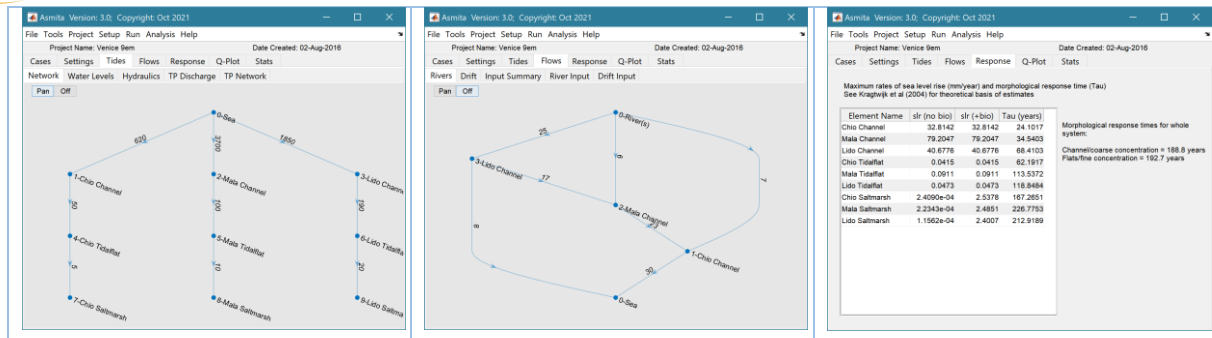
Response: summary of morphological response times based on system definition.

Q-Plot: displays a quick-plot defined for the class of the selected case (display only).

Stats: displays a table of results for any analyses that have been run (can be copied to clip board).

Figure 3 – Tabs in Main GUI





Adding a river flow:

Setup>River: define channel element that river flows into, flow rate and sediment concentration.

Setup>Estuary>Advection: define how the river flows through the model. Again, this is specified as from-to and the total flow into any element must equal the flow out.

Setup>Run Parameters>Conditions: check the flag for River flow offset if you want the flow to be accounted for in the initial condition.

Note: river flows can also be added and edited in the Advection definition table but this still requires the concentration of any sediment load to defined using the **Setup>River** user interface.

Adding interventions:

Setup>Interventions: select element and define the years in which there is a change. Positive is an increase in water volume or surface area.

Setup>Run Parameters>Conditions: check flag to include interventions. (Note that once interventions have been defined, they can be included or omitted by using this flag).

To run the model:

Run>Check input: Checks that what components have been set-up and lists those that have not been set-up (Figure 4).

Run>Run model: Runs setup check and then runs model. If not enough components are defined a warning dialogue is shown and the run aborted. If the model runs the user is prompted for a description of the scenario (model run).

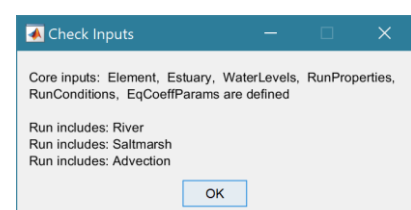
If the time step specified is likely to be unstable the user is prompted with a suggested time step and the number of steps to simulate the same period of time. The user can select this alternative, use the value specified in the model setup, or abort the run.

A summary of the total volume changes for the run can be viewed on the **Q-Plot** tab.

To plot the results:

Analysis>Plot menu: launches the plotting UI. Using the **Time** Tab, Cases can be selected and the results for any (or All) elements can be plotted for each variable (volumes, prism, water levels, concentration, etc).

Figure 4 – Check inputs



3 Sample models

The example folder of the App contains a number of sample models as *.mat files. The parameters needed to run the models are saved in these files (along with any scenarios that have been run). These include:

Humber 3EM.mat – a simple model to illustrate the original delta-channel-flat concept (Section 3.1).

Venice 9EM.mat – a model of Venice lagoon with 3 inlet channels with river inputs and linked tidal flats and saltmarshes. Illustrates use of multiple inlets and saltmarsh element (Section 3.2).

Ameland 3EM2S.mat – model of one of the inlets in the Wadden Sea that illustrates the use of mixed sediments (Section 3.3).

Severn 16EM.mat – a representation of the Severn estuary with multiple river sources and a series of channel reaches and tidal flats to illustrate the inclusion of tidal pumping (Section 3.4).

Inlet 7EM.mat – a model of a small harbour on the south coast of England to illustrate the inclusion of littoral drift across the inlet entrance (Section 3.5).

3.1 Humber Model (simple 3 element model: delta-channel-flat)

The settings used in the Humber3EM model are as follows:

Setup>Estuary>System Parameters: Coarse Fraction Equilibrium Concentration = 0.72 kg/m³. Fine fraction value is the same in this example (see Section 3.3 for details of how to introduce mixed sediments).

Setup>Estuary> Water Levels: Tidal Amplitude = 2.92 m (at mouth)

Setup>Estuary> Water Levels: LW/HW amplitude ratio = 1 if the tidal signal (or related cycles) is not symmetric about mean tide level, this ratio allows an adjustment to be made (the default is 1).

Setup>Estuary> Water Levels: Mean Sea Level at t=0 = 0 m. Allows for any offset of MSL to the local datum used for elevations.

Setup>Estuary> Water Levels: Rate of sea level rise = 0.0018 m/year

Setup>Estuary> Water Levels: Number of Cycles = 2

The next three fields can have multiple values to include the number of cycles specified

Setup>Estuary> Water Levels: Amplitude = 0.1 0.04 m

Setup>Estuary> Water Levels: Period = 18.6 180 years.

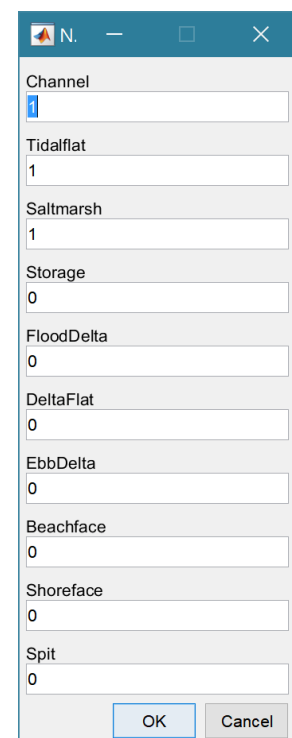
Setup>Estuary> Water Levels: Phase = 12.2 128 years. Defines the offset to give the correct timing of each cycle in the Julian calendar.

Setup>Element>Define Elements: specify how many elements of each type are needed. A table then lists all elements and the user can amend the Type of element, the name given to the element and whether it is erodible or not.

The elements are then created and the parameters for each element must all be defined as follows:

Setup>Element>Element Parameters: select element and define parameters. The following table details the parameters used in the 3 element Humber model:

Figure 5 – Specify number of each type of element



Channel	1
Tidal flat	1
Saltmarsh	1
Storage	0
Flood Delta	0
Delta Flat	0
Ebb Delta	0
Beachface	0
Shoreface	0
Spit	0

	Channel	Tidal Flat	Ebb Delta
Initial volume (m ³)	1.079e9	3.97e8	7.719e8
Initial plan area (m ²)	1.867e8	9.85e7	5.0e7
Length (m)	1.34e5	1.34e5	1.0e4
Vertical exchange (m/s)	0.003	0.003	0.003
Transport coefficient (-)	2	2	-2
Bed density (kb/m ³)	1350	1350	1350
Sediment mobility	1	1	1

Notes:

- Volumes are between the upper and lower surface of the element eg low water and bed for channel elements and high water and low water for tidal flat elements
- Plan areas are measured on the upper surface. For Tidal flats this is the area over the flat at high water and does not include the area of the channel.
- Length is the along channel length of the element.
- Vertical exchange and Transport coefficient are determined as explained in Townend et al (2016). The Transport coefficient for the Ebb Delta is negative because the equilibrium condition of this type of element is in terms of sediment volume, whereas the others are all in terms of water volume.
- Bed density is a representative density for the bed in each element.
- Sediment mobility is only used when tidal pumping is included to adjust the along channel concentrations.

Setup>Run Parameters>Time Step: Time Step = 0.5 (years)

Setup>Run Parameters>Time Step: Number of Time Steps = 640 (i.e., 320 years)

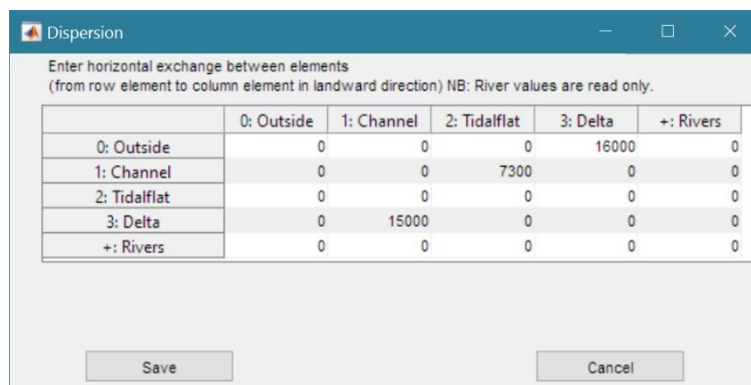
Setup>Run Parameters>Time Step: Output Interval = 2 (i.e., every other time step)

Setup>Run Parameters>Time Step: Start Year = 1703 (default is zero and only needed if doing a historical hindcast. However, any Interventions must be defined relative to the same time frame)

Setup>Run Parameters>Equilibrium Coefficients: select ‘Default’.

Setup>Estuary>Dispersion: define the horizontal exchange between elements. A row element defines the ‘supply’ and the column element defines the ‘receptor’ for a flooding tide to define a landward flow path from the most seaward element(s), Figure 6. This is used to define the structure of the element network and to calculate the tidal prism in each reach.

Figure 6 – Add horizontal exchange rates using the Dispersion table UI



Dispersion

Enter horizontal exchange between elements
(from row element to column element in landward direction) NB: River values are read only.

	0: Outside	1: Channel	2: Tidalflat	3: Delta	+: Rivers
0: Outside	0	0	0	16000	0
1: Channel	0	0	7300	0	0
2: Tidalflat	0	0	0	0	0
3: Delta	0	15000	0	0	0
+: Rivers	0	0	0	0	0

Save Cancel

A convenient way to check the Dispersion set-up is to use the *Tides>Network* tab to plot the tidal network from Outside (Sea) to the most landward elements (including any branches)

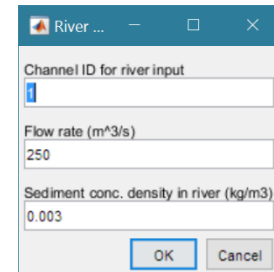
Setup>Run Parameters>Conditions: check flag to ‘Scale to initial values’.

This is sufficient information for the model to run.

Adding a river flow:

Setup>River: define channel element that river flows into, flow rate and sediment concentration, Figure 7. The sample model uses a constant river discharge and sediment load. It is possible to use a time varying input for the river discharge and sediment load, as explained in Sections 4.4 and 5.3.1.

Figure 7- Define a river input to chosen channel, with flow rate and concentration density of sediment load



Channel ID for river input

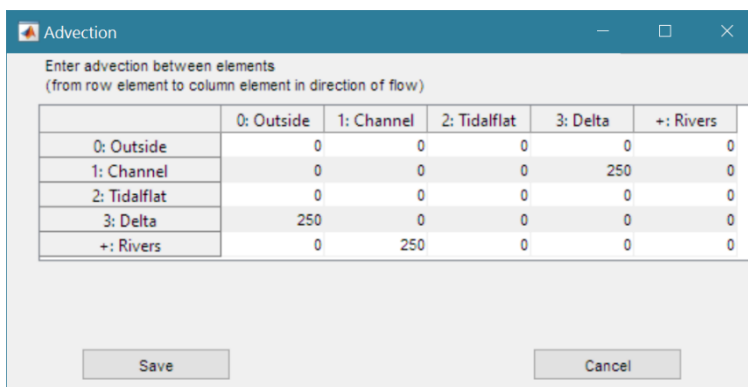
Flow rate (m³/s)

Sediment conc. density in river (kg/m³)

OK Cancel

Setup>Estuary>Advection: define how the river flows through the model. Again, this is specified as from row element - to column element and the total flow into any element must equal the flow out, Figure 8.

Figure 8 – Add advection flow rates using the Advection Table UI



Enter advection between elements
(from row element to column element in direction of flow)

	0: Outside	1: Channel	2: Tidalfat	3: Delta	+: Rivers
0: Outside	0	0	0	0	0
1: Channel	0	0	0	250	0
2: Tidalfat	0	0	0	0	0
3: Delta	250	0	0	0	0
+: Rivers	0	250	0	0	0

Save Cancel

A convenient way to check the Advection set-up is to use the **Flows>Rivers** tab to plot the flow network from Source (River or Rivers) to Outside (Sea).

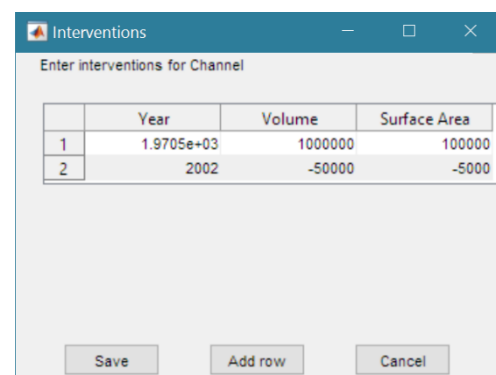
Setup>Run Parameters>Conditions: check the flag to ‘Include river advection’ and the flag for ‘River flow offset’, if you want the flow to be accounted for in the initial condition.

Adding interventions:

Setup>Interventions<Add: select element from list table and use the table UI to define the years in which there is a change, Figure 9. Positive is an increase in water volume or surface area. Interventions can also be loaded from a file using **Setup>Interventions<Load**, see Section 5.5.1 for details.

Setup>Run Parameters>Conditions: check flag to include interventions, Figure 10. (Note that once interventions have been defined, they can be included or omitted from an individual model by using this flag, but remain saved in the model set-up).

Figure 9 - Define interventions by year, volume change and plan area change. Use add row to include additional changes



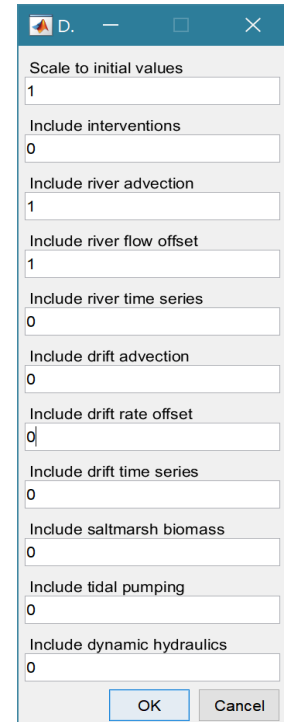
Enter interventions for Channel

	Year	Volume	Surface Area
1	1.9705e+03	1000000	100000
2	2002	-50000	-5000

Save Add row Cancel

Figure 10 – Conditions UI allows conditions and constraints to be included or omitted from model run

Set values to 1 to include and 0 to exclude.



3.2 Venice model (multiple inlets and saltmarsh)

The steps to setup the model follow those outlined above. The main differences here are the addition of multiple inlets which are defined using the Dispersion and Advection parameters and the inclusion of saltmarsh. The following summarises the settings used in the Venice 9EM model and details the additional steps to setup the model.

Setup>Estuary>System Parameters: Coarse Fraction Equilibrium Concentration = 0.04 kg/m³.

Setup>Estuary> Water Levels: Rate of sea level rise = 0.0014 m/year

The element parameters are as detailed in Figure 11. The dispersion and advection are shown in Figure 3. To include the river flows in the model run use **Setup>Run Parameters>Conditions:** to check the flag to ‘Include river advection’ and the flag for ‘River flow offset’.

Figure 11 – Element definitions for Venice 9EM model

Element Name	Type	Volume	Surface area	Length	ws	n	cb	c(ero)	Ero...
Chio Channel	Channel	56260000	45210000	25500	0.0030	2	0.2000	1	1
Mala Channel	Channel	164700000	110100000	38500	0.0030	2	0.2000	1	1
Lido Channel	Channel	170400000	110600000	73000	0.0030	2	0.2000	1	1
Chio Tidalflat	Tidalflat	7170000	12990000	25500	3.0000...	2	0.2000	1	1
Mala Tidalflat	Tidalflat	14380000	28610000	38500	3.0000...	2	0.2000	1	1
Lido Tidalflat	Tidalflat	27850000	53780000	73000	3.0000...	2	0.2000	1	1
Chio Saltmarsh	Saltmarsh	316200	2866000	25500	3.0000...	2	0.2000	1	1
Mala Saltmarsh	Saltmarsh	1465000	13560000	38500	3.0000...	2	0.2000	1	1
Lido Saltmarsh	Saltmarsh	2839000	27200000	73000	3.0000...	2	0.2000	1	1

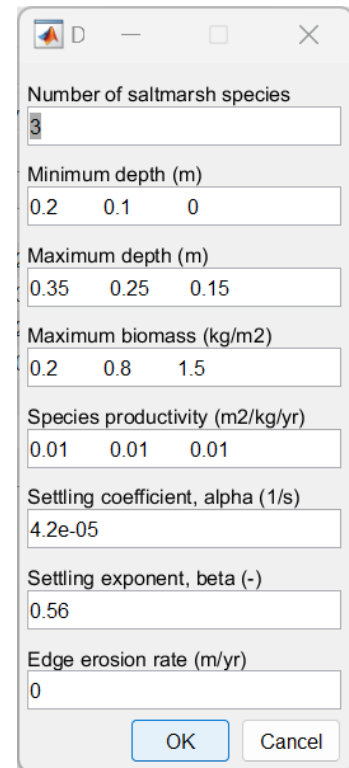
Setup>Tidal Constituents: M2 amplitude = 0.242 m.

Setup>Tidal Constituents: S2 amplitude = 0.143 m.

Setup>Saltmarsh>Species Parameters

The user interface to enter the saltmarsh parameters allows the user to enter multiple species values with values separated by a space as illustrated in Figure 12 (applies to minimum depth, maximum depth, Maximum biomass and Species productivity). Further explanation of the saltmarsh parameters is given in Section 1.

Figure 12 – Define saltmarsh species and enhanced settling parameters



Marsh edge erosion rate due to wave action

Setup>Run Parameters>Conditions: check the flag to ‘Include saltmarsh biomass’. If the flag is not checked, the model treats the saltmarsh element like any other element and uses the prism-based relationship to define the equilibrium volumes (see Section 6.1).

Setup>Run Parameters>Equilibrium Coefficients: select ‘Venice’.

3.3 Amelander model (mixed sediments – experimental code)

This model is used to explore the influence of introducing variable bed density and mixed (fine and course sediments). The original model was detailed in van Goor *et al.* (2003). This model was revisited to consider the influence of fine sediment in Wang *et al.* (2014) and extended to consider the combined influence of mixed sediment and bed density in Townend *et al.* (2016b). The file provided has the parameters setup for case (vi) in Table 2 of Townend *et al.* (2016b).

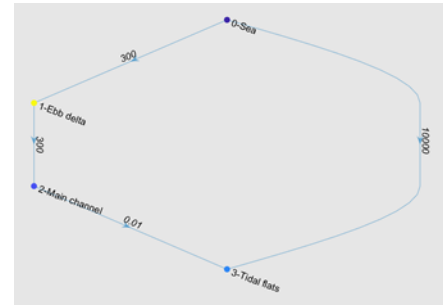
Setup>Estuary>System Parameters: *Coarse Fraction Equilibrium Concentration* = 0.64 kg/m³.

Setup>Estuary>System Parameters: *Fine Fraction Equilibrium Concentration* = 0.2 kg/m³.

Figure 13- Element definitions for Amelander 3EM2S model

Element Name	Type	Volume	Surface area	Length	ws	n	cb	c(ero)	Ero...
Ebb delta	Delta	131000000	74700000	4000	0.0100	-5	0.6000	1	1
Main channel	Channel	302000000	98300000	20000	0.0100	5	0.6000	1	1
Tidal flats	Tidalflat	120000000	178000000	20000	2.0000...	-3	0.3538	1	1

The element parameters are shown to the right, with dispersions of 300, 300, 10,000 as the exchanges from Outside to Ebb Delta, Ebb Delta to Channel and Outside to Tidal Flat respectively. The small exchange from the Channel to the Tidal Flat ensures that the water volume of the tidal flat is included in the calculation of tidal prism. If this value is too large the sediment mass balance will show a small discrepancy.



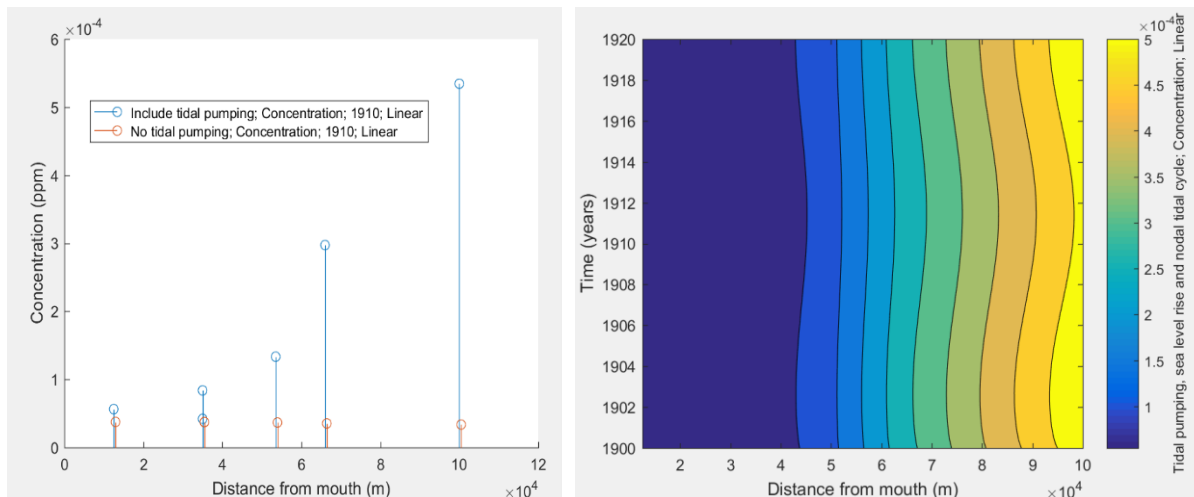
Setup>Estuary> Water Levels>Rate of sea level rise = 0.006 m/year

Setup>Run Parameters>Equilibrium Coefficients: select 'Amelanders'.

3.4 Severn model (tidal pumping)

This model illustrates the inclusion of tidal pumping. The set-up procedure is similar to that used for the Humber (Section 3.1). The only addition is to set the condition to include tidal pumping by selecting the 'Include tidal pumping' condition in *Setup>Run Parameters>Conditions*. For the use of tidal pumping to be meaningful, the channel needs to be sub-divided into several elements (6 are used in this case). The model also shows the use of multiple tidal flats linked to each channel reach (one for each side of the estuary). This can be useful for illustrating change but the dynamics of the model are the same if a single tidal flat is used for each reach (assuming the parameters are the same for each pair of flats). This model also allows the plotting of variations along the channel using the Distance and XYZ tabs in the plot GUI, as illustrated in Figure 14.

Figure 14 – Plots from Severn model showing along channel variation at a point in time and as space-time surface

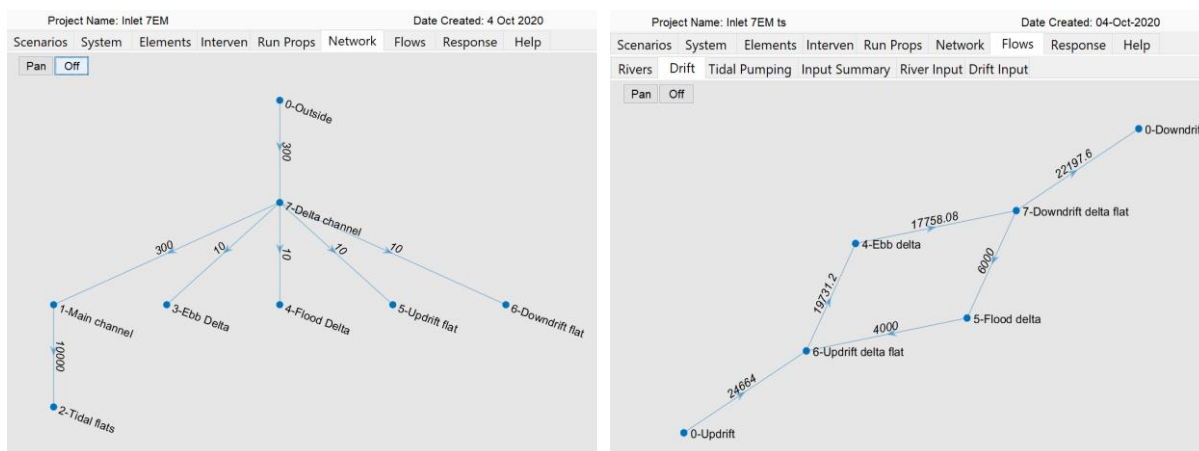


3.5 Inlet model

This model illustrates possible ways of including littoral drift into the model. Two versions are provided (Inlet 7EM.mat and Inlet7EMts.mat). Whilst both use the same element configuration, one has constant drift rates defined in the advection matrix (*Setup>Littoral Drift>Drift advection*) and the other has a time series prescribed for each element (*Setup>Littoral Drift>Drift timeseries*). A simple 3 element model with an ebb delta element can include drift by simply defining a drift source and the drift rate (advection) to and from the delta (see Section 4.4). To illustrate the process a more extensive delta network is defined in the Inlet model with 4 delta elements and delta channel. The delta elements comprise flood and ebb delta and updrift and downdrift delta flat. The estuary dispersion (horizontal exchange) defines the connectivity for tidal propagation. The delta channel connects to the outside

world and upstream to the first inlet/estuary channel. All the delta elements have a horizontal exchange with the delta channel. This ensures that all delta elements are governed by the same upstream tidal prism, allowing equilibrium volume to be defined in terms of tidal prism, as used for a single delta element. The Generic set of equilibrium conditions include settings for all delta elements drawn from the literature and are loaded using *Setup>Run Parameters>Select Eq. Coefficients* (see Sections 4.4 and 5.4). Alternatively, site specific values can be defined and loaded using the specification file *userprismcoefficients.m* (see Section 6.1) or by editing the selected values using *Setup>Run Parameters>Edit Eq. Coefficients*. As with other cases the equilibrium defined by any set of coefficients can be scaled to the existing volumes by selecting the ‘Scale to existing values’ option in *Setup>Run Parameters>Conditions*.

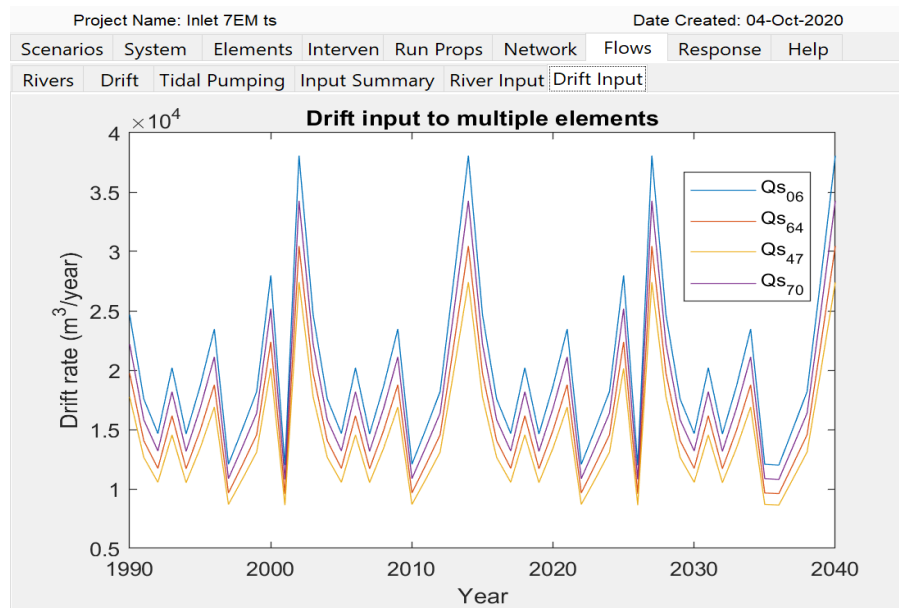
Figure 15 – Network and drift Flow connectivity for the Inlet 7EM model



The advection is defined by first adding a drift input and then defining the pathway through the network using *Setup>Drift>Drift inputs* and *Setup>Drift>Drift Advection*, respectively. For the Inlet model supplied, a circulation cell within the delta is defined so that the pathway is from the upstream delta flat to the ebb delta and on to the downstream delta flat, with a return pathway taking some of the sediment arriving at the downstream flat to the flood delta and then on to the upstream delta flat (NB: this is for illustrative purposes only). Once Advection throughout the network has been defined the flow routes can be checked using the *Flows>Drift* tab. One or more timeseries of drift rates can be included using the *Setup>Drift>Drift Timeseries* menu option, as explained in Sections 4.4 and 5.3.2. For these to be included in the run use *Setup>Run Parameters>Conditions* to check the flag to ‘Include river advection’ and the to ‘Include drift rate offset’, if you want the flow to be accounted for in the initial condition.

It is possible to add a river advection pathway, as used in some of the previous models, in addition to a drift pathway.

Figure 16 – tab to display drift time series loaded. The naming convention is suffixes for from and to node ids, e.g. *QS64* is a drift from element 6 to element 4.



4 Application Menus

The UI comprises a series of drop-down menus that provide access to a number of commonly used functions such as file handling, management of run scenarios, model setup, running and plotting of the results. In addition, Tabs are used to display set-up information of the Cases that have been run. In this manual text in *Red italic* refers to drop down menus and text in *Green italic* refers to Tab titles.

4.1 File

File>New: clears any existing model (prompting to save if not already saved) and a popup dialog box prompts for Project name and Date (default is current date).

File>Open: existing models are saved as *.mat files. User selects a model from dialog box.

File>Save: save a file that has already been saved.

File>Save as: save a file with a new or different name.

File>Exit: exit the program. The close window button has the same effect.

4.2 Tools

Tools>Refresh: updates *Cases* tab.

Tools>Clear all>Project: deletes the current project, including setup parameters and all Cases.

Tools>Clear all>Figures: deletes all results plot figures (useful if a large number of plots have been produced).

Tools>Clear all>Cases: deletes all cases listed on the *Cases* tab but does not affect the model setup.

4.3 Project

Project>Project Info: edit the Project name and Date.

Project>Cases>Edit Description: select a scenario description to edit.

Project>Cases>Edit Data Set: edit a data set. Initialises a data selection UI to define the record to be edited and then lists the variable in a table so that values can be edited. The user can also limit the data set retrieved based on the variable range and the independent variable (X) or time. This can be useful

in making specific edits (e.g., all values over a threshold or values within a date range). Using the Copy to Clipboard button also provides a quick way of exporting selected data.

Project>Cases>Save: select the Case to be saved from the list of Cases, select whether to save the Case as a *dstable* or a *table* and name the file. The dataset *dstable* or *table* are saved to a mat file.

Project>Cases>Delete: select the Case(s) to be deleted from the list of Cases and these are deleted (model setup is not changed).

Project>Cases>Reload: select a previous model run and reload the input values as the current input settings.

Project>Cases>View settings: display a table of the model input parameters used for a selected Case.

Project>Import/Export>Import: load a Case class instance from a Matlab binary 'mat' file. Only works for data sets saved using Export.

Project>Import/Export>Export: save a Case class instance to a Matlab binary 'mat' file.

These last two functions can be used to move Cases between projects or models.

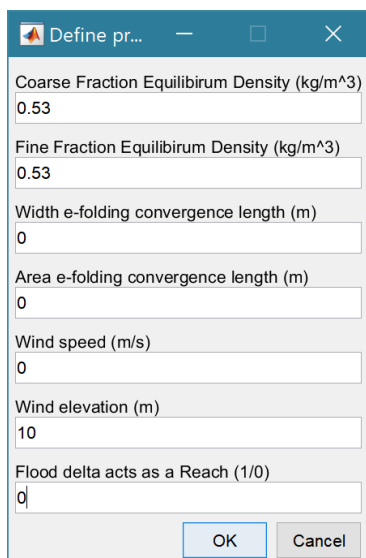
NB: to export the data from a Case for use in another application (e.g., text file, Excel, etc), use the **Project>Cases>Edit Data Set** option to select and then use the 'Copy to Clipboard' button to paste the selection to the clipboard.

4.4 Setup

The setup menu provides a series of menus to enable different components of the model to be defined. This includes menus for estuary wide parameters, element properties, saltmarsh properties, river and littoral drift advective inputs, interventions, inclusion of an hydraulic model, and the run parameters.

4.4.1 Estuary

Setup>Estuary>System Parameters:



Define the concentration that is target or global equilibrium concentration. If included, the fine fraction is used for tidalflats, saltmarsh and storage elements and the coarse fraction for all other element types. When only a single fraction is being considered the fine fraction should be the same as the coarse value, empty or 0.

The width e-folding length is the rate of width convergence and is used if tidal pumping or dynamic hydraulics is included.

The area e-folding length is the rate of cross-sectional area convergence and is used if tidal pumping or dynamic hydraulics is included.

Wind speed and the elevation at which the wind is measured (not used in current version).

If a Flood Delta element is included, this can be treated as a Reach or as a component of a Reach. Set value to 1 to include the flood delta as a reach. Otherwise Reaches are defined by the Channel Elements.

Setup>Estuary>Water Levels:

Tidal range and period are defined at the entrance to estuary, or inlet.

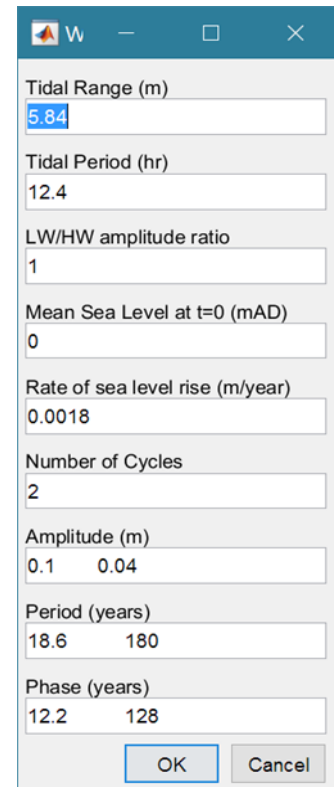
The LW/HW amplitude ratio accounts for any difference in the amplitude to low water and high water relative to the define mean sea level¹. Used to adjust influence of changes in tidal range.

The mean sea level at the start of the model run (allows for offsets from zero datum).

Rate of sea level rise as a linear rate (positive or negative). This can also be used to define an exponential rate of slr or to call a user function. See Section 6.2 for the details of how to use these options.

Number of cycles to be included that vary the tidal range as a function of time. The number of entries for amplitude, period and phase must agree with the number of cycles specified. The phase defines a shift in the cycle to align with the Julian calendar. This is only of relevance if the model is being used to hindcast change and compare volume changes with observed changes (e.g. Townsend et al., 2007).

The Water Level UI has a second page for entering certain tidal constituents which are used if Saltmarsh or Tidal Pumping are included.



W

Tidal Range (m)
5.84

Tidal Period (hr)
12.4

LW/HW amplitude ratio
1

Mean Sea Level at t=0 (mAD)
0

Rate of sea level rise (m/year)
0.0018

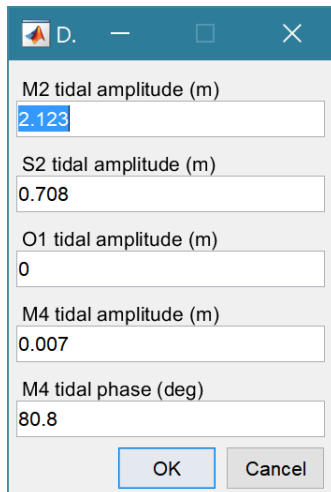
Number of Cycles
2

Amplitude (m)
0.1 0.04

Period (years)
18.6 180

Phase (years)
12.2 128

OK Cancel



D.

M2 tidal amplitude (m)
2.123

S2 tidal amplitude (m)
0.708

O1 tidal amplitude (m)
0

M4 tidal amplitude (m)
0.007

M4 tidal phase (deg)
80.8

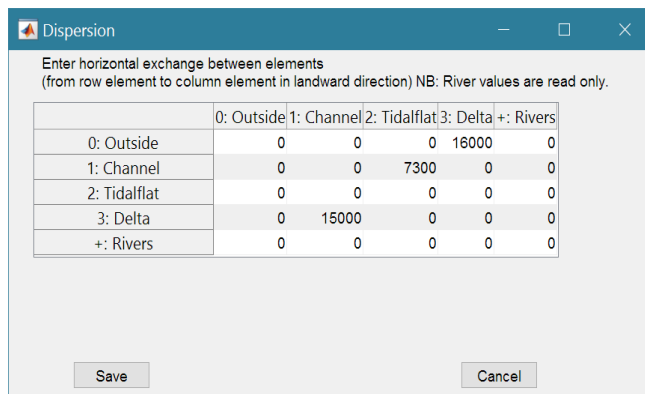
OK Cancel

The model currently assumes a semi-diurnal tide and the M2, S2 and O1 amplitudes are used to estimate the concentration over the marsh and the submergence time if Saltmarshes are included in the model.

The M4 amplitude and phase are used to estimate the influence of the over-tide in the tidal pumping component.

¹ Option included because of observed bias (e.g., Humber nodal tidal cycle has an amplitude of 0.23m with a high-water amplitude of ~0.14 and low water amplitude of ~0.08; i.e. ratio = 0.08/0.14 = 0.57). The default value is 1.

Setup>Estuary>Dispersion:



Enter horizontal exchange between elements
(from row element to column element in landward direction) NB: River values are read only.

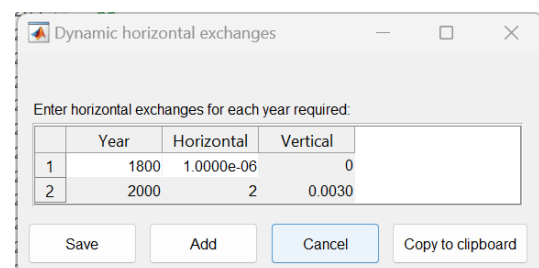
	0: Outside	1: Channel	2: Tidalflat	3: Delta	+: Rivers
0: Outside	0	0	0	16000	0
1: Channel	0	0	7300	0	0
2: Tidalflat	0	0	0	0	0
3: Delta	0	15000	0	0	0
+: Rivers	0	0	0	0	0

Save Cancel

Define the horizontal exchange between elements (m/s). Exchanges proceed from the most seaward element(s) in a landwards direction. Values are entered in the cell which is from a row element going to a column element (e.g., from row Delta to column Channel). These are used to determine the network connectivity and the resultant network can be viewed on the **Tides>Network** tab. For detail of how to estimate appropriate values for the horizontal exchange see Wang *et al.* (2008) or Townend *et al.* (2016a).

Setup>Estuary>Dynamic Exchanges:

Defining one or more values in the dispersion matrix (above) with a value of NaN, invokes the use of Dynamic Exchanges. For this to work, at least one dynamic exchange must be defined using this menu option. The UI is initialised with values of '0' for both year and vertical exchange rate and 1e-6 for the horizontal exchange. As a minimum, set the Year in row 1 to the start year of the model run and the Exchanges to the initial value required. With just one row defined from the start year, the model will run as if the horizontal value had been defined in the dispersion matrix and using the vertical exchange defined for the element. By adding, additional rows the dispersion matrix and transient vertical exchanges are updated with the new exchange rates, as the run progresses, based on the defined change year. The new values are then used until the next change.



Enter horizontal exchanges for each year required:

	Year	Horizontal	Vertical
1	1800	1.0000e-06	0
2	2000	2	0.0030

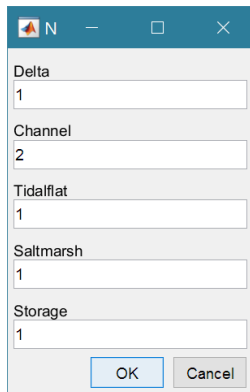
Save Add Cancel Copy to clipboard

Using these settings in conjunction with Interventions, provides the option to activate elements during the run (see Section 5.5.3 for further information about using this setting to represent the inclusion of managed realignments and storage schemes). For example, setting an initially small value for the horizontal exchange limits the ability of the linked elements to interact and subsequently increasing the value then emulates increasing the rate of exchange between elements. Similarly altering the vertical exchanges, adjusts the rate of erosion or deposition possible within an element. Setting the vertical exchange to 0 has the effect of inhibiting any subsequent change in volume and, if saltmarsh bio-productivity is active, there is no further change in biomass volume until the vertical exchange becomes non-zero.

Setup>Estuary> Model Constants: various constants are defined for use in models, such as the acceleration due to gravity, viscosity and density of sea water, and density of sediment. Generally, the default values are appropriate (9.81, 1.36e-6, 1025, 2650 respectively) but these can be adjusted and saved with the project if required.

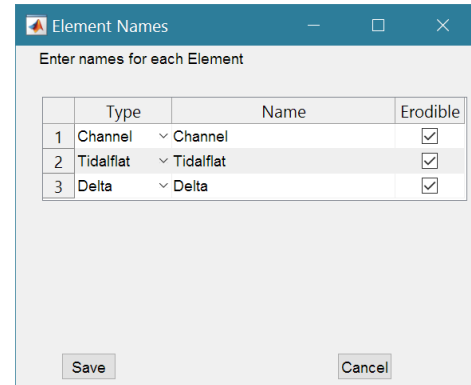
4.4.2 Element

Setup>Element>Define Elements:



When setting up a new model a listing of the different types of element allows the user to define the number of each type element to include in the model (elements can subsequently be added or deleted). Once a set of elements have been defined the user is prompted to edit the default names for each element, change the element type and choose to make the element non-erodible (for cases where the element can infill with sediment but if cannot erode).

After the initial setup, the user is taken straight to the Element Names dialog to edit the available list of elements. To add or delete elements use the dialogs from the main menu



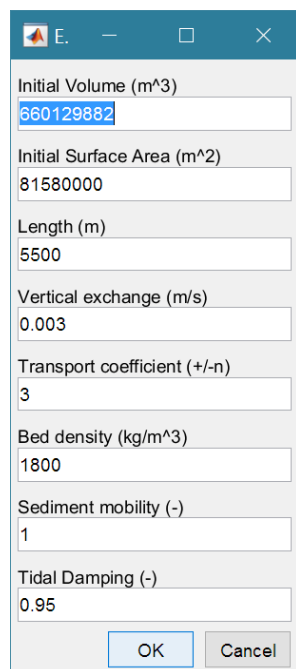
	Type	Name	Erodible
1	Channel	Channel	<input checked="" type="checkbox"/>
2	Tidalflat	Tidalflat	<input checked="" type="checkbox"/>
3	Delta	Delta	<input checked="" type="checkbox"/>

Setup>Element>Add and *Setup>Element>Delete*.

The definitions of the Types of Element currently available are given in Section 5.1.

Setup>Element>Element Parameters:

Definition of elements is on a continuous loop. Select Cancel from the element list box to finish defining element parameters.



Volume and surface area are defined for each element. The length is the distance along the flow axis (usually from seaward limit to landward limit). The vertical exchange (m/s) is similar to a fall velocity.

The transport coefficient represents the scaling of the dominant mode of transport (i.e. velocity to some power)². Note where sediment volumes are being used instead of water volumes the value should be **negative**.

The basis for setting the Vertical exchange and the Transport coefficient is discussed in Wang *et al.* (2008) or Townend *et al.* (2016a).

The wet or bulk density of the bed of the element.

The sediment mobility is only relevant if tidal pumping is included and is used to adjust the relative influence of elements in the along estuary channel concentration gradient.

The tidal damping allows along channel changes in tidal amplitude to be included. This defines the change in amplitude relative to the value at the mouth. This mainly influences the calculation of the tidal prism, which in turn determines the equilibrium volumes. NB *only values for elements that define reaches are used* (eg Channel).

Setup>Element>Load Parameters:

Rather than enter data one element at a time, using the GUI shown above, this option allows the parameters to be loaded from a file. This can be an advantage for models with many elements.

Element are defined in the usual way using *Setup>Element>Define Elements*, and then the property data from the file is loaded. This must be in the order in which elements are defined. See Section 5.1.1 for details of the file format.

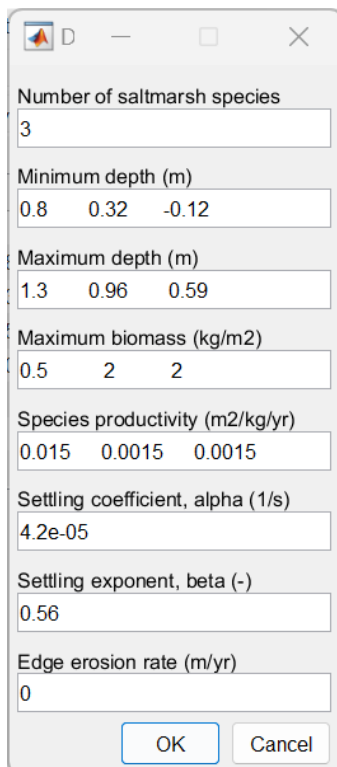
² If over the long-term the model is considered to represent a diffusion process, this would have a value of 2. Where advective transport (such as fluvial import dominates) then a value of 3 based on suspended load transport may be more representative, or a value of 5 where bed load transport dominates sediment exchanges.

Setup>Element>Add: select element type from list. The Define Element dialog box is then displayed allowing the user to edit the element name and whether it is erodible or not. Element parameters and any Dispersion or Advection then need to be added using the individual dialog boxes for these components.

Setup>Element>Delete: select element to be deleted from the list of model elements.

4.4.3 Saltmarsh

Setup>Saltmarsh>Species Parameters:



Number of saltmarsh species
3

Minimum depth (m)
0.8 0.32 -0.12

Maximum depth (m)
1.3 0.96 0.59

Maximum biomass (kg/m2)
0.5 2 2

Species productivity (m2/kg/yr)
0.015 0.0015 0.0015

Settling coefficient, alpha (1/s)
4.2e-05

Settling exponent, beta (-)
0.56

Edge erosion rate (m/yr)
0

OK Cancel

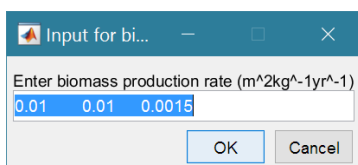
Any number of saltmarsh species can be entered and a check is made to ensure that the number of values entered for minimum depth, maximum depth maximum biomass and species productivity is consistent with the number of species specified.

Coefficients for biomass dependent enhanced settling rate

Marsh edge erosion rates due to wave action

Setup>Saltmarsh>Equilibrium Marsh Depth:

This utility allows the user to explore the response of the marsh depth to the defined saltmarsh parameters and to see how this response varies by adjusting the biomass production rates for each of the species included. On selecting this option, a dialog box presents the currently defined values of the biomass production rates and the user has the option to change these. *[This does not alter the values saved in the model.]* Three plots are then presented in a single figure.



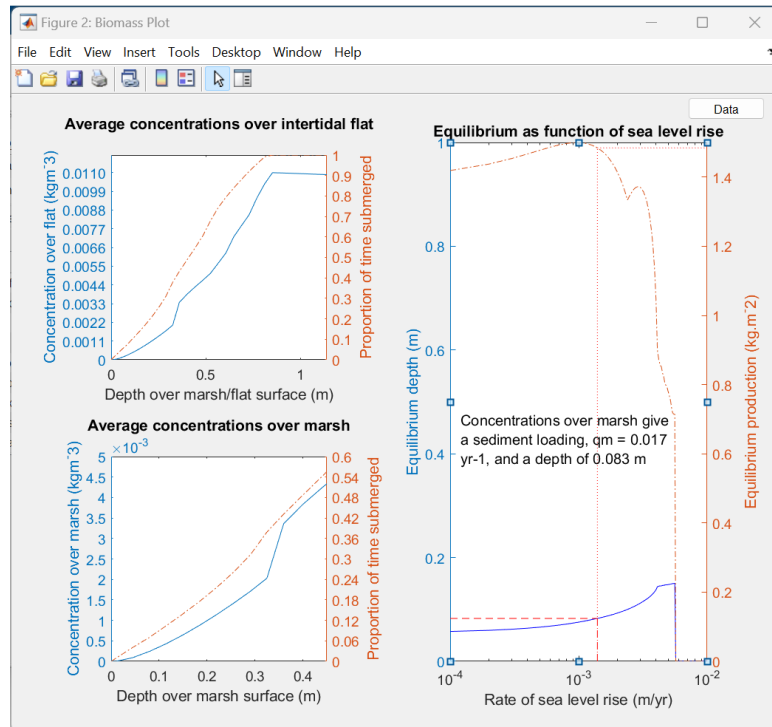
Input for bi...

Enter biomass production rate (m²kg⁻¹yr⁻¹)
0.01 0.01 0.0015

OK Cancel

These show the average concentration and the proportion of time submerged for the intertidal flat (top left) and for the marsh surface (bottom left). The plot to the right shows how the equilibrium depth (blue solid line) and the equilibrium production of the marsh (orange dash-dot line) vary as a function of sea level rise. The dashed and dotted lines highlight the values for the rate of sea level rise defined in the model. The 'Data' button in the top right of the plot panel, generates a table that shows the saltmarsh parameters used to create the plot.

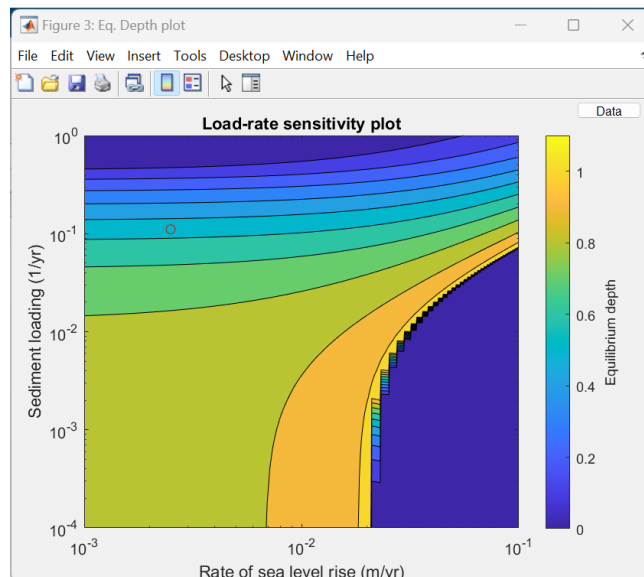
For further discussion of the saltmarsh model and the bio-morphological response, see Townend *et al.* (2010).



Setup>Saltmarsh>Biomass Distribution: plots the variation in biomass production of each species is distributed relative to tidal flat elevation (also displays on the **Settings>Saltmarsh** tab).

Setup>Saltmarsh>Sensitivity Plot: plots the variation of equilibrium depth as a function of sediment load, q_m , and rate of sea level rise, $dlsr$. Plot also includes a point for the initial conditions currently set in the model (uses the mean saltmarsh values of the vertical rate of exchange, ws , the saltmarsh settings and the global equilibrium concentration, cE).

Setup>Saltmarsh>Marsh-flat Animation: animation of the variation in marsh elevation over time starting from a bare tidal flat based on the current saltmarsh and water level settings'



4.4.4 River advection

Setup>Rivers>River inputs: This component can only be added once some channel elements have been setup in the model. If no rivers are defined, the dialog box requires the Channel id number, along with the flow rate and the concentration density of the rivers sediment load. If one or more rivers have already been added, a list box prompts the user to select from a list of channels with river inputs or to Add a river.

To remove an input, use **Setup>Rivers>Delete input** and select the source to be deleted from the list of channels with river inputs.

River flows can also be defined in the **Setup>Rivers>River advection** table but it is still necessary to add the Sediment concentration using the **Setup>Rivers>River inputs** dialog box.

Setup>Rivers>River advection:

Define the advective flows through the channel network (m^3/s). For river flows, these are defined from the River source to the Outside (the open coast). Values are entered in the cell which is from a row element going to a column element (e.g., from Delta to Outside). There should be a mass balance in and out of any element and along the flow path(s) from Source(s) to the Outside.

The sediment concentrations associated with the river inputs are defined in **Setup>Rivers>River inputs**. If a new river flow is added in the Advection table, a River object is added and the user is prompted to use the **Setup>Rivers>River inputs** component to add the sediment concentration.

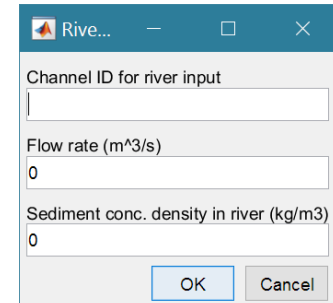
A river input with a sediment load that is not equal to the coarse grain equilibrium concentration will act as a perturbation to the model (a forced change). To avoid this there is an option to include the advective flows in the **Setup>Run Parameters>Conditions**: by setting the 'River flow offset' check box. The details of this are explained in the discussion on advective flows in Townend *et al.* (2016a).

Once Advection throughout the network has been defined the flow routes can be checked using the **Flows>Rivers** tab.

Setup>Rivers>River timeseries: If there is more than one river input, the user is prompted to select a river input channel. The user is then prompted for a text file with time series data to define river flow rates. The input file can also include sediment concentrations if time varying sediment flux data are available. Alternatively, varying flows with a constant concentration can be used. See Section 5.3.1 for details of the file format to use. To add another timeseries to a different river input channel, simply repeat the process. For the timeseries to be included in the run use **Setup>Run Parameters>Conditions** to check the flag to 'Include river timeseries'.

The flow path still needs to be defined using defined in the **Setup>Rivers>River advection** but the magnitude is not critical for the pathways with a timeseries source (not all river inputs need to use a timeseries) as long as the ratio of the flow is correctly apportioned at all branches (e.g. if at a branch, the flow in one downstream branch is twice the other, you could define the inflow to the element as having a magnitude of 3 and outflows with magnitudes of 1 and 2).

To change the timeseries, simply repeat the above step. To delete the use of a time series and revert to using a constant advection, use **Setup>Rivers>Delete input** and remove the selected input, then re-define the constant input using **Setup>Rivers>River inputs**.

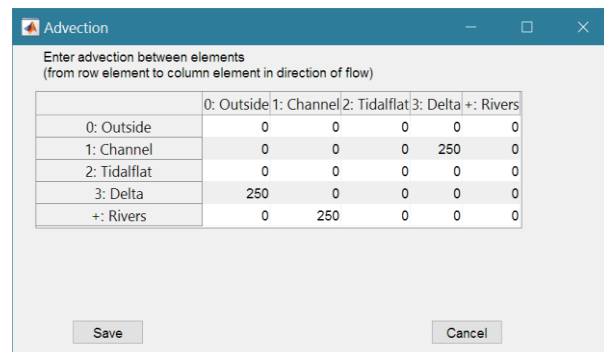


Channel ID for river input

Flow rate (m^3/s)

Sediment conc. density in river (kg/m^3)

OK Cancel



Enter advection between elements
(from row element to column element in direction of flow)

	0: Outside	1: Channel	2: Tidalfat	3: Delta	+: Rivers
0: Outside	0	0	0	0	0
1: Channel	0	0	0	250	0
2: Tidalfat	0	0	0	0	0
3: Delta	250	0	0	0	0
+: Rivers	0	250	0	0	0

Save Cancel

Timeseries start and end dates should cover the period being modelled based on the start date and run duration (end date = start date + time step x number of time steps) defined in *Setup>Run Parameters>Time step*. If the run extends beyond the range of the timeseries, the constant input values for flow rate and concentration, specified in *Setup>Rivers>River inputs*, are used.

The flow path can be seen on the *Flows>Rivers* tab, the inputs defined are tabulated on the *Flows>Input Summary* tab and if a time series is included this can be plotted on the *Flows>River Input* tab (NB: this tab plots all river inputs (and sediment loads if included) in a single graph).

Setup>River>Delete Input: allows user to delete a selected input.

4.4.5 Drift advection

A similar procedure to that used for rivers is followed to include drift in and out of elements. The drift is specified for each linkage as littoral drift (i.e., a volume of sediment per year – m³/year). There is an option, similar to the river setup to edit inputs to the system (*Setup>Drift>Drift inputs*). However, there is no need to specify a concentration and so drifts can be set up by just using the advection table.

Setup>Drift: Littoral drift is included in much the same way as river flows, except that a mass balance is not imposed. The source or inputs of drift can be defined as a drift rate (m³/year) using *Setup>Drift>Drift inputs*. However, there is no need to assign a concentration to the drift. The flow through the system is then defined using the advection table *Setup>Drift>Drift advection*, in the same way as for river flows. Drift inputs can be removed using *Setup>Drift>Delete input*.

Once Advection throughout the network has been defined the flow routes can be checked using the *Flows>Drift* tab.

Setup>Drift>Drift timeseries: To add a drift times series first define a source input element using *Setup>Drift>Drift inputs*. and the flow path for the drift using *Setup>Drift>Drift advection*. Then use *Setup>Drift>Drift timeseries* to select an input file for the pathway. The user is prompted for a text file with time series data to define drift rates as volumes of sediment per year (m³/year). This can either be a single input drift which is then passed through the network, based on the partitioning given by the initial advection matrix definition. Alternatively, a rate can be specified for selected links in the path. See Section 5.3.2 for details of the file format to use.

If there is more than one drift pathway and hence more than one drift input (e.g., upper and lower shoreface), the user is prompted to select a drift input element before proceeding to select a file for that pathway, in the same way as for a single pathway explained above.

Each flow path still needs to be defined using *Setup>Drift>Drift advection*. When only a source drift is defined, the values in the initial advection matrix determine the relative scaling in the exchange from one element to the next. There is no requirement for the drift rates in and out of a node to balance. This allows for changes in drift rate due to changes in orientation of the shore or on/offshore movement of sediment. This applies for both constant drift rates and a source defined by a timeseries. However, when there are multiple timeseries inputs along a particular pathway, the values in the advection matrix simply define the links between elements. If only some of the links in the pathway are assigned input timeseries, the remaining links use the constant rate defined in the advection matrix.

To change the timeseries, simply repeat the above step. For the timeseries to be included in the run use *Setup>Run Parameters>Conditions* to check the flag to 'Include drift timeseries'.

To delete the use of a time series and revert to using a constant advection, use *Setup>Drift>Delete input* and remove the selected input, then re-define the constant input using *Setup>Drift>Drift inputs*.

Timeseries start and end dates should cover the period being modelled based on the start date and run duration (end date = start date + time step x number of time steps) defined in *Setup>Run*

Parameters>. If the run extends beyond the range of the timeseries, the constant input value specified for the drift rate in **Setup>Drift>Drift inputs** is used.

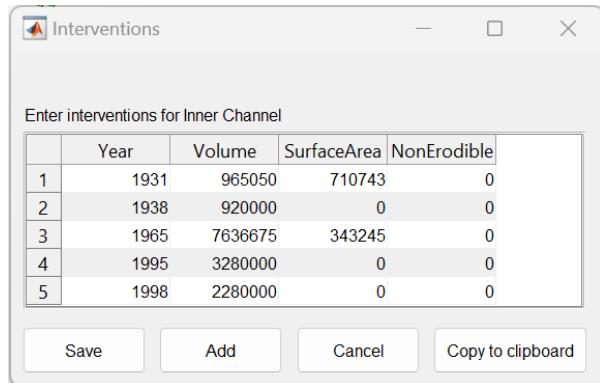
The flow path can be seen on the **Flows>Drift** tab, the inputs defined are tabulated on the **Flows>Input Summary** tab and if a time series is included this can be plotted on the **Flows>Drift Input** tab (NB: this tab plots all drift inputs in a single graph).

Setup>Drift>Delete Input: allows user to delete a selected input.

4.4.6 Interventions

Setup>Interventions>Add: select element from list of elements and then use table to add interventions. Selection of elements is in a continuous loop. Select Cancel from the element list box to finish adding interventions. Define the Year of the intervention (this must be consistent with the Start Year and duration of the model run defined in **Setup>Run Parameters>Time step**).

The Volume and Surface Area are the changes in water volume and element plan area respectively (regardless of whether the element is defined in terms of water, or sediment volume). Use negative values to represent reductions (e.g., due to a reclamation). The right hand column is a flag with a value of 0 if the change is erodible and 1 if it is non-erodible.



	Year	Volume	SurfaceArea	NonErodible
1	1931	965050	710743	0
2	1938	920000	0	0
3	1965	7636675	343245	0
4	1995	3280000	0	0
5	1998	2280000	0	0

Use the Add button to add a row to the bottom of the table. This will be a row of NaNs (extend window, or scroll to see the row). If the Year is left as NaN the row is removed when the table is saved. If the year and some values are set, the remaining NaN values are set to zero when saved. To delete a row, set the Year to NaN and Save the table. The data can be entered in any order as it is sorted to date order when saved.

For the defined interventions to be included in the model run the Include Interventions condition needs to be set. This allows runs to be made with and without the interventions, without the need to redefine them each time.

The interventions defined for any element can be viewed using the **Settings>Interventions** tab.

Setup>Interventions>Load: interventions can also be loaded from a file. Select an element from the list of elements and then select the file to load from the file selection dialogue. Selection of elements is in a continuous loop. Select Cancel from the element list box to finish adding interventions. The format for an element specific interventions file is a header and data entered in space or tab separated columns, as shown.

Setup>Run Parameters>Conditions: check the flag to 'Include Interventions'.

year	volume	area	non-erodible
1926	0	0	0
1927	420000	0	0
1928	420000	0	0
1929	420000	0	0
1930	420000	0	0
1931	965050	710743	0
1932	420000	0	0
1933	420000	0	0
1934	420000	0	0
1935	420000	0	0
1936	420000	0	0
1937	420000	0	0

4.4.7 Hydraulics

Setup>Hydraulics>Additional Parameters: define the additional parameters needed to run the CSTmodel.

The model uses the rates of width and CSA convergence defined using *Setup>Estuary>System Parameters* and the mean sea level tidal amplitude and period defined using *Setup>Estuary>Water Levels*. In addition, the estuary length and widths and cross-sectional areas at the mouth and te upstream river input.

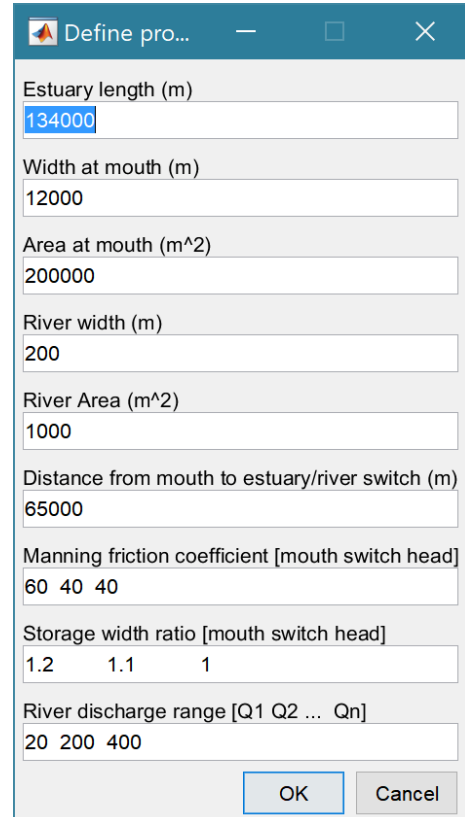
The distance from the mouth to estuary/river switch defines an intermediate point to allow the friction and storage widths to be varied along the channel. Each are specified as mouth, switch point and head values and the model uses linear interpolation to determine the along channel variation.

The range of river discharge is used to determine the number of intervals and river input to be used of each run of th4e CSTmodel. This is then used to create a look up table which is interpolated for any value of river discharge that may be defined as a model input (eg when using a river discharge timeseries).

NB: the CSTmodel App needs to be installed to use this option and further details are provided in the documentation and manual for the CSTmodel.

Setup>Hydraulics>Create Look-up: Uses the values specified in the Discharge Range vector to create a look-up table for a range of river discharges.

Setup>Hydraulics>Summary Plot: generates a summary plot of the along channel variations in water levels and velocities as a function of river discharge, allowing the range of variability as a function of river discharge to be investigated (this can also be viewed on the *Tides>Hydraulics* tab).



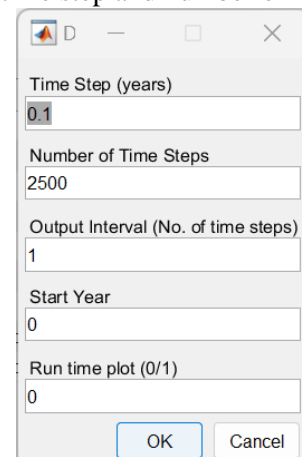
Estuary length (m)	134000
Width at mouth (m)	12000
Area at mouth (m ²)	200000
River width (m)	200
River Area (m ²)	1000
Distance from mouth to estuary/river switch (m)	65000
Manning friction coefficient [mouth switch head]	60 40 40
Storage width ratio [mouth switch head]	1.2 1.1 1
River discharge range [Q1 Q2 ... Qn]	20 200 400

OK Cancel

4.4.8 Run parameters

The time step parameters comprise the time step (years or parts of a year), the number of time steps and the intervals at which the model output should be saved. The product of time step and number of steps gives the run duration. The Start Year is relevant where and time dependent components are included that have used Years defined in the Julian calendar. If the results display erratic or chaotic behaviour, try reducing the length of the time step. There is an internal stability check but this does not cover all situations.

Note: a check is made at the start of a model run and the user is warned if the time step is too long and given the option to use (i) the value suggested by the stability check, (ii) the value specified in the model setup, or (iii) to abort the run. However, instabilities may still occur, especially when there is a large difference in the size of elements that make up the system. In such cases it is worth experimenting with shorter time steps to see if this removes or reduces the instabilities. If a warning is given and the user



Time Step (years)	0.1
Number of Time Steps	2500
Output Interval (No. of time steps)	1
Start Year	0
Run time plot (0/1)	0

OK Cancel

selects to continue with the specified time step, the code pads the first time step with additional time steps. In some cases this allows the model to rapidly adapt to the forcing conditions and run successfully.

If problems occur the Run time plot can be set to 1. This generates and updates a plot as the model runs showing the wet volumes and surface areas ($n > 0$) and the total tidal prism. Note: that this plot slows the run time of the model but can be a useful diagnostic.

Note: the Start Year and format for the time in AsmitaModel uses calendar years from 0 AD. To use the model for periods prior to this date requires the date format used in the code to be edited.

Setup>Run Parameters>Conditions:

Several model components can be defined in the model setup and the user can then select whether, or not, to include them in a particular model run (see Figure 10). The details of the conditions included in a particular run are saved with each run scenario and can be viewed using the *Project>Cases>View Case Settings*.

The Conditions that can be selected are defined as follows:

Condition	Description
Scale to initial values	Equilibrium volumes are scaled by the ratio of the initial volume and the initial prism based equilibrium volume
Include interventions	Interventions defined in the Setup>Interventions option are included in model runs when this option is selected
Include river advection	Allows river advection to be included or excluded
Include river flow offset	To avoid river inputs acting as a perturbation, the initial equilibrium is adjusted to account for the flow (see Section 3.4 of Townend <i>et.al.</i> , 2016a).
Include river time series	If a time series has been loaded for one or more of the river inputs, the data are used in the model run.
Include drift advection	Allows drift advection to be included or excluded
Include drift rate offset	To avoid drift inputs acting as a perturbation, the initial equilibrium is adjusted to account for the drift (see Section 6 of Townend <i>et.al.</i> , 2016b).
Include drift time series	If a time series has been loaded for drift sources or drift between elements, the data are used in the model run.
Include saltmarsh biomass	Once defined using Setup>Saltmarsh, the influence of saltmarsh productivity on morphology is included when this option is selected See Section 1 and Section 3 in Townend <i>et.al.</i> , 2016b.
Include tidal pumping	The influence of tidal asymmetry can be approximated by including tidal pumping (see Section 2 in Townend <i>et.al.</i> , 2016b for the theoretical background).
Include dynamic hydraulics	Use the lookup tables derived using the CSTmodel (see <i>Hydraulics>Create Look-up</i>) to define the along channel variation in water levels as a function of discharge. When a timeseries of river discharge is included this can mean that there is an additional change in the moving surface volumes due to changes in the water level stage.

Setup>Run Parameters>Select Eq. Coefficients: Select the *set* of equilibrium coefficients to be used from the list provided. These define the relationship to tidal prism for the various type of inlet/estuary element. The coefficient *sets* are defined in `userpriscoeffs.m`. If a new *set* of coefficients is added, the name of the *set* can be added using **Setup>Run Parameters>Edit Eq. Coefficients List**.

Setup>Run Parameters>Edit Eq. Coefficients: Edit the selected coefficients by Element Type. The UI prompts to select an element type and displays the current values for that type which can be edited. Further details on the equilibrium coefficients are given in Section 5.4.

Setup>Run Parameters>Edit Eq. Coefficients List: edit an existing list entry or add a new entry (NB: the name entered must agree with the case name used in `userprismcoeffs.m`).

4.5 Run

Run> Check input: checks mass balance of any advection flows included and that sufficient components have been defined for the model to run. This is not a full diagnostic check, so it does not ensure that the model output is valid – simply that the model will run.

Run> Run model: runs model, prompts for scenarios description which is added to the listing on the **Cases** tab.

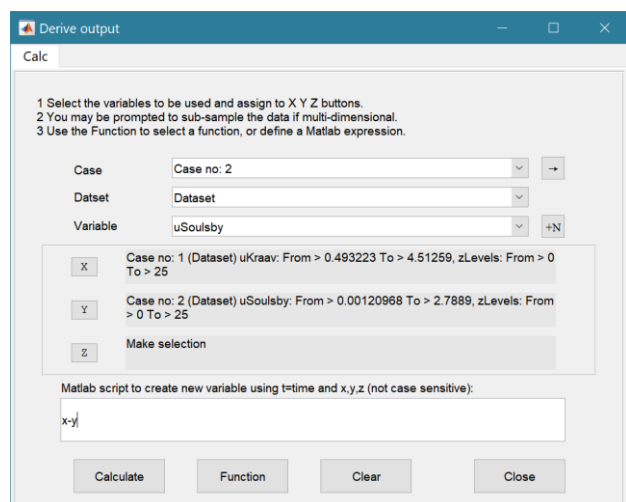
Run> Derive Output: data that has been added (either as data or modelled values) can be used to derive new variables. The UI allows the user to select data and use a chosen selection of data/variable/range to define either a Variable, XYZ dimension, or Time. Each data set is sampled for the defined data range. If the data set being sampled includes NaNs the default is for these to be included (button to right of Var-limits is set to '+N'). To exclude NaNs press the button so that it displays '-N'.

The selection is assigned by clicking one of the X, Y or Z buttons. The user is prompted to assign a Variable, XYZ dimension, or Time (the options available varies with the type of variable selected) – see Section 4.9 for details of how this works. An equation is then defined in the text box below using the x, y, z or t variables³. Based on the user selection the routine applies the defined variable ranges to derive a new variable. In addition text inputs required by the call and the model object (`mobj`) can also be passed. Comments can be used to pass additional instructions, such as the inclusion of the `RowNames` in the output to be saved as a new dataset, using either `%time` or `%rows`.

For example, any of the following could be entered into the equation box:

`x.^2+y %time`

`myfunction1(x,y,t,'usertext')`



³ Various pre-defined function templates can be accessed using the 'Function' button. Alternatively, text can be pasted into the equation box from the clipboard by right clicking in the text box with the mouse.

```
myfunction2(x,mobj)
```

The output from function calls can be figures or tables, a single numeric value, or a dataset to be saved (character vector or array). External functions must return output as a cell array with the new variable in the first cell and data to be used to define RowNames in the second cell. If the %time or %rows instruction is included in the call, row data are added providing that the length of the input dataset matches the output dataset. If there is no output to be passed back the function should return a cell array containing the string 'no output' to suppress the message box, which is used for single value outputs. For expressions that return a result that is the same length as one, or more, of the variables used in the call, there is also the option to add the variable to one of the input datasets as a new variable. In all there are three ways in which results can be saved:

1. Expression or function returns a result that is the same number of rows as one or more of the input datasets. Option to (a) add as a new variable to an existing data set, or (b) create a new dataset with no assignment to the RowNames property.
2. As (1) with the comment of %time or %rows in the call. Attempts to use the RowNames property of one of the inputs to define RowNames in a new dataset. Requires the input variables to have same number of rows. (Plan to add interpolation so this may change).
3. Expression or function returns a result with a new variable and time time in a 2-element cell array. The variable is saved as a new dataset.

An alternative when calling external functions is to pass the selected variables as dstables, thereby also passing all the associated metadata and RowNames for each dataset selected. For this option up to 3 variables can be selected but they are defined in the call using dst, for example:

```
myfunction3(dst,'usertext',mobj)
```

This passes the selected variables as a struct array of dstables to the function. Using this syntax the function can return a dstable, or struct of dstables, or a cell array containing one or more data sets. The options for saving the data are the same, with the additional option that when a dstable, or struct of dstables, is returned, these are saved directly and it is assumed that the dsproperties have been defined in the function called.

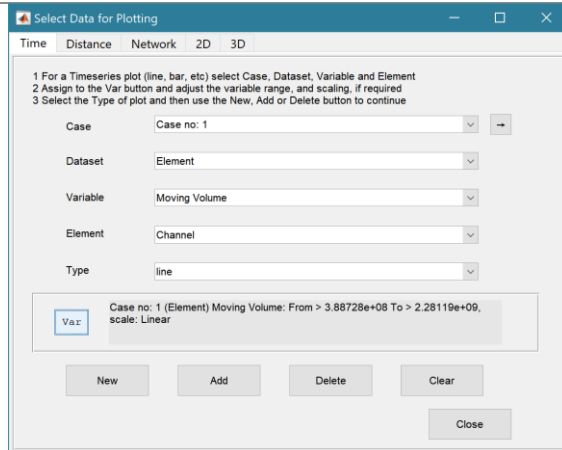
Some further details on using this option and the '**Function**' library available are provided in Section 5.11.2.

4.6 Analysis

Plotting and Statistical Analysis both use the standard Data selection UI. These both require Case, Dataset and Variables to be selected from drop-down lists and assigned to a button. Further details of how this works are given in Section 4.9.

4.6.1 Plotting

Analysis>Plot menu: initialises the Plot UI to select variables and produce several types of plot. The user selects the Case, Dataset, and Variable to be used and the plot Type from a series of drop-down lists. There are then buttons to create a New figure, or Add, or Delete variables from an existing figure for 2D plots, or simply a Select button for 3D plots. The following figures illustrate the options available.



Time

For each selection choose the Case, Dataset, Variable and Element to be used.

> Assign variable to the Var button.

Each selection can be scaled (log, normalised, etc) and the range to be plotted can be adjusted when assigning the selection to a button.

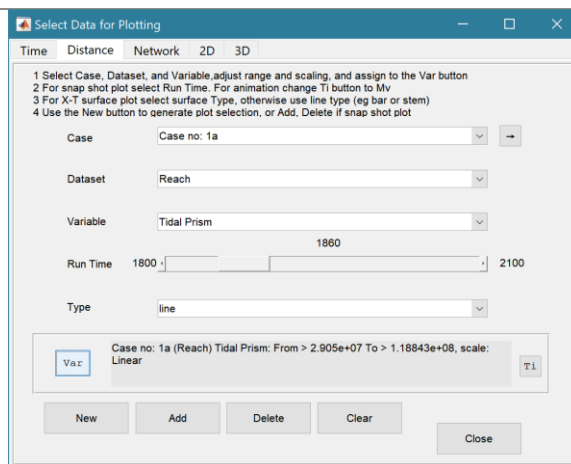
> Select plot type (line, bar, scatter, stem, etc)

> Use the New, Add and Delete buttons to generate and modify the plot.

Control Buttons:

→ : updates the list of Cases.

Pivot year button on figure – see below



Distance

For each selection choose the Case, Dataset and Variable to be used.

> Assign a variable to the Var button.

Each selection can be scaled (log, normalised, etc) and the range to be plotted can be adjusted when assigning the selection to a button.

> Set the required Run Time, or press 'Ti' button to switch to an animation for the duration of the run.

> Select plot type as bar, scatter, stem, etc for snapshot or animation and surf, contour, etc for X-T plot.

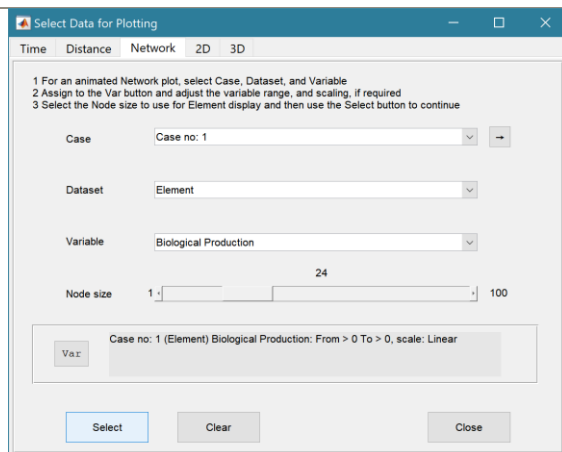
> Use the New/Add/Delete buttons to generate plot.

Control Buttons:

→ : updates the list of Cases.

Ti/Mv : switch between Snapshot and Animation

Note: If there are only a few reaches in the model, the plot works better using bar or step plot types.



Network

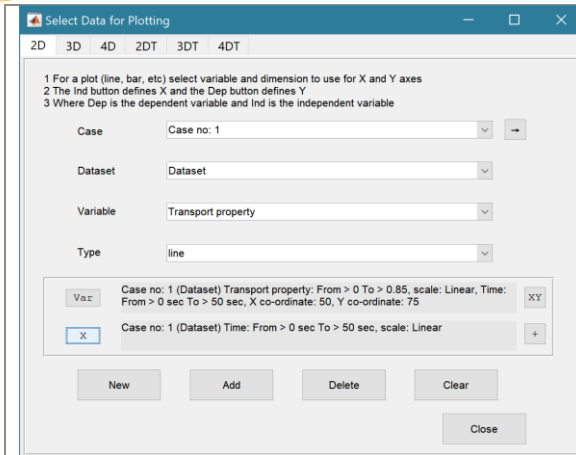
For each selection choose the Case, Dataset and Variable to be used.

> Assign a variable to the Var button.

Each selection can be scaled (log, normalised, etc) and the range to be plotted can be adjusted when assigning the selection to a button.

> Set the node size. This defines the size of the circle used to depict each Element or Reach, depending on the variable selected.

> Use the select button to generate the animated plot of the network graph (colour map used to show changes in selected variable).



2D plot

For each selection choose the Case, Dataset and Variable to be used.

> Assign a variable, or a dimension, to the Var and X buttons to set the Y and X axes, respectively.

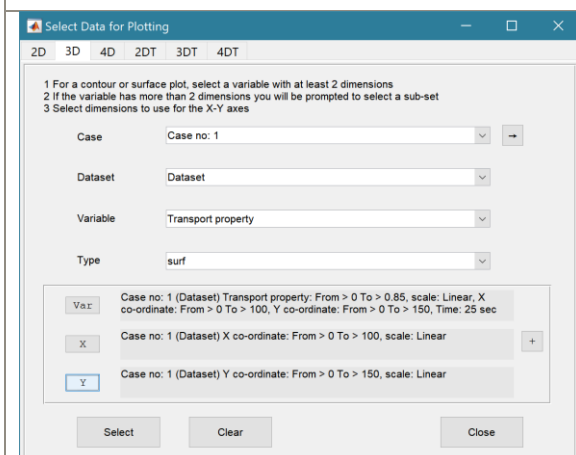
Each selection can be scaled (log, normalised, etc) and the range to be plotted can be adjusted when assigning the selection to a button.

> Select plot type (line, bar, scatter, stem, etc)

Control Buttons:

→ : updates the list of Cases.

XY: swaps the X and Y axes



3D plot

For each selection choose the Case, Dataset and Variable to be used.

> Assign selections to the Var, X and Y buttons

Take care to ensure that the assignments to X and Y correctly match the dimensions selected for the variable (including any adjustment of the dimension ranges to be used).

> Select plot type.

Control Buttons: see 2D plot above.

For all plot types, when the data has more dimensions than the plot or animation the user is prompted to sub-select from the data (by selecting sampling values for the dimensions that are not being used).

Selection of User plot type

Calls the user_plot.m function, where the user can define a workflow, accessing data and functions already provided by the particular App or the muitoolbox. The sample code can be found in the muitoolbox/psfunctions folder and illustrates the workflow to a simple line plot using x-y data from the 2D tab and a surface plot using x-y-z data from the 3D tab.

Time plot 'pivot' year button

When timeseries plots are generated using the Time tab option, a 'pivot' button is added to the figure that is generated. This allows the user to adjust the y-values for all traces on the plot, to an initial value at a defined point in time (the pivot year). The user is prompted set these values along with the x-axis limits for the new plot and new figure is created.

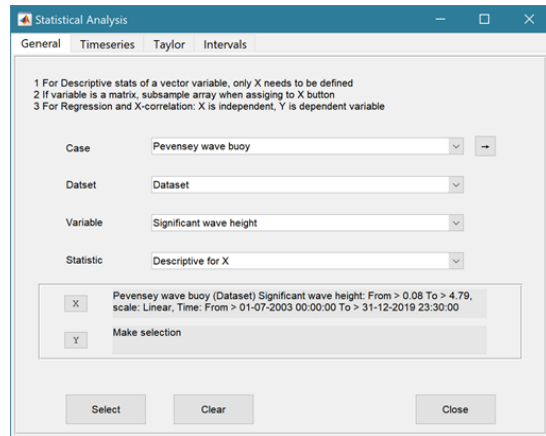
4.6.2 Statistics

Analysis> Statistics: several statistical analysis options have been included within the Statistical Analysis GUI. The tabs are for *General* statistics, and model comparisons using a *Taylor* Plot.

General tab

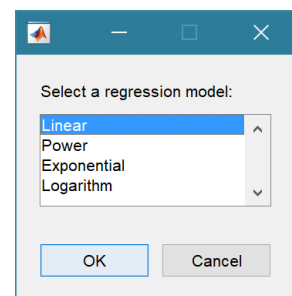
The General tab allows the user to apply the following statistics to data loaded in ModelUI:

- 1) **Descriptive for X:** general statistics of a variable (mean, standard deviation, minimum, maximum,



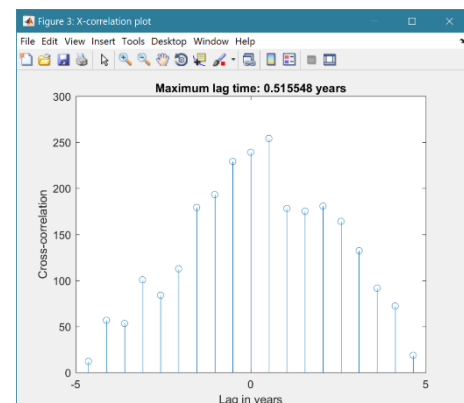
sum and linear regression fit parameters). Only X needs to be defined. The range of the variable can be adjusted when it is assigned to the X button (see Section 4.9). If the variable being used is a multi-dimensional matrix (>2D), the user is prompted to define the range or each additional dimension, or select a value at which to sample. The function can return statistics for a vector or a 2D array.

The results are tabulated on the **Stats>General** tab and can be copied to the clipboard for use in other applications.



- 2) **Regression:** generates a regression plot of the dependent variable, Y, against the independent variable, X. For time series data, the default data range is the maximum period of overlap of the two records. For other data types the two variables must have the same number of data points. After pressing the Select button, the user is prompted to select the type of model to be used for the regression. The results are output as a plot with details of the regression fit in the plot title.

- 3) **Cross-correlation:** generates a cross-correlation plot of the reference variable, X, and the lagged variable, X (uses the Matlab 'xcorr' function). For time series data, the default data range is the maximum period of overlap of the two records. For other data types the two variables must have the same number of data points. This produces a plot of the cross-correlation as a function of the lag in units selected by the user.



- 4) **User:** calls the function user_stats.m, in which the user can implement their own analysis methods and display results in the UI or add output to the project Catalogue. Currently implements an analysis of clusters as detailed for Timeseries data below.

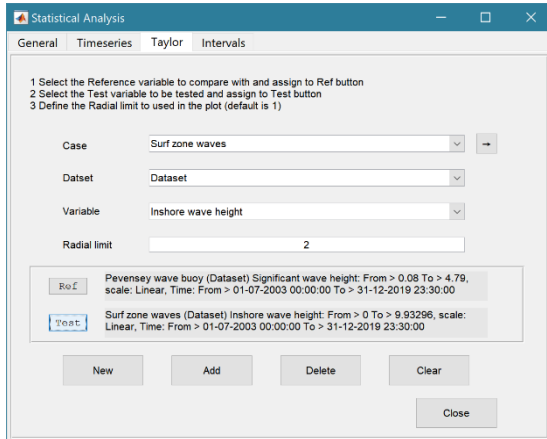
Taylor tab

The Taylor tab allows the user to create a Taylor Plot using 1D or 2D data (e.g timeseries or grids):

A Reference dataset and a Test dataset are selected. Datasets need to be the same length if 1D, or same size if 2D. If the data are timeseries they are clipped to a time-period that is common to both, or any user defined interval that lies within this clipped period. The statistics (mean, standard deviation, correlation coefficient and centred root mean square error) are computed, normalized using the reference standard deviation and plotted on a polar Taylor diagram (Taylor, 2001).

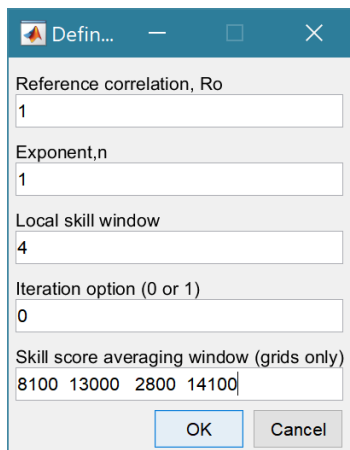
[The ModelSkill App provides additional tools to test data and the ModelSkill App manual provides further details of the methods used.]

Selecting New generates a new Taylor Plot. Selecting the Add button adds the current selection to an existing plot and the Delete button deletes the current selection. The Clear button resets the UI to a blank selection.



Once New or Add are selected, the user is asked whether they want to plot the skill score (Yes/No). If Yes, then the user is prompted to set the skill score parameters. As further points are added to the plot, this selection remains unchanged (i.e. the skill score is or is not included). To reset the option it is necessary to close and reopen the Statistics UI.

If the number of points in the Reference and Test datasets are not the same the user is prompted to select which of the two to use for interpolation.



This is the maximum achievable correlation (see Taylor (2001) for discussion of how this is used).

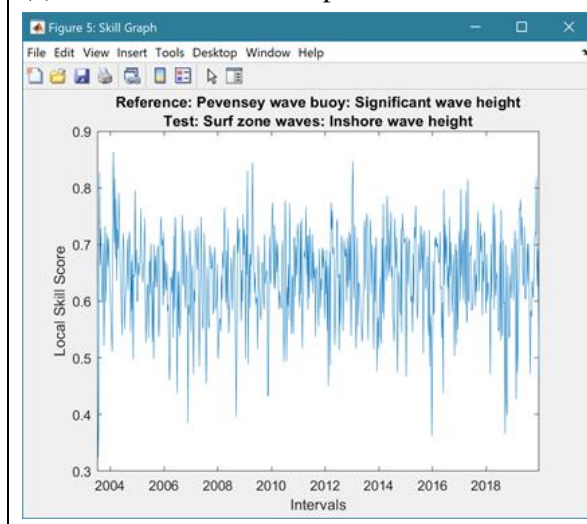
Exponent used in computing the skill score

Number of points ($\pm W$) used to define a local window around the i th point. If $W=0$ (default) the local skill score is not computed.

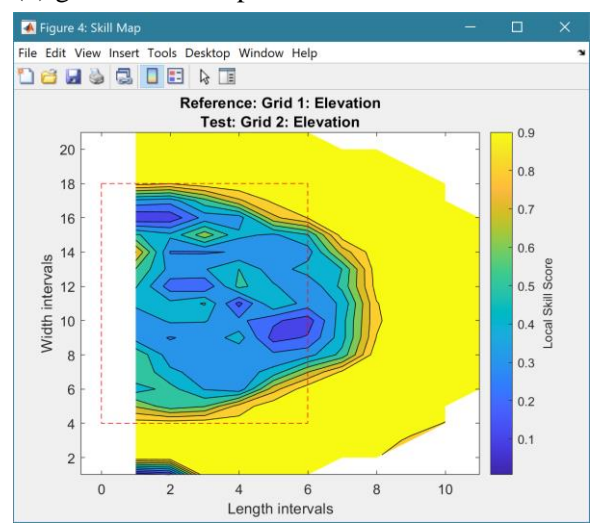
Local skill score is computed for window around every grid cell ($=1$), or computes score for all non-overlapping windows ($=0$)

Window definition to sub-sample grid for the computation of the average **local** skill score. Format is [xMin, xMax, yMin, yMax].

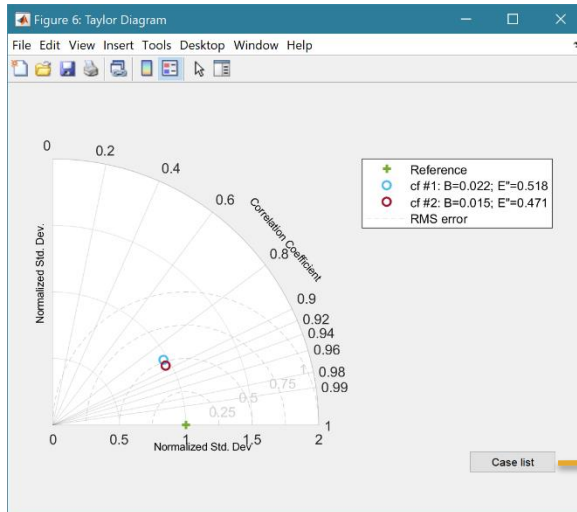
(a) time series skill score plot



(b) grid skill score plot

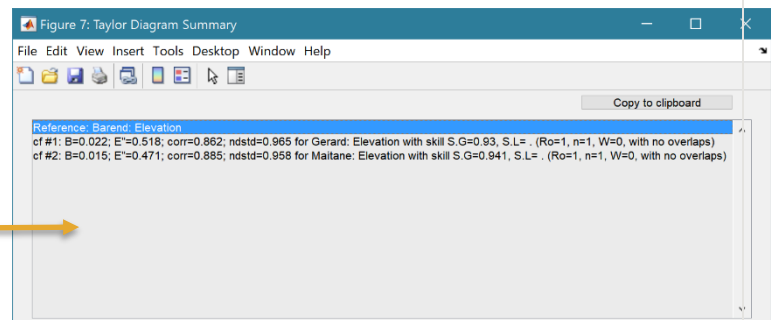


The Taylor Plot shows the Reference point as a green cross and the Test points as coloured circles. The legend details the summary statistics and the Case List button generate a table figure listing all the results. These can be copied to the clipboard.



Taylor diagram legend includes: B – bias; E'' – normalised RMS difference

The normalised standard deviation and correlation coefficient are also given in the Case List table, along with the global skill score, Sg, and the average local skill score, Sl.



4.7 Help

The options from the help menu include:

Help>Documentation - initialises the App documentation in the Matlab™ Supplemental Software documentation.

Help>Manual – access the Asmita Manual (this document).

Help>Theory 1 – access the paper: Development and extension of an aggregated scale model: Part 1 – Background to ASMITA. China Ocean Engineering, 30(4), 482-504.

Help>Theory 2 – access the paper: Development and extension of an aggregated scale model: Part 2 – Extensions to ASMITA. China Ocean Engineering, 30(5), 651-670.

4.8 Tabs

To examine what has been set-up the Tabs provide a summary of what is currently defined. Note: the tabs update when clicked on using a mouse but values displayed cannot be edited from the Tabs.

Cases: lists the cases that have been run with a case id and description. Clicking on the first column of a row generates a table figure with details of the variables for the case and any associated metadata. Buttons on the figure provide access the class definition metadata and any source information (files input or models used).

Settings tab includes the following sub-tabs:

System: tabulates the system parameters and flags that have been set (display only).

Elements: tabulates the key parameters of each element (display only).

Saltmarsh: tabulates the key parameters of each element (display only) and plots the species biomass distribution.

Interventions: plots the volume and surface area changes defined for individual elements.

Run Parameters: tabulates the key run conditions and equilibrium coefficients (display only).

Tides tab includes the following sub-tabs:

Network: graphic of the element network showing the connectivity and horizontal exchanges between all elements (m^3/s).

Water Levels: plot of the changes in water levels (HW, MSL, LW) over the run period.

Hydraulics: plot of the along-channel variation in water levels and velocities if the CSTmodel is used to define channel hydraulics

TP Discharge: plot of tidal pumping discharge as a function of distance from the mouth

TP Network: graphic of the elements that have a tidal pumping advection, showing the connectivity and the flow rate (m^3/s).

Flows tab includes the following sub-tabs:

Rivers: graphic of the elements that have a riverine advection, showing the connectivity and the flow rate (m^3/s).

Drift: graphic of the elements that have a riverine advection, showing the connectivity and the flow rate (m^3/s).

Input Summary: displays what river and drift inputs are currently defined

River Input: plots the river discharge and sediment load if a timeseries data set has been loaded.

Drift Input: plots the drift input(s) if timeseries data has been loaded.

Response: summary of morphological response times based on system definition.

Q-Plot: displays a quick-plot defined for the class of the selected case (display only).

Stats: displays a table of results for any analyses that have been run (can be copied to clip board).

When using the **Network**, **TP Network**, **River**, and **Drift** tabs, there are two buttons in the top left corner of the tab. The 'Pan' button enables the option to move the graph in the tab window. Right clicking the mouse provides options to zoom or reset the plot – 'Pan' button toggles to 'Zoom'. Clicking on the 'Zoom' button when in zoom mode returns the setting to pan mode. When in use for pan or zoom, the second button is set to 'On'. Clicking on the 'On' button disables the pan and zoom options and resets the button to 'Off'.

NB: *All other tab actions are disabled when the pan button is in use. Ensure the 'On/Off' button id 'Off' when finished with the pan/zoom options, or before moving to another tab.*

When using the **Drift** tab, there is an additional button in the top right corner of the tab. When set to 'Qs' the drift rates as input are displayed. When set to 'Fs' the equivalent flow rates as used in the calculation of net exchanges are displayed. This allow the 'effective' influence of the Drift to be compared with other advection/dispersion exchanges (units are m/s).

4.9 UI Data Selection

Functions such as Derive Output (4.5), Plotting (4.6.1) and Statistics (4.6.2) use a standardised UI for data selection. The Case, Dataset and Variable inputs allow a specific dataset to be selected from drop down lists. One each of these has been set to the desired selection the choice is assigned to a button. The button varies with application and may be X, Y, Z, or Dependent and Independent, or Reference and Sample, etc. Assigning to the button enables further sub-sampling to be defined if required. Where an application requires a specific number of dimensions (eg a 2D plot), then selections that are not already vectors will need to be subsampled. At the same time, the range of a selected variable can be adjusted so that a contiguous window within the full record can be extracted. In most applications, any

scaling that can be applied to the variable (e.g., linear, log, relative, scaled, normalised, differences) is also selected on this UI. The selection is defined in two steps:

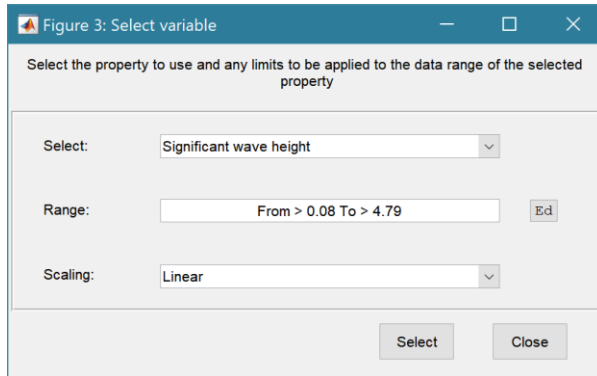


Figure 3: Select variable

Select the property to use and any limits to be applied to the data range of the selected property

Select: Significant wave height

Range: From > 0.08 To > 4.79 Ed

Scaling: Linear

Select Close

Step 1.

Select the attribute to use. This can be the variable or any of its associated dimensions, which are listed in the drop-down list.

The range for the selection can be adjusted by editing the text box or using the Edit (Ed) button.

Any scaling to be applied is selected from the drop-down list.

Press Select to go to the next step or Close to quit.

The number of variables listed on the UI depends on the dimensions of the selected variable. For each one Select the attribute to use and the range to be applied.

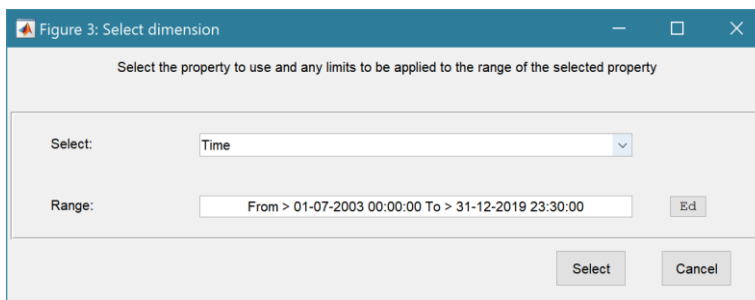


Figure 3: Select dimension

Select the property to use and any limits to be applied to the range of the selected property

Select: Time

Range: From > 01-07-2003 00:00:00 To > 31-12-2019 23:30:00 Ed

Select Cancel

Step 2 - Variable only has dimension of time.

No selection to be made.

Edit range if required.

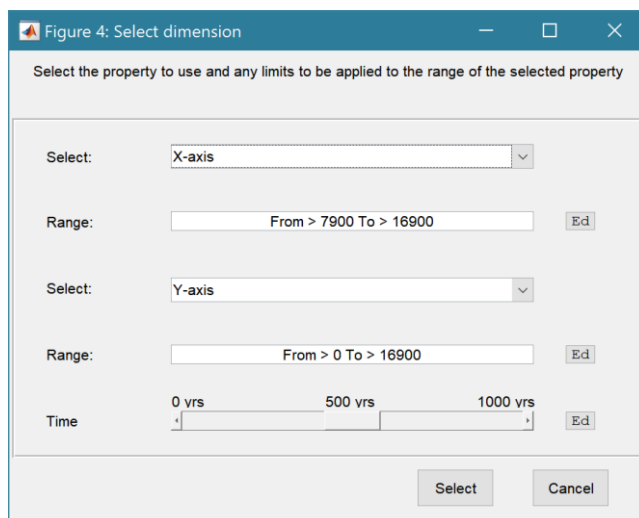


Figure 4: Select dimension

Select the property to use and any limits to be applied to the range of the selected property

Select: X-axis

Range: From > 7900 To > 16900 Ed

Select: Y-axis

Range: From > 0 To > 16900 Ed

Time 0 yrs 500 yrs 1000 yrs Ed

Select Cancel

Step 2 - Variable has 3 dimensions but only 2 are needed for the intended use.

Select the 1st variable to use as a dimension.

Edit range if required.

Select the 2nd variable to use as a dimension.

Edit range if required.

Use the slider or the Edit (Ed) button to set the value of the dimension to use. (A value of t=500 is selected in the example shown).

Press Select to accept the selection made.

[NB: Only unused dimensions can be selected from the Select drop-down lists. To adjust from the default list this can sometimes require that the second Select list-box is set first to allow the first Select list-box to be set to the desired value.]

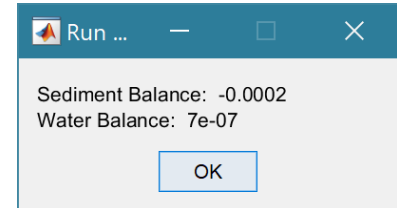
The resulting selection is then detailed in full (including the ranges or values to be applied to all dimensions) in the text box alongside the button being defined.

Note where a variable is being selected as one property and a dimension as a second property, any sub-selection of range must be consistent in the two selections. This is done to allow variables and dimensions to be used as flexibly as possible.

4.10 Model outputs

At the end of a model run the sediment and water balances are displayed in a message window. The water balance considers only elements that are within the estuary, i.e., channel, tidal flats, saltmarsh and storage element. If tidal flats are defined as sediment volumes, the flats are excluded from the water balance.

Each model run is saved as a Case and listed on the *Cases* tab. The results can be exported to an Excel spreadsheet along with the model settings using the *Project>Scenarios>Save* option. The output includes the following values for each element in the model:



Variable	Units	Description
Moving volume	m ³	Volume of the element including any changes due to water level variations (e.g., due to slr and tidal range).
Fixed volume	m ³	Morphological volume of the element (excludes water volume change)
Equilibrium volume	m ³	Target volume defined by the specified equilibrium relationship
Moving depth	m	Effective total depth of element (Vm/Sm) ¹
Fixed depth	m	Effective morphological depth of element (Vf/Sf) ²
Equilibrium depth	m	Effective equilibrium depth of element (Ve/Se) ²
Biological production	m ³	Volume of organic matter produced in the element
Concentration	ppm	Sediment concentration in the element
Surface area	m ²	Surface area of the element
Reach prism	m ³	Volume of the tidal prism within each reach (channel element)
Tidal prism	m ³	Cumulative tidal prism for all channel elements upstream ³
Upstream CSA	m ²	Cross-sectional area of the upstream interface of a channel element
River flow	m ³ /s	River discharge at the upstream interface of a channel element
Mean sea level	mAD	Mean sea level in each element
High water level	mAD	High water level in each element
Low water level	mAD	Low water level in each element
Tidal range	m	Tidal range (2.amplitude) including any tidal cycles

mAD – metres above datum. 1) depth of element is a water depth if the transport coefficient is positive (n>0) and a sediment depth if negative (n<0). 2) For fixed area model: Sm = Sf = Se. 3) includes reach prism of the element

4.11 Interpretation when using water and sediment volumes ($\pm n$)

The fixed volume is given by: $V_f(t) = V_f(t-1) + \Delta V_f$

and the moving volume is given by: $V_m(t) = V_m(t-1) + \Delta V_f + \Delta V_m$

where $\Delta V_f = \frac{1}{c_b} (\mathbf{B}\underline{\gamma} - \underline{d}) \cdot \Delta t$, $\Delta V_m = \text{sign}(n) \cdot \Delta s_{wl} \cdot S$ and $\gamma = (V_e/V_m)^n$

see Townend et al (2016a) for definition of terms. The sum of ΔV_f and ΔV_m is the change in accommodation space, i.e.: $\Delta V_A = \text{sign}(n) \cdot (\Delta V_f + \Delta V_m)$

For an unperturbed system, where $V_m = V_f = V_e$, the accommodation space is zero. Once perturbed some form of accommodation space is created. In the case of linear sea level rise, this takes the form of a constant over-depth, such that $\Delta V_A = 0$ and $\Delta V_f = \Delta V_m$, once the initial perturbation has been accommodated (i.e. after model spin up).

4.11.1 Elements using water volumes

The changes to the fixed volume, V_f , are due to sediment exchange, ignoring any changes in the water level. It can be thought of as the change in the water volume below a fixed initial surface (such as high water), or the morphological change. In contrast, the moving volume, V_m , includes both the morphological changes and the water volume changes, and is equivalent to the total change in the water volume. Another way of thinking of this is that positive changes of the water volume, $+\Delta V_m$, increases the accommodation space, whereas the import of sediment into the element ($-\Delta V_f$) reduces it. Hence for a linear sea level rise the typical behaviour is for the fixed volume to reduce linearly and the moving volume to initially increase to create an excess water volume (model spin up in response to the imposition of the sea level perturbation) and then, if there is sufficient sediment supply, to be approximately constant. The adjustment of the moving volume, relative to the initial volume, implies the development of a constant over-depth (because the area of each element is constant). Any excess accommodation space due sea level rise is infilled albeit with a lag that reflects the morphological response time (Townend et al., 2007).

4.11.2 Elements using sediment volumes

Change in the fixed volume, V_f , reflect import or export of sediment. The moving volume, V_m , changes relative to the initial volume to reflect changes in sediment volume and any changes in the water volume. In this case it is only meaningful to think of this as changes in the accommodation space, relative to some initial volume. Under rising sea levels, changes in the moving volume, ΔV_m , are negative (because of the change in sign of n) and, if the element infills, there is a positive change in the fixed volume, $+\Delta V_f$. Thus, as with water volume elements, an increase in water level increases the accommodation space and import of sediment reduces it. Hence, in an infilling element the fixed volume, V_f , increases over time, whereas the moving volume initially reduces and then becomes approximately constant. This again reflects the creation of a constant excess water volume, or over-depth, once the initial excess accommodation space, caused by the imposition of sea level rise, is infilled.

4.11.3 All Elements output

The Element output table includes two columns for ‘All’ elements. One column gives the output for water elements ($n>0$) and the other is for sediment elements ($n<0$). The output given is dependent on the type of variable. For Volumes, this is the sum of all elements, e.g., $V_{all} = \sum(V(n<0))$, is the sediment volumes for elements such as deltas and tidal flats (if $n<0$). For other variables in the Element table the mean of the element values is used. For concentrations and water levels this is consistent. However, for depths this needs to be interpreted with caution. If all the elements are water volumes this is the mean depth of the elements but NOT necessarily the mean depth of the inlet or estuary.

4.12 Accessing data from the Command Window

In addition to the options to save or export data provided by the *Project>Cases>Save* and *Project>Import/Export* options, data can also be accessed directly for use in Matlab™, or to copy to other software packages. This requires use of the Command Window in Matlab™, and a handle to the App being used. To get a handle, open the App from the Command Window as follows:

```
>> myapp = <AppName>;           e.g., >> as = Asmita;
```

Simply typing:

```
>> myapp
```

Which displays the results shown in the left column below with an explanation of each data type in the right hand column.

myapp = <AppName> with properties:	Purpose
Inputs: [1×1 struct]	A struct with field names that match all the model parameter input fields currently
Cases: [1×1 muiCatalogue]	muuiCatalogue class with properties DataSets and Catalogue. The former holds the data the latter the details of the currently held records.
Info: [1×1 muiProject]	muiProject class with current project information such as file and path name.
Constants: [1×1 muiConstants]	muiConstants class with generic model properties (e.g. gravity, etc).

To access current model settings, use the following:

```
>> myapp.Inputs.<InputClassName>
```

To access the listing of current data sets, use:

```
>> myapp.Cases.Catalogue
```

To access imported or model data sets, use:

```
>> myapp.Cases.DataSets.<DataClassName>
```

If there are more than one instance of the model output, it is necessary to specify an index. This then provides access to all the properties held by that data set. Two of these may be of particular interest, RunParam and Data. The former holds the input parameters used for that specific model run.

RunParam is a struct with fields that are the class names required to run the model (similar to Inputs above). The Data property is a model specific struct with field names defined in the code for the model class. If there is only a single table assigned this will be given the field name of 'Dataset'. To access the *dstable* created by the model, use:

```
>> myapp.Cases.DataSets.<DataClassName>(idx).Data.Dataset
```

```
>> myapp.Cases.DataSets.<DataClassName>(idx).Data.<ModelSpecificName>
```

To access the underlying *table*, use:

```
>> myapp.Cases.DataSets.<DataClassName>(idx).Data.Dataset.DataTable
```

The result can be assigned to new variables as required. Note that when assigning *dstable*s it may be necessary to explicitly use the copy command to avoid creating a handle to the existing instance and potentially corrupting the existing data.

5 Supporting Information for Model Setup

5.1 Model elements

Decide how the estuary or tidal inlet should be sub-divided. This will usually involve a number of channel elements with associated tidal flat elements. In addition, there may be a need to include a tidal delta and any tidal flats associated with the delta. As far as possible it is worth trying to examine the constraints on the system (geology, sediment sources, sea walls, etc), as a basis for identifying “reaches” that are likely to have a consistent internal behaviour. It is also better to avoid very large differences in the size of elements (several orders of magnitude), as this can give problems in some of the matrix calculations.

Sometimes it is useful to set up a whole estuary model (comprising just a channel and tidal flats, or delta, channel and tidal flats) to get a feel for the overall behaviour of the system, prior to developing a more detailed representation. However, if river inflows with a sediment load are to be included, it should be noted that the response of a “single” channel element model, will be different to a multi-element model. Consequently, it is sometimes better to ignore the river inflows for the single channel element case.

The Element types currently available include:

Channel	Water volume below MLW and area at MLW. <i>Can be linked to Outside, delta, tidalflat, storage and other channel elements.</i>
Tidalflat	Water volume between MLW and MHW over flats (i.e., not including the prism over the channel) and area over flats. Flats on both sides of channel in each reach can be included in a single flat element. This element can also be defined as a sediment volume by defining a negative value of Transport coefficient in <i>Setup>Element>Element Parameters</i> . <i>Can be linked to any element within the inlet or estuary.</i>
Saltmarsh	Water volume over the marsh and associated area. Saltmarsh on both sides of a channel in each reach can be included in a single saltmarsh element. Can also be used to represent an upper tidal flat by not setting the Include Saltmarsh condition in <i>Setup>Run Parameters>Conditions</i> . <i>Can be linked to any element within the inlet or estuary.</i>
Storage	Water volume and area of storage unit. Treated as locally independent of system (not dependent on upstream prism) but contributing to downstream prism. <i>Can be linked to any element within the inlet or estuary but should not have any ‘upstream’ elements.</i>
Ebb Delta	Sediment volume of the delta (where the delta has a volume relative to the undisturbed coastal bed). A negative value of Transport coefficient should be defined in <i>Setup>Element>Element Parameters</i> . <i>Can be linked to channel, Outside and other delta elements.</i>
Flood Delta	Sediment volume of the delta (where the delta has a volume relative to the undisturbed coastal bed). A negative value of Transport coefficient should be defined in <i>Setup>Element>Element Parameters</i> . <i>Should be linked to a Delta Channel. See example in Section 3.5.</i>
Delta Flat	Water volume between MLW and MHW over flats (i.e., not including the prism over the channel) and area over flats. Flats on both sides of channel in each reach can be included in a single flat element. This element can also be defined as a

	sediment volume by defining a negative value of Transport coefficient in <i>Setup>Element>Element Parameters</i> . <i>Should be linked to a Delta Channel. See example in Section 3.5.</i>
Beach Face	Water volume between MLW and MHW and area over beach face. This element can also be defined as a sediment volume by defining a negative value of Transport coefficient in <i>Setup>Element>Element Parameters</i> . <i>Can be linked to Spit, Shore Face, delta elements and Channel elements.</i>
Shore Face	Water volume below MLW and area over shore face. This element can also be defined as a sediment volume relative to some arbitrary datum with a negative value of Transport coefficient in <i>Setup>Element>Element Parameters</i> . <i>Can be linked to Beach Face, Spit, delta elements and Channel elements.</i>
Spit	Sediment volume of spit (where the spit has a volume relative to the undisturbed coastal bed, typically MLW). A negative value of Transport coefficient should be defined in <i>Setup>Element>Element Parameters</i> . <i>Can be linked to Beach Face, Shore Face, delta elements and Channel elements.</i>

The linkages noted above refer to the network connectivity defined by the dispersion matrix of horizontal exchanges. Advective flows can be defined between any set of elements, subject only to any conditions imposed on the flow type (e.g., mass balance for river flows).

Using a suitable DGM the volumes and areas can be calculated in one of two ways:

- (i) Define a horizontal plane at high and low water level and calculate the water volumes below the plane and the surface areas on the planes.
- (ii) Use the surfaces for high and low water derived from a hydraulic model in conjunction with a DGM to calculate the water volumes below the surface and the surface areas at high and low water. This added complexity is not needed for small or relatively short estuaries but can be important for longer estuaries that are near resonant (Thames, Humber, Severn, etc).

It is also possible to define elements as sediment volumes rather than wet volumes. If this is preferred, then the sediment transport coefficient, n , should be expressed as a negative value. This was the basis for the original model development which used fixed surface areas (Kragtwijk et al., 2004; Stive et al., 1998) and works for this form of configuration.

The bulk density of sediment relates to the density of the bed within the element. This is included to take account of the consolidation of sediment that is deposited and is used in the code to adjust the vertical rate of exchange. The default value of 1350 kg.m^{-3} is typical for tidal flats in many UK estuaries with a higher value of around 1650 kg.m^{-3} in the channels.

The sediment transport coefficient reflects the nature of the transport process. For simple diffusion a value of 2 can be adopted. Where bed or suspended transport dominate a value between 3 and 5 may be more appropriate. As this model is considering long-term evolution it can be argued that the process can be treated as essentially diffusive, suggesting a value of 2. Furthermore, it is the product of n and c_E that determines the morphological response time, so that using a set value may simply result in a slightly different value of c_E .

The vertical exchange (although not strictly a sediment fall velocity) should be based on the fall velocity for the sediment type that characterises most of the element. There is an inter-dependence between the fall velocity and the horizontal exchange which is explained further in the Section on Dispersion.

5.1.1 File format for Element Parameters

The option to load the element parameters, using *Setup>Element>Load Parameters*, requires a text file with a header and data entered in space or tab separated columns, as follows:

ID	Volume	SurfaceArea	Length	VerticalExchange	TransportCoeff	BedDensity	SedimentMob	Damping
1	1.08E+09	1.87E+08	4000	0.003	-3	1550	1	1
2	3.94E+08	7.49E+07	134000	0.003	3	1550	1	0.95
3	1.40E+07	3.14E+07	134000	0.003	3	1350	1	0.92

The header is used to define the number of columns and text separation must ALL use ‘space’ or ‘tab’. Note, however, the column order of the data is not checked. The columns, in order, are element ID, volume (m³), surface area (m²), length of element (m) vertical exchange (m/s), transport coefficient (negative is sediment volumes are being used), bed density (kg/m³), sediment mobility (-) and damping coefficient (-). The number of rows must match the number of elements already defined using *Setup>Element>Define Elements* and the element ID is used to match the parameters to the correct element (the ID follows the order listed on the *Elements* tab, or as shown for the nodes IDs on the *Network* tab). Use *Setup>Element>Define Elements* to view the element Type and Name. To change element definitions, use *Setup>Element>Load Parameters*, to overwrite the existing definitions.

5.2 Dispersion

5.2.1 Exchange between channel elements (and channel and delta)

The horizontal diffusion coefficient is related to the vertical exchange by the relationship:

$$\frac{D}{uh} = \varepsilon \frac{u}{w_s} \quad (1)$$

where u is the velocity (at mtl or peak) between two channel elements, h is the hydraulic depth (assumed to be at mtl) and w_s is the vertical exchange. The basis of this equality is explained in Wang *et al.* (2007), where the parameter ε is a coefficient of proportionality, which is suggested as a calibration coefficient of order 0.1. Given the uncertainty in defining the horizontal exchange coefficient, as set out below, the coefficient is taken to be unity and accounted for by simply adjusting the horizontal exchange values directly.

For use within ASMITA, this has to be expressed as a horizontal rate of exchange between elements, given by:

$$\delta = D.A/L_x \quad (2)$$

where A is the cross-sectional area between the elements and L_x is the distance between the two elements. However, this is rather dependent on how the elements happen to be defined and a better way of determining L_x is to use the tidal excursion distance given by:

$$L_x = u_p.T_p/\pi \quad (3)$$

where u_p is the peak velocity and T_p is the tidal period in seconds and simply assumes a sinusoidal variation in velocity over the tidal cycle. Model output can be used to define the excursion, if available.

5.2.2 Exchange between channel and flats

Three methods have been used to-date for estimating the exchange between channel and flats.

- (i) Use the channel estimate of δ , equation (2), and replace the c_{sa} , A , with the A_{int} , given by:

$$A_{int} = L_R \cdot tr / 2 \quad (4)$$

where L_R is the length of the element (reach) and tr is the tidal range. The half makes this an average value based on the variation between high and low water.

- (ii) Use equation (1) with the hydraulic depth based on the values for the tidal flat (note: this can be estimated using $h_{fl} = V_{fl}/S_{fl}$). Then obtain δ from equation (2), with the value of A defined by A_{int} as given in equation (4).

- (iii) Use the rate of propagation across the tidal flat to estimate the transverse velocity:

$$u_{fl} = \pi \cdot W_{fl} / (n_{bk} \cdot T_P) \quad (5)$$

where W_{fl} is the average width of the tidal flats and can be estimated from S_{fl}/L_R and n_{bk} is the number of banks with flats associated with the width, W_{fl} . Then use equation (1) with h_{fl} and u_{fl} to obtain D_{fl} . The horizontal exchange onto the flat is then given by a modified form of equation (2):

$$\delta_{fl} = D_{fl} \cdot A_{int} / W_{fl} \quad (6)$$

Experience to-date suggests that the third method is the most suitable but further work is needed to establish which of these is the most appropriate.

5.3 Advection

The model provides for advective flows through the system. Necessarily the flow in and out of an element must sum to zero and the code checks for this. The use of this facility is typically to include one or more river flows into the system. If a flow is introduced into the most upstream (landward) channel element, then this must be propagated through each of the downstream channel elements to the open sea boundary. (For a single channel system, this means there should be the same flow in and out of the channel element).

Each inflow to the system can, however, have a different sediment load. This is specified as the concentration imported by advection in **Setup>River**, expressed as a concentration density (kg.m-3) and an estimate can usually be obtained from measured data of the riverine input.

It is important to recognise that the model is seeking to impose an equilibrium concentration over the system, as the basis for volumetric equilibrium. This is why the option to take account of the flow in the definition of the equilibrium condition can be included by checking the box in **Setup>Run Parameters>Conditions**. (The basis of the flow adjustment to the equilibrium condition is explained in Section 3.4 of Townend et al. (2016a). If this option is not checked the flow conditions are in effect a perturbation to the system. The magnitude of this influence can be explored by setting all other changes (forced changes, transgression, slr and tidal range) to zero and running the model with and without this box checked. When the 'River Flow offset' option is selected the element volumes should remain constant. When it is not selected the response will depend on the model configuration.

For a single (channel) element model a concentration associated with advective flow through the element, that is less than c_E , will give rise to net export from the system. This is because the advective flow out of the element is removing sediment at the concentration of c_E , whereas the inflow is only replacing this at the lower rate. The converse is obviously the case when the concentration of the inflow is larger, resulting in net import. It also follows that a value that is equal to c_E is equivalent to the no flow case.

For a multi-element definition of the channel, this is no longer the case, providing the dispersion between channel elements varies, and there are two or more channel elements, the element connected to the river inflow will adjust according to the exchange with the downstream element and this is

distinct from the exchange of the last element that is connected to the open sea environment. The system can therefore co-adjust to take account of the different concentrations at the respective boundaries.

5.3.1 River timeseries file format

The option to load a timeseries description of the river inputs, using *Setup>Rivers>River timeseries*, requires a text file with a header and data entered in space or tab separated columns, as follows:

Year	Discharge	Concentration
1930.250000	35000	4.96E-01
1930.333333	20990	2.60E-01
1930.416667	25200	3.28E-01
1930.500000	30417	4.16E-01
1930.583333	48045	7.39E-01
1930.666667	46194	7.03E-01
1930.750000	41147	6.08E-01
1930.833333	35452	5.04E-01
1930.916667	27043	3.58E-01
1931.000000	11643	1.24E-01
1931.083333	9293	9.33E-02

The date/time is defined as decimal years, discharge in m³/sec and concentration density in kg/m³. Inclusion of concentration is optional. If this is not defined, the model uses the time varying discharge with the concentration value defined for constant flow input (*Setup>Rivers>River inputs*).

The values used in the model are obtained by interpolating the timeseries provided using the Matlab timeseries 'resample' function at the time (decimal years) of the time step. To capture seasonal variations the timestep therefore needs to be at least as short as the interval of interest. *However, it is important to remember that ASMITA models long-term change and not short-term dynamics.*

5.3.2 Drift timeseries file format

The option to load a timeseries description of the littoral drift inputs, using *Setup>Drift>Drift Timeseries*, requires a text file with a header and data entered in space or tab separated columns. For a single source input, only the Date(years) and Drift_in columns are needed as follows (drift is in m³/year):

Date(years)	Drift_in
1990	24661
1991	17584
1992	14664
1993	20202
1994	14626
1995	18671
1996	23462

(Note: in this case the headers are for file readability and are not used when loading the data)

Where the drift is being defined along a pathway then the header uses the format shown below:

Date(years)	driftin(0,5)	drift(5,1)	drift(5,4)	drift(1,6)	drift(4,6)	driftout(6,0)
1990	24661	19731	4932	17758	4439	22197
1991	17584	14067	3516	12660	3165	15825
1992	14664	11731	2932	10558	2639	13197
1993	20202	16161	4040	14545	3636	18181
1994	14626	11700	2925	10530	2632	13163
1995	18671	14936	3734	13443	3360	16803
1996	23462	18769	4692	16892	4223	21115

The header comprises Date(years), and ‘drift’ columns. The text description is not used and is for text file readability but the subsequent numbers in parenthesis define the Element ID of the start and end nodes for the drift pathway (a zero is used for the outside). These must be defined as a comma separated pair in parenthesis in the start, end order, i.e. ‘sometext(i,j)’ for a drift from element ‘i’ to element ‘j’.

The values used in the model are obtained by interpolating the timeseries provided using the Matlab timeseries ‘resample’ function at the time (decimal years) of the time step. To capture seasonal variations the timestep therefore needs to be at least as short as the interval of interest.

If only one timeseries is included in the file, the drift rate is applied at all nodes on the flowpath, and any branches are apportioned based on the proportions defined in the Advection matrix. Where there are multiple timeseries then these are assigned to the specified link and any links without an a timeseries input use the drift rate defined for the link in the Advection matrix.

5.4 Specification of Equilibrium

The relationships used to define the equilibrium volume as a function of tidal prism are specified in userprismcoeffs.m and selected as explained in Section 4.4. These can be made to be estuary/inlet specific (e.g as for the Ameland case) or generic based on the literature.

The ‘Default’ set assigns all coefficients a value of 1. This implies that the equilibrium volume of each element type scales linearly with tidal prism and would usually be used by scaling these prism-based equilibrium volumes to the initial element volumes by checking the ‘Scale to initial values’ flag in *Setup>Run Parameters>Conditions*.

The ‘Generic’ sets are based on published values as discussed in Section 3.1 of Townend *et.al.* (2016a) as detailed in the following tables.

Generic – linear law

Element Type	α	β	Sources
Channel	0.42	1	Low water channel values for UK estuaries
Flat	0.16	1	Tidal flat values for UK estuaries

Generic – power law:

Element Type	α	β	Sources
Delta and Ebb Delta	0.005	1.23	Original data from Walton & Adams (1976) converted to metric. Regression suggests a slightly different value of α .
Channel	0.005	1.23	Low water channel values for UK estuaries
Flat	0.08	1.08	Tidal flat values for UK estuaries
Flood Delta	7000	0.37	Gulf coast values (Powell et al., 2006)

The values provided can be used for an initial setup and then edited to meet the specific model requirements or as part of the model calibration using *Setup>Run Parameters>Edit Eq. Coefficients*.

These and other sets can all be used in the manner described above, where the existing system is “assumed” to be close to an equilibrium condition and the equilibrium volumes are scaled by the ratio of the initial volume to the initial prism-based equilibrium volume, by checking the ‘Scale to initial values’ flag in *Setup>Run Parameters>Conditions*.

In addition to α and β , the file `userprismcoeffs.m` defines a variable ‘*eqtype*’. The value of *eqtype* determines whether the [tidal prism] (*eqtype*=1) or [plan area x tidal range] (*eqtype*=0) is used when computing the equilibrium volume. The default is for all element types to have a value of 1. The formula for equilibrium volume is the same in both cases ($\alpha \cdot x^\beta$) and the values of α and β need to be modified to reflect the use of tidal prism, or tidal range. The default values of α and β for all element types is 1. Note that by setting $\beta = 0$, the equilibrium volume is set to a constant value of α .

Equilibrium for Spit, Beach Face, Shore Face and Delta Flat elements can define equilibrium in various ways that depend on whether the drift is time varying and the value of *eqtype*.

- (i) Constant drift and *eqtype* = 1, uses $(\alpha \cdot x^\beta)$, where x is the [tidal prism];
- (ii) Constant drift and *eqtype* = 0, uses $(\alpha \cdot x^\beta)$, where x is the [plan area x tidal range];
- (iii) Variable drift and *eqtype* = 1, uses $V_0 \cdot (1 + \alpha \cdot ((Q_{si} - Q_{s0}) / Q_{s0})^\beta)$, where V_0 is the initial volume, Q_{si} is the drift rate and Q_{s0} is the initial drift rate.
- (iv) Variable drift and *eqtype* = 0, uses $(\alpha \cdot x^\beta)$, where x is the [plan area x tidal range].

The default value of α and β for these element types is 1. The values of α and β need to be modified to reflect the chosen option and the model that best captures the beach dynamics.

The elements to be included in the calculation of the tidal prism are defined by the ‘REtypes’ property in Asmita and currently includes TidalFlat, Saltmarsh, Storage, FloodDelta and DeltaFlat element types.

5.5 Interventions

Changes introduced by developments on an estuary can be represented as perturbations to the system at a given time step, or over a defined time interval. This might be useful to represent a capital dredge and the subsequent maintenance dredging as a series of volume changes in the relevant channel elements. If the dredging impacts on the tidal flats then associated changes in the areas of both the tidal flat and channel elements need to be defined. Similarly, a reclamation could be defined as a loss of volume and area (usually on the tidal flats). The Volume and Surface Area are defined as changes in water volume and element plan area respectively, regardless of whether the element is defined in terms of water, or sediment volume (this is handled in the code).

Element plan areas are constant, so that changes in plan area due to interventions, produce a fixed change in area (i.e., each change produces a permanent change in area until the next change due to interventions or marsh edge erosion). Volumes are dynamic and each element can erode or accrete following a change in volume unless the non-erodible flag for the intervention is set to a value of 1. When the flag is zero, changes are made to the moving and fixed volumes but not the equilibrium volume, which may subsequently change due to any accompanying change in the tidal prism. When the flag is 1, the equilibrium volume is also modified by adjusting the dynamic values (based on prism or tidal range) for all subsequent time steps.

There is also a dependence on how equilibrium is defined. If a scaling is applied to scale equilibrium relative to initial conditions, this continues to be applied following the addition of any interventions. This then affects the equilibrium volume and the degree to which the system adjusts if the changes introduced alter the tidal prism. If only equilibrium coefficients are used (α and β) this issue is avoided.

5.5.1 File format for loading interventions

The option to load a timeseries description of interventions, using *Setup>Interventions>Load*, requires a text file with a header and data entered in space or tab separated columns, as follows:

The year is the year in which the intervention takes place (can be decimal), volume (m^3) is the change in volume (water or sediment depending on how the element is specified – see Section 5.1), area (m^2) is the surface plan area and non-erodible is a flag to indicate if the change is erodible, 0, or non-erodible, 1. The example shows changes in water volumes for a channel with dredging taking place annually and a large dredge taking place in 1931.

year	volume	area	non-erodible
1926	0	0	0
1927	420000	0	0
1928	420000	0	0
1929	420000	0	0
1930	420000	0	0
1931	965050	710743	0
1932	420000	0	0

5.5.2 Fixed Elements and Interventions

Each element can be specified as Erodible or non-Erodible (Section 4.4.2). Interventions can be transient or fixed, the former being able to adjust with the rest of the system, the latter being constrained. Other possible constraints include defining a marsh edge erosion rate, setting dynamic (time varying) exchanges between elements and making element active/inactive. This section details how these options are implemented and any interactions between them.

Non-Erodible

The element can accrete but not erode relative to a reference volume defined by the initial element volume. If any fixed interventions take place (non-erodible) these will alter the reference volume (i.e., initial volume + intervention). The vertical exchange is adjusted based on the sediment available relative to the reference volume. If there is no available sediment the vertical exchange is set to a very small value ($\epsilon(100)$). Hence, if the element has accreted, some erosion is allowed until all the surplus sediment (amount above reference volume) has been exhausted. Adjusting the vertical exchange, $ws \Rightarrow 0$, makes the diagonal matrix, \mathbf{W} , ill-conditioned. However, the full solution matrix is generally not ill-conditioned and this approach ensures a correct mass balance between elements.

Interventions

Interventions are imposed changes in volume and/or area that can be transient or fixed. How these are handled depends on the model configuration. When the changes are introduced, they modify the moving and fixed volumes, the surface area and the equilibrium surface area (+ve or -ve change). Thereafter the elements will adjust based on the defined equilibrium conditions for the element. In this version of the model, there is no basis for an equilibrium area. It therefore makes no sense to retain the initial area as the “equilibrium” condition. Consequently, the area is adjusted, to give a new surface area, that then remains constant until any further intervention. Such changes may alter the tidal prism (e.g., for a tidal flat with equilibrium defined in terms of area and tidal range), which will in turn influence elements that use the prism to define equilibrium. Any interventions that change the volume are applied to the current element volume but are not applied to the equilibrium volume. The unadjusted equilibrium volume is then the basis of determining subsequent adjustments (i.e., based on $\gamma = (V_e/V_m)^{-n}$).

The fixed case is more complicated and depends on the type of element and the form of equilibrium condition being used. The changes depend on whether the element contributes to the tidal prism and whether the equilibrium is defined in terms of tidal prism or tidal range. In this version of Asmita, the area is constant and any the adjustment to the equilibrium area only needs to be made once.

Equilibrium volumes, however, are reset each time step based on the chosen relationship (usually based on prism or surface area). The volumes are adjusted as follows:

	Change ¹	Equilibrium	Type ²	Action ³
1	V+ S	Area	Prism	Change alters area and hence equilibrium. To maintain the change in volume, need to account for the change in prism. $dV_{fix} = dV - V_e(dS)$
2	V	Area	Prism	V_e depends on surface area and is not altered directly by change. Equilibrium volume needs to be corrected for subsequent timesteps, $dV_{fix}=dV$.
3	S	Area	Prism	Modify area of element at time of intervention, $dV_{fix}=0$.
4	V+S	Prism	Prism	Change alters tidal prism and hence equilibrium, $dV_{fix}=0$. To maintain the change in volume, need to account for the change in prism. $dV_{fix} = dV - V_e(dV)$.
5	V	Prism	Prism	
6	S	Prism	Prism	Modify area of element at time of intervention, $dV_{fix}=0$.
7	V+S	Prism	Non-prism	<i>Channel</i> Change alters prism. The change depends on the relative extent of changes in v and S and how the change in S influences the tidal prism. $dV_{fix} = dV - V_e(dS)$. As $dS=>0$, $dV_{fix}=>dV$. As dS increases so the total change in equilibrium increases reflecting the change in prism <i>Delta</i> Change does not alter prism. Equilibrium volume needs to be corrected for subsequent timesteps, $dV_{fix}=dV$.
8	V	Prism	Non-prism	<i>Channel</i> Change does not alter prism. Equilibrium volume needs to be corrected for subsequent timesteps, $dV_{fix}=dV$. <i>Delta</i> Change does not alter prism. Equilibrium volume needs to be corrected for subsequent timesteps, $dV_{fix}=dV$.
9	S	Prism	Non-Prism	Modify area of element at time of intervention, $dV_{fix}=0$.

- (1) Volume and/or Surface area. If not stated, change is zero.
- (2) Element types that contribute to the prism include tidal flat, saltmarsh, storage, and delta flat. Non-prism elements are channel, flood delta and ebb delta. Element types outside the estuary/inlet (Beach face, Shore face, and Spit) are treated as transient.
- (3) V is volume, S is surface area, P is tidal prism. dV_{fix} is a correction applied to V_e at each time step. dS_{fix} is similar: $V_e(t) = V_e(t) + dV_{fix}$; $S_e = S_e + dS$ ($dS_{fix}=0$ for all cases for the fixed area model). The offset used to correct dV depends on any scaling being applied, (i) to initial conditions and (ii) to account for any advection offset. $V_e(dS) = K \cdot \alpha dS^\beta$, where K is any scaling, α and β are equilibrium coefficients. Similarly, $V_e(dV) = K \cdot \alpha dV^\beta$, where K is any scaling, α and β are equilibrium coefficients

In addition, if an intervention is specified with a surface area of +999 m², this is treated as a flag to make the element inactive, rather than as a change in area. The surface area is set to +999 m².

Active/Inactive Elements

By initialising an element with a surface area of 999 m² the element is considered inactive, regardless of the volume specified. Similarly, if an intervention removes an area from an element that makes the area less than 999 m², this is set to 999 m² and the element becomes inactive. This can be reversed simply by reinstating the area with another intervention.

Horizontal and Vertical Exchanges

This option is invoked by setting one or more links in the Dispersion matrix to NaN. The dynamic exchanges are then defined as changes in either value at a defined time in the model run period. In the current version of Asmita, these changes are applied to all elements with NaN links. Altering the horizontal exchange of the NaN link and the vertical exchange in the element that is the up-tide (landward) element of the link. Varying the horizontal exchange may be appropriate when introducing a reclamation over part of a tidal flat. The exchange with the flat depends on the length of the element and the tidal range. The reclamation may reduce the length of the flat frontage open for tidal exchange with the channel and hence the exchange rate. Altering the vertical exchange rate may be of use if there is a need to adjust the response time of the up-tide element.

Marsh Edge Erosion

The defined rate is used to compute the change in volume and area for each time step. If an element is defined as non-Erodible or inactive (see above) then there is no edge erosion. If a marsh is reduced to an area less than 999 m², the marsh is deemed to be inactive and the area is set to 999 m².

5.5.3 Managed realignment, reclamation and storage schemes

With managed realignments and water storage schemes, the requirement is usually to introduce new elements into the system at some point in time. However, changing the element configuration dynamically is not provided for in the current version. There is therefore a need to include such elements in the initial model definition, and adjust the horizontal exchange in the dispersion matrix (see Section 4.4) and/or the vertical exchange for the relevant element. Such changes can, therefore, be represented using the following workflow:

1. Define the initial marsh element volume this can be the volume upon introduction of the realignment site or some small value (e.g., 100 m³), and the surface area is set to a value of 999.
2. Define an intervention for the marsh element that increases the volume and plan area to the full extent of the new marsh, at the desired time within the model run time.
3. Check that interventions and saltmarsh biomass are included in the 'Conditions' settings.
4. When adding an intervention that is initially too deep for saltmarsh species to be present, the element is treated as a bare flat, using the prism relationship defined by the Equilibrium Coefficients. These values may need to be adjusted for the saltmarsh to get meaningful depths (e.g., reduce alpha to O[0.001] or less and set beta=1 and eqtype=0).

If the initial element volume and area are too small this can generate a time-step warning. Try running with the specified value. If this results in NaN values for the mass and sediment balance, output that is oscillating, or output with large perturbations, try either reducing the time-step, or increasing the size of the initial marsh element.

Note: Fixed depths when the element is in the "closed" state have no meaning and can be very large and negative when, for example, an existing marsh is closed some time into a run.

Reclaiming an existing marsh and subsequently restoring it uses a similar process.

1. Set up the marsh as an element in the model.

2. Define an intervention for the marsh element and set the plan area to 999 at the time of the closure, and another intervention restoring the surface area. Note: the volume can also be removed but has not effect on the results
3. Check that interventions and saltmarsh biomass are included in the ‘Conditions’ settings.

5.6 Saltmarsh

Saltmarsh elements can be specified by defining the water volume over the marsh and the surface area. There is not a lot of available data with which to define the parameters of saltmarsh species (see HR Wallingford report IT.573 for a review of available data). The following table gives some indicative parameter values based on species reported in the literature. Some information is provided by Randerson (1979) as modelled peak biomass for a range of species and the zonation information is gleaned from the work of Gray (1992). Lower and upper species bounds are given as factors, κ , of the MHWN elevation (mODN) so that the depth is given $D = a - \kappa \cdot z_{MHWN}$, where a is the tidal amplitude. Some data on species ranges in Venice Lagoon are presented by Silvestri *et al.* (2005) as given in the columns headed (VL) below.

Species	Min depth factor, κ	Max depth factor, κ	Min depth (VL)	Max depth (VL)	Max biomass kg.m ⁻²
Spartina	1.7	1.2	0.13	0.28	0.8
Puccinella	1.9	1.5	0.08	0.18	1.5
Halomione	1.9	1.55	0.03	0.13	1.7
Salicornia	1.8	1.4	0.18	0.38	0.2
Limonium	-	-	0.13	0.23	-
Sarcocornia	-	-	0.03	0.28	-
Sueada	1.8	1.4	-	-	0.2
Aster	1.8	1.2	-	-	0.6

For North Inlet delta in US, Morris *et al.* (2002) suggest values for sediment loading, $q_m = 0.00018 \text{ yr}^{-1}$ and biogenic production rate, k , in the range $1.5\text{-}2.4 \times 10^{-2} \text{ m}^2 \text{ kg}^{-1} \text{ yr}^{-1}$ for short and tall forms of *S. alterniflora* respectively. Mudd *et al.* (2004) suggest a value of $k_b = 1.68 \times 10^{-4} \text{ yr}^{-1}$ and Marani *et al.* (2007) use $k_m = 2.4 \times 10^{-3} \text{ m}^3 \text{ kg}^{-1} \text{ s}^{-1}$. Each of these authors uses a slightly different representation of k , hence the different dimensions. ASMITA uses values with the same dimensions as Morris’ k and the range he cites can be used as initial estimates. Using the parameter values given in Mudd *et al.* (2004), it is possible to derive estimates of α and β as $\alpha = 3.4 \times 10^{-5} \text{ s}^{-1}$ and $\beta = 0.5$ using the tabulated values, and $\alpha = 4.2 \times 10^{-5} \text{ s}^{-1}$ and $\beta = 0.56$ using the equation in Figure 3 of the paper.

For saltmarsh to be included as a bio-geomorphological element with biomass production the ‘Include saltmarsh biomass’ box needs to be checked in *Setup>Run Parameters>Conditions*. If this is NOT checked the saltmarsh is treated the same as other elements and the equilibrium volume is based on the tidal prism relationship using the parameters specified for saltmarsh for the selected equilibrium coefficient set. This is specified by selecting an equilibrium coefficient set in *Setup>Run Parameters>Equilibrium Coefficients* (the coefficients are defined in userprismcoeffs.m).

The utility in *Setup>Saltmarsh>Equilibrium marsh depth*: (see Section 4.4) allows the influence of the defined saltmarsh parameters to be explored by varying species productivity. This can be a useful diagnostic in helping to understand the role of the saltmarsh with the defined model setup.

5.7 Storage

Storage areas are independent of the main channel network and can be linked to channel or flat elements. The volume of water exchanged contributes to the downstream prism of the estuary but the estuary does not influence the storage element, other than controlling the rate of sediment supply. The basis for defining equilibrium of this type of sub-system is poorly defined and needs further development. The current model defines the equilibrium volume based on the tidal prism in the same way as other elements. This is specified by selecting an equilibrium coefficient set in *Setup>Run Parameters>Equilibrium Coefficients* (the coefficients are defined in *userprismcoeffs.m*).

5.8 Hydraulics

The core model uses the horizontal exchanges defined by the Dispersion matrix and any imposed advection fluxes to determine the exchanges between elements. For very large systems, with river discharges that vary over a wide range, the mean water level along the channel can also vary. This results in an additional change in water volumes, due to the river-tide interaction, which is not accounted for in the basic model set-up. To address this, the Hydraulics option allows the CSTmodel App to be pre-run to create a look-up table of surface water elevations and tidal amplitudes for a range of river discharges. The CSTmodel computes the mean tide level, high and low water levels, tidal velocity amplitude and river velocity, along the length of a convergent estuary using the analytical model of Cai, Savenije and Toffolon - hence the CSTmodel (Cai, 2014). The CSTmodel output is interpolated for a given river discharge input to Asmita and as a function of distance from the mouth to determine representative values for each reach of the model (taken at the mid-point of each reach). The reach values are then assigned to all elements in a given reach to define the water level change at each time step. This will be determined by high water for some elements (eg tidal flats and saltmarsh) and low water for others (eg channels and deltas). This then contributes to the total volume changes at each time step.

5.9 Tidal Pumping

For large macro-tidal estuaries, the equilibrium state can have a variable along-estuary concentration, from a relatively low value at the mouth to a maximum some distance into the estuary. This reflects (i) flood dominance due to internally generated asymmetry in the tidal flow transporting sediment landward and (ii) ebb dominance due to the river flow moving sediment downstream. The convergence of the two gives rise to a turbidity maximum at some points in the estuary depending on the relative strength of the two influences. The landward mechanism has been characterised as tidal pumping (Friedrichs *et al.*, 1998; Scully and Friedrichs, 2007; Yu *et al.*, 2014) and can be thought of as an upstream directed discharge and associated sediment flux. This allows the concept to be incorporated into the ASMITA model.

The parameters needed for the inclusion of tidal pumping are largely obtained from parameters used to set up ASMITA, or derived when the model is run, with the addition of the M4 amplitude and phase, the sediment mobility coefficient and the width and CSA channel convergence lengths. If the CSTmodel is being used to modify water levels along the channel, the adjusted water levels are used in the tidal pumping algorithm. However, this only has a significant effect for large systems with high river discharge. In many systems the inclusion of the CSTmodel is not necessary but it can help represent the along-channel variation found in estuaries such as the Yangtse and Amazon.

5.10 Morphological Response

The *Response* tab provides a summary of morphological response times based on defined parameter settings. The tabulated values list:

- (i) limiting rate of sea level rise for inorganic case
(max_slr = **dd**/cb/sm, see Townend *et.al.* (2016) for definitions)
- (ii) limiting rate of sea level rise including influence of saltmarsh organic contribution
(max_slr = (**dd**+ Blim.vm.cb)/cb/sm, see Townend *et.al.* (2016) for definitions)
- (iii) the eigenvalues ‘system timescales’ for each element based on the inverse eigenvalues of the matrix function **BF** (see Kragtwijk *et al.*, 2004).

An estimate of the morphological timescale assuming the system is a single element is based on Eq. 11 of Kragtwijk *et.al.* (2004), with estimates given based on the coarse and fine sediment equilibrium concentration values.

5.11 Derive Output

The *Run> Derive Output* option allows the user to make use of the data held within App to derive other outputs or, pass selected data to an external function (see Section 4.5). The equation box can accept t, x, y, z in upper or lower case. Time can be assigned to X, Y, or Z buttons, or simply included in the equation as t (if the data being used in one of the variables includes a time dimension). Each data set is sampled for the defined data range. If the data set being sampled includes NaNs, the default is for these to be included (button to right of Variable is set to ‘+N’). To exclude NaNs press the button so that it displays ‘-N’. The selection is based on the variable limits defined whenever a variable is assigned to X, Y or Z using the X, Y, Z buttons.

The equation string entered in the UI is used to construct an anonymous function as follows:

```
heq = str2func(['@(t,x,y,z,utext,mobj) ',usereqn]); %handle to anonymous function
var = heq(t,x,y,z,utext,mobj);
```

This function is then evaluated with the defined variables for t, x, y, and z and optionally *utext* and *mobj*. *utext* allows text string to be passed (any string enclosed in single quotes, e.g. ‘Test’) and *mobj* passes the handle to the main UI.

Some functions may alter the length of the reference co-ordinates (x, y, z, t), or return more than one variable. These can be handled by passing a comment appended to the function definition e.g.

subsample(x,t, thr,mobj) %time. In the current version of the toolbox, only ‘time’ is handled as a key word and this allows functions to derive a new timeseries with different time input values to those of the input variable(s).

If the function returns a single valued answer, this is displayed in a message box, otherwise it is saved, either by adding to an existing dataset, or creating a new one (see Section 4.5). The variables selected are sub-sampled based on any adjustments made when each variable is assigned to a button. The dimensions of the vector or array with these adjustments applied need to be dimensionally correct for the function being called.

Equations can use functions such as diff(x) - difference between adjacent values - but the result is n-1 in length and may need to be padded, if it is to be added to an existing derived data set. This can be done by adding a NaN at the beginning or the end: e.g.: [NaN; diff(x)]. NB: the separator needs to be a semi-colon to ensure the correct vector concatenation. Putting the NaN before the equation means that the difference over the first interval is assigned to a record at the end of the interval. If the NaN is put after the function, then the assignment would be to the records at the start of each interval.

NB1: functions are forced to lower case (to be consistent with all Matlab functions), so any external user defined function call must be named in lower case.

Another useful built-in function allows arrays to be sub-sampled. This requires the array, *z* to be multiplied by an array of the same size. By including the dimensions in a unitary matrix, the range of each variable can be defined. For a 2D array that varies in time one way of doing this is:

```
>> z.*repmat(1, length(t), length(x), length(y))
```

*NB2: the order of the dimensions *t*, *x*, *y* must match the dimensions of the array, *z*.*

5.11.1 Calling an external function

This interface can also be used as an interface to user functions that are available on the Matlab search path. Simply type the function call with the appropriate variable assignment and the new variable is created. (NB: the UI adopts the Matlab convention that all functions are lower case). Some examples of functions provided in Asmita are detailed in the next Section.

The input variables for the function must match the syntax used for the call from the Derive Output UI. In addition, functions must return output as a cell array. This can be a single value, a new variable vector or array or a time series with one or more variables. If *time* is included then this should be in the first cell with variables in the subsequent cells.

If the %time or %rows instruction is included in the call, row data are added providing that the length of the input dataset matches the output dataset. If there is no output to be passed back, the function should return a cell array containing the string 'no output' to suppress the message box which is used for single value outputs.

An alternative when calling external functions is to pass the selected variables as dstables, thereby also passing all the associated metadata and RowNames for each dataset selected. For this option up to 3 variables can be selected but they are defined in the call using *dst*, for example:

```
mufunction3(dst, 'utext', mobj) where 'utext' and mobj are options as defined above.
```

This passes the selected variables as a struct array of dstables to the function. Using this syntax the function can return a dstable or struct of dstables, or a cell array containing one or more data sets.

Adding functions to the Function library

To simplify accessing and using a range of functions that are commonly used in an application, the function syntax can be predefined in the file `functionlibrarylist.m` which can be found in the `utils` folder of the `multoolbox`. This defines a struct for library entries that contain:

- *fname* - cell array of function call syntax
- *fvars* - cell array describing the input variables for each function
- *fdesc* - cell array with a short description of each function

New functions can be added by simply editing the struct in `functionlibrarylist.m`, noting that the cell array of each field in the struct must contain an entry for the function being added. In addition a sub-selection of the list can be associated with a given App based on the class name of the main UI. To amend the selection included with an App or to add a selection for a new App edit the '`switch classname`' statement towards the end of the function.

The Function button on the Derive Output UI is used to access the list, select a function and add the syntax to the function input box, where it can be edited to suit the variable assignment to the XYZ buttons.

5.11.2 Pre-defined functions

The following examples are provided within Asmita, where the entry in the UI text box is given in Courier font and X, Y, Z, refer to the button assignments.

Some useful examples primarily for timeseries data include:

1. **Moving Average.** There are several moving average functions available from the Matlab Exchange Forum, such as `moving.m`. The call to this function is:

`moving(X, n, 'func')`, where `x` is the variable to be used, `n` specifies the number of points to average over and '`func`' is the statistical function to use (e.g. mean, std, etc). If omitted the *mean* is used. Add `%time` to the call, to include time in the output dataset.

2. Moving average (or similar) of timeseries, over defined duration, advancing at defined interval

`movingtime(x, t, tdur, tstep, 'func')`, where `x` is the variable to be used and `t` the associated datetimes (defined by variable selection), `tdur` is the duration over which to apply the statistic, `tstep` is the interval to advance the start time for the averaging period and '`func`' is the statistical function to use (e.g. mean, std, etc). If omitted the *mean* is used. `tdur` and `tstep` are both duration character strings of form '2.5 d'. Any of the following duration intervals can be used: y, d, h, m, or s. Returns a time series based on the defined `tstep`, where the time used is for the beginning of each stepping interval, i.e., every `tstep` from the start of the record to the nearest interval that is less than `tdur` from the end of the record.

3. **Down-sampling a time series.** This allows a timeseries to be resampled at a different interval (that must be less than the source timeseries). The call to this function is:

`downsample(x, t, 'period', 'method')`, where `x` is the variable to be resampled, `t` is the associated time for that variable, `period` can be 'year', 'month', 'day', 'hour', 'minute', 'second', and `method` can be any valid function call such as 'mean', 'std', etc. The 'period' is required but the 'method' is optional and if omitted the mean is used. For timeseries with gaps the 'nanmean' function is particularly useful but requires the Statistics toolbox.

4. **Interpolate and add noise.** To infill a record with additional points and, if required, add some random noise to the interpolated values. This is called using:

`interpwithnoise(x, t, npad, scale, method, ispos)`, where `X` is the variable, `t` is time, `npad` is the number of points to add between the existing data points, `scale` determines the magnitude of the random noise (a value of 0 results in an interpolated record with no noise), `method` is the Matlab algorithm used for the interpolation (the default is linear) and `ispos` is a true/false flag which sets negative values to zero if true.

5. **Subsample one record at the time intervals of another record** (e.g. subsample water levels to be at the same intervals as the wave data). Function is:

`subsample_ts(X, t, mobj)`, where `X` and `t` are the variable to be subsampled and `mobj` is the UI handle (must be `mobj`). The user is prompted to select the dataset to be used to define the time intervals. A time series is returned and added as a Derived data set. The user is prompted to define the metadata for the new data set.

6. **Subsample one record based on a threshold defined for another record** (e.g. subsample waves based on a threshold water level). Function is:

`subsample(X, t, thr, mobj)`, where `X` and `t` are the variable to be subsampled, `thr` is the threshold value and `mobj` is the UI handle (must be `mobj`). The user is prompted to select the dataset and variable to be used to define the condition and a condition operator (`<=`, `==`, etc). A time series is returned and added as a Derived data set. The user is prompted to define the metadata for the new data set.

7. **Recursive plot.** Generates a plot of a variable plotted against itself with an offset (e.g. $x(i)$ versus $x(i+1)$). This is called from the Derive Output GUI using:

`recursive_plot(x, 'varname', nint)`, where x is the variable, '*varname*' is a text string in single quotes and *nint* is an integer value that defines the size of the offset.

8. **Phase plot.** This function is similar to the recursive plot function but generates a plot based on two variables that can, optionally, be functions of time. The call to this function is:

`phaseplot(X, Y, t)`, where X and Y are the variables assigned to the respective buttons and t is time (this does not need to be assigned to a button and t can be omitted if a time stamp for the datapoints is not required).

6 User functions

To allow further research of system behaviour to be undertaken without disrupting the underlying model structure, selected functions have been made external to the class definitions

6.1 User Prism Coefficients function – `userprismcoeffs.m`

The default method for calculating the equilibrium volumes is as a function of tidal prism in the form:

$V_e = \alpha \cdot P^\beta$, P is the tidal prism upstream of the element (including the prism within the reach that the element belongs to) and α and β are coefficients that have been found to provide a means of representing the relationship between cross-sectional area and prism (the so called O'Brien or Jarrett relationship), which can also be used to characterise other gross properties of estuary systems (Townend, 2005). This is the default used when the variable `eqtype` = 1. The option to use tidal range instead of tidal prism is also available and is invoked by setting `eqtype` = 0. The values of α and β need to be modified to reflect the use of tidal prism or tidal range. The default values of α and β for all element types is 1.

The choice of coefficients is selected from the list of available coefficient sets using *Setup>Run Parameters>Equilibrium Coefficients*. The coefficient sets are defined in the `userprismcoeffs.m` external function for each element type. Using default values of 1 and setting the 'Scale to initial values flag' (*Setup>Run Parameters>Conditions*), has the effect of setting $\alpha = V_{m0}/P$, where V_{m0} is the element volume at time $t=0$.

Setting element type specific values (as is the examples in the file provided) results in V_e being calculated using these values. If the 'Scale to initial values flag' is set, V_e is scaled based on the initial ratio of V_{m0}/V_e . This allows two scenarios to be considered:

- (i) With element specific parameters defined and no scaling based on initial conditions, any difference with the defined element volumes at $t=0$ is a perturbation and the system will seek to adjust to the defined equilibrium conditions
- (ii) With element specific parameters defined and scaling based on initial conditions, there is no initial perturbation but any non-linear variation over time is represented. This captures the assumption that the values used at the start of the run are close to an equilibrium condition but may not agree with exactly with the equation being used and so are adjusted to remove this initial difference by re-scaling them.

This function can be varied in conjunction with changes in `ASM_model` (see Section 6.4) to introduce different relationships to define the equilibrium conditions. The default, a generic set and two location specific cases, for the Venice and Ameland, are included in the version of `userprismcoeffs.m` provided. Note that in the Ameland model a constant value is assigned to tidal flat equilibrium by setting alpha to this value and beta to zero.

Additional settings can be added as cases of the switch in the `userprismcoeffs.m` function file. For these to be shown in the pop-up selection list, when using *Setup>Run Parameters>Equilibrium Coefficients*, the property setting in the `RunProps` class needs to be edited:

```
%list of options used in UserEqPrismCoeffs
UserEqCoeffOptions = {'Default', 'Generic', 'Venice', 'Ameland', 'Pagham'};
```

6.2 UserSLRate

The rate of sea level rise is defined in *Setup>Estuary> Water Levels* and can take several forms. The amount of sea level rise at time, t , is computed in `sealevelrise.m`. The settings possible use different functions in `sealevelrise` and include:

- (i) a linear rate of sea level rise - slr input as a single positive value in m/yr.
e.g., input *Rate of sea level rise (m/yr)*: 0.002

- (ii) an exponential rate of sea level rise. The exponential rate of increase (m/yr) is input as a negative value. The exponential increase starts in 1900 and the rate prior to 1900 is taken as 0.001 m/yr.
e.g., input *Rate of sea level rise (m/yr)*: -0.011
- (iii) an exponential rate of sea level rise, specified as a vector with values for the exponential rate increase (m/yr), start year of exponential increase and linear rate to be used prior to start year.
e.g., input *Rate of sea level rise (m/yr)*: 0.011 1900 0.001

$$input = [r, N, m_N]$$

$$slr(t) = \begin{cases} m_N \cdot t & t \leq N \\ slr(N) + \exp(r(t - N)) & t > N \end{cases}$$
This gives the same as option (ii) for the example shown above.
A pivot year can also be included (see below).
e.g., input *Rate of sea level rise (m/yr)*: 0.011 1900 0.001 1980
- (iv) extends option (iii) to include the Holocene. Also specified as a vector with 6 values. The first 3 are as defined for option (iii). The remaining values are the scale, shape and offset for an exponential that approximately captures the Holocene back to about 8000 BP or 6000 BC. For example, values of 1.32, 2400, 0.6 provide a good approximation against data for the Humber estuary.
e.g., input *Rate of sea level rise (m/yr)*: 0.011 1900 0.001 1.32 2400 0.6
A pivot year can also be included (see below).
e.g., input *Rate of sea level rise (m/yr)*: 0.011 1900 0.001 1.32 2400 0.6 1980
- (v) a user defined function by specifying the rate of sea level rise as NaN. The desired function should then be called from or added to case 4 in sealevelrise.m.
e.g., input *Rate of sea level rise (m/yr)*: NaN

Options (iii) and (iv) can also include a pivot year. This defines the year in which mean sea level is zero to the datum used in the model. The year is appended to the vector that defines the Rate of sea level rise in the Water Levels input UI. For example [0.011 1900 0.001 1980] would produce the same sea level curve as option (iii) with the curve offset so that mean sea level is zero in 1980. **If the pivot year is not specified a default value of 2000 is used.**

For example, by defining a rate ≤ -0.01 m/year, option (ii) is invoked and returns a linear rise of 1mm/year prior to 1900 and an exponential rate of increase thereafter that approximates IPCC projections used in the UK (circa 2010). However, this function can be replaced by any function that returns the amount of sea level rise at model run time from $t=0$ by using option (v).

6.3 Command Line option

Whilst the model is designed to be run from the UI, it can also be run from the command line, using a script or in batch. In the Asmita App folder there is an 'asmita_functions' sub-folder with the following functions and scripts:

Function or Script	Description
asmita_command_line_script.m	Script to run a model, which is specified in the script, extract some results and plots them. The script includes some sample code to modify a selection of model properties before running the model.

asmita_command_line_loop_script.m	Script to select multiple models, run them in a loop, extract some results and plot them. Uses the properties defined in the source mat file of each model.
asmita_command_line_run.m	Function to run Asmita App using the supplied model path and model mat file name.

These can be used as templates for other applications, as required.

6.4 Asmita_model and AsmitaModel

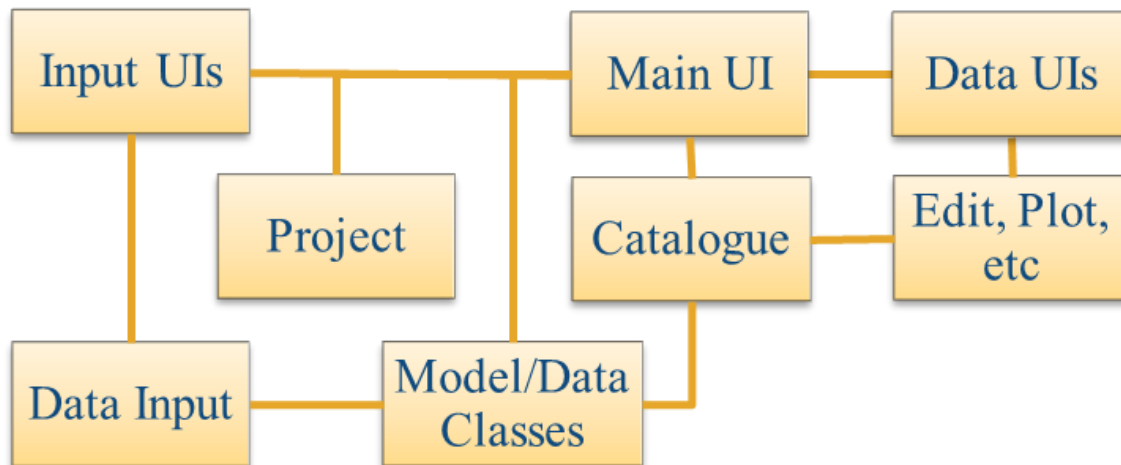
The core Asmita model is written as an interface which is then implemented by ASM_model where the user can overload any of the functions in the underlying ASMinterface definitions. This is illustrated by the inclusion of the function asmitaEqFunctions in ASM_model.m. The model is run using the AsmitaModel class which handles initialisation, time stepping and post processing results to create the model output tables.

7 Program Structure

7.1 Classes and functions

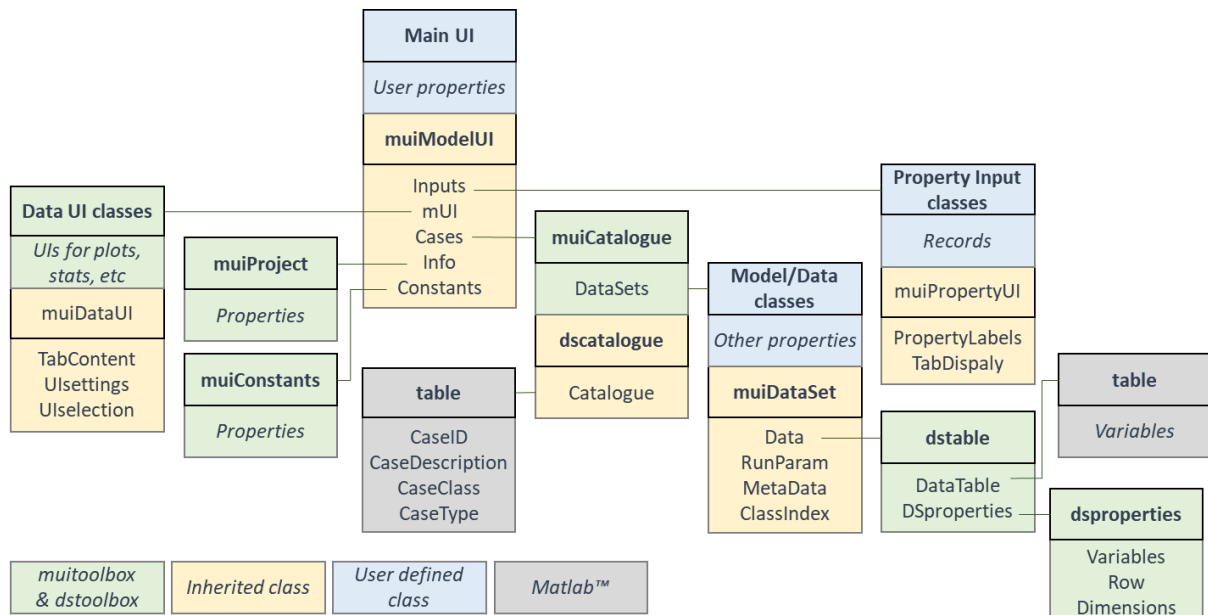
The overall structure of the code is illustrated schematically in Figure 17. This is implemented through several classes that handle the graphical user interface and program workflows (Main UI) and several classes that handle the data manipulation and plotting (Input UIs and Data UIs).

Figure 17 – High level schematic of program structure



The interfaces and default functionality are implemented in the Asmita App using the following mui toolbox classes depicted in Figure 18, which shows a more detailed schematic of the program structure. See the mui toolbox and dstoolbox documentation for more details.

Figure 18 – schematic of program structure showing how the main classes from mui toolbox and dstoolbox are used



In addition, the Asmita App uses the following classes and functions:

Advecton – handles all types of advective flow, including rivers, drift and tidal pumping

ASM_PlotsUI – implements the muiDataUI interface for a range of plot types

ASM_Plots – inherits muiPlots and customises plots for the ASMITA application

- Asmita** – implements GUIinterface for the ASMITA application
- ASMinterface** – basic functionality of the ASMITA model
- Asmita_model** – implements ASMinterface to allow functions in ASMinterface to be overloaded
- AsmitaModel** – class to run the model. Handles initialisation, time stepping and post processing results to create the model output tables.
- CSThydraulics** – handles the initialisation and calling of the CSTmodel App
- Drift** – setup and provide access to littoral drift sediment inputs
- Element** – setup and provide access to element parameters
- EqCoeffParams** – select and access the equilibrium coefficients to be used in the model (uses definitions in userprismcoeffs.m)
- Estuary** – parameters and methods for system, such as dispersion and linkages
- Interventions** – setup and provide access to imposed changes to system geometry (volume and area)
- Reach** – setup and provide access to parameters for reaches (a channel element plus linked elements)
- River** – setup and provide access to river flow/sediment inputs
- RunConditions** – define the specific conditions for a model run
- RunPropertiess** – define the model run time parameters
- Saltmarsh** – setup and access to saltmarsh parameters and functions to define behaviour
- WaterLevels** – setup and access to water level definitions, tidal parameters and changes over time

The following functions are also used:

Saltmarsh functions - Functions that implement generic aspects of the saltmarsh model

bioenhancedsettling.m - Adjust the vertical exchange due to biological enhanced settling for the specified depth.

concovermarsh.m - Calculate time averaged concentration over the marsh for a range of depths and return a dstable of depth, concentration and submergence time for an emulated spring-neap cycle.

interpdepthload.m - Iterate to find the equilibrium depth and sediment load, $qm(s^{-1})$

marsh_conc.m - Calculate the concentration over a marsh flat using method proposed by Krone(1987).

morris_biocoeffs.m - Calculate the biomass coefficients for $Bc=aD+bD^2+c$ where D is the depth at peak biomass.

morris_eqdepth.m - Solve the Morris equation for equilibrium depth.

Network functions - Functions used to manipulate networks and graphs of flow pathways

ele2node.m - find graph node ids for specified element ids.

graph2matrix.m - extract the exchange matrix and vectors of external exchanges from a directed graph with labels defined in nodetxt.

inverse_graph.m - To reverse the direction of a directed graph by taking the inverse of the adjacency matrix.

matrix2graph.m - use an exchange matrix and a vector of external exchanges to construct a directed graph with labels defined in *nodetxt*.

rescale_graph.m - rescale the exchanges in a network based on the inputs defined in the *exchIn* vector'.

setgraph.m - update the Network or Flow tabs with directed graphs of network

setmatrix.m - generate UI to edit dispersion or advection matrix

setnodetxt.m - generate UI to edit dispersion or advection matrix

type_sub_graph.m - sub-sample a graph network based on node type

Other functions - Other functions used in Asmita

readinputdata.m -read advection timeseries data from file. Read format is file specific but can handle multiple columns with space or tab separators.

sealevelrise.m - function to compute sea level rise using linear or exponential rate of change.

simple_tide.m - function to compute a tidal water level time series using the main constituents scaled to the required tidal amplitude.

userprismcoeffs.m - parameters to define volume-prism relationships of the form:

volume = $\alpha \cdot \text{prism}^\beta$ or volume = $\alpha \cdot \text{tidal range}^\beta$.

asmita_command_line_run.m - function to run Asmita App using the supplied model path and model mat file name.

Scripts

asmita_command_line_script.m - script to run a model, which is specified in the script, extract some results and plots them. The script includes some sample code to modify a selection of model properties before running the model.

asmita_command_line_loop_script.m - script to select multiple models, run them in a loop, extract some results and plot them. Uses the properties defined in the source mat file of each model'

The order in which functions are called during initialisation and runtime is important because of the interdependence of parameters across the various classes. The call sequence is detailed in Appendix A – Function call sequence in AsmitaModel and the UML class diagrams are included in Appendix B – UML Class diagrams.

7.2 Unit Testing

Unit testing allows selected models to be run and compared with results derived using a previous version of the model. If the model passes the unit test, then the output has not changed significantly because of any changes that have been made. The code for Unit Testing is in the 'asmita_unit_tests' folder. This contains the following classes, functions, scripts and subfolders:

Classes and functions	Description
UserUnitTesting.m	Function to select and call selected unit test of Asmita
AsmitaTest.m	Class to implement the unit testing
runasmitamodel.m	Function to initialise Asmita, load selected test model and get results
UserSaveTestData.m	Function to save a test data set for use in AsmitaTest unit testing

Call `UserUnitTesting` from the command line. The user is prompted to select a model to test, which is then run and compared with the test data file. The results are reported in the Command Window. The following test models are included:

`H1EMTest` – single element model based on the Humber estuary.

`HumberTest` – 3 element model of the Humber estuary.

`Amelandertest` – 3 element model to compare with results from the original Fortran model.

`VeniceTest` – 9 element model of the lagoon with 3 inlets from the sea.

`SevernTest` – 16 element model, 6 channel elements and multiple river inflows, to test tidal pumping.

`InletTest` – 7 element model, with time series inputs of littoral drift.

`SouthamptonTest` – 4 element model of Southampton Water including interventions.

`YangtzeTest` – 30 element model which includes along channel hydrodynamics using the `CSTmodel`.

8 Bibliography

- Cai, H., 2014. A new analytical framework for tidal propagation in estuaries. PhD, TU Delft, TU Delft, 187 pp.
- Friedrichs, C.T., Armbrust, B.D., de Swart, H.E., 1998. Hydrodynamics and equilibrium sediment dynamics of shallow, funnel-shaped tidal estuaries. In: J. Dronkers, M.B.A.M. Scheffers (Eds.), *Physics of Estuaries and Coastal Seas*. Balkema, Rotterdam, pp. 315-327.
- Garnier, R., Townend, I., Monge-Ganuzas, M., de Santiago, I., Liria, P., Abalia, A., Epelde, I., Campo, A.d., Chust, G., Valle, M., González, M., Mader, J., Gómez, M., Castillo, C., Uriarte, A., 2022. Modelling the morphological response of the Oka estuary (SE Bay of Biscay) to climate change. *Estuarine, Coastal and Shelf Science*, 279, 108133. [<https://doi.org/10.1016/j.ecss.2022.108133>]
- Gray, A.J., 1992. Saltmarsh plant ecology: zonation and succession revisited. In: J.R.L. Allen, K. Pye (Eds.), *Saltmarshes: Morphodynamics, conservation and engineering significance*. Cambridge University Press, Cambridge, pp. 63-79.
- Knaapen, M.A.F., Spearman, J., Townend, I.H., Rossington, S.K., Fletcher, C.A., 2008. ASMITA - Marshes: Development of an ASMITA Saltmarsh Routine, Wallingford.
- Kragtwijk, N.G., Stive, M.J.F., Wang, Z.B., Zitman, T.J., 2004. Morphological response of tidal basins to human interventions. *Coastal Engineering*, 51, 207-221. [<https://doi.org/10.1016/j.coastaleng.2003.12.008>]
- Lodder, Q., Huismans, Y., Elias, E., de Looff, H., Wang, Z.B., 2022. Future sediment exchange between the Dutch Wadden Sea and North Sea Coast - Insights based on ASMITA modelling. *Ocean & Coastal Management*, 219. [10.1016/j.ocecoaman.2022.106067]
- Powell, M.A., Thieke, R.J., Mehta, A.J., 2006. Morphodynamic relationships for ebb and flood delta volumes at Florida's tidal entrances. *Ocean Dynamics*, 56(3-4), 295-307.
- Randerson, P.F., 1979. A simulation of saltmarsh development and plant ecology. In: B. Knights, A.J. Phillips (Eds.), *Estuarine and coastal land reclamation and water storage*. Saxon House, Farnborough, pp. 48-67.
- Scully, M.E., Friedrichs, C.T., 2007. Sediment pumping by tidal asymmetry in a partially mixed estuary. *Journal of Geophysical Research*, 112(C07028).
- Silvestri, S., Defina, A., Marani, M., 2005. Tidal regime, salinity and salt marsh plant zonation. *Estuarine, Coastal and Shelf Science*, 62, 119-130.
- Taylor, K.E., 2001. Summarizing multiple aspects of model performance in a single diagram. *Journal of Geophysical Research - Atmospheres*, 106(D7), 7183-7192. [10.1029/2000JD900719]
- Townend, I.H., 2005. An examination of empirical stability relationships for UK estuaries. *Journal of Coastal Research*, 21(5), 1042-1053.
- Townend, I.H., Rossington, S.K., Knaapen, M.A.F., Richardson, S., 2010. The dynamics of intertidal mudflat and saltmarshes within estuaries. In: J. McKee Smith (Ed.), *Proceedings of the International Conference on Coastal Engineering*. Coastal Engineering Research Council, Shanghai, pp. 1-10.

- Townend, I.H., Wang, Z.B., Rees, J.G., 2007. Millennial to annual volume changes in the Humber Estuary. *Proc.R.Soc.A*, 463, 837-854. [<https://doi.org/10.1098/rspa.2006.1798>]
- Townend, I.H., Wang, Z.B., Stive, M.J.E., Zhou, Z., 2016a. Development and extension of an aggregated scale model: Part 1 – Background to ASMITA. *China Ocean Engineering*, 30(4), 482-504. [<https://doi.org/10.1007/s13344-016-0030-x>]
- Townend, I.H., Wang, Z.B., Stive, M.J.E., Zhou, Z., 2016b. Development and extension of an aggregated scale model: Part 2 – Extensions to ASMITA. *China Ocean Engineering*, 30(5), 651-670. [<https://doi.org/10.1007/s13344-016-0042-6>]
- van Goor, M.A., Zitman, T.J., Wang, Z.B., Stive, M.J.F., 2003. Impact of sea-level rise on the morphological equilibrium state of tidal inlets. *Marine Geology*, 202, 211-227. [[https://doi.org/10.1016/s0025-3227\(03\)00262-7](https://doi.org/10.1016/s0025-3227(03)00262-7)]
- Walton, T.D., Adams, W.D., 1976. Capacity of inlet outer bars to store sand. Fifteenth Coastal Engineering Conference. American Society of Civil Engineers, pp. 1919-1936.
- Wang, Z.B., de Vriend, H.J., Stive, M.J.F., Townend, I.H., 2007. On the parameter setting of semi-empirical long-term morphological models for estuaries and tidal lagoons. In: C.M. Dohmen-Janssen, S.J.M.H. Hulscher (Eds.). *River, Coastal and Estuarine Morphodynamics*. Taylor & Francis, pp. 103-111.
- Wang, Z.B., de Vriend, H.J., Stive, M.J.F., Townend, I.H., 2008. On the parameter setting of semi-empirical long-term morphological models for estuaries and tidal lagoons. In: C.M. Dohmen-Janssen, S.J.M.H. Hulscher (Eds.), *River, Coastal and Estuarine Morphodynamics*. Taylor & Francis, Utrecht, pp. 103-111. [<https://doi.org/10.1201/noe0415453639-c14>]
- Wang, Z.B., Lodder, Q.J., Townend, I.H., Zhu, Y., 2024. Future sediment transport to the Dutch Wadden Sea under severe sea level rise and tidal range change. *Anthropocene Coasts*, 7(1). [10.1007/s44218-024-00044-y]
- Wang, Z.B., Townend, I.H., Stive, M.J.E., 2014. Modelling of morphological response of tidal basins to sea-level rise revisited, Proceedings of the 17th Physics of Estuaries and Coastal Seas (PECS) conference, Porto de Galinhas, Pernambuco, Brazil.
- Yu, Q., Wang, Y., Gaio, J., Gao, S., Flemming, B., 2014. Turbidity maximum formation in a well-mixed macrotidal estuary: The role of tidal pumping. *Journal of Geophysical Research: Oceans*, 119, 7705-7724. [10.1002/2014JC010228.]

Appendix A – Function call sequence in AsmitaModel

The following table provides a listing of the calls made in AsmitaModel and the dependency of the functions called on other (sub) functions, along with a summary of input and output parameters.

ID	Calls in AsmitaModel.runModel	First Sub-function calls	Further Sub-function calls	Inputs	Outputs	Depends on
	is ValidModel			From initial model setup		0
A	AsmitaModel.CheckInput					
B	initialise run time properties, setRunParam, setIncParam					
C	AsmitaModel.InitialiseModel					
1	not needed - new instance	Results.clearRunResults			Cases.Year,RunData,EleData set to []	0
2		ASM_model.setASM_model			instantiates AsmitaModel	0
		AsmitaModel.initialiseModelParam - this is also used to initialise various	gui functions (eg Response)			
		Estuary.setReachTypes		incFDreach	sets mobj.Rctypes and mobj.Retypes	0
3		Element.initialiseElements		initial volumes and areas	all element transient properties except EqConcentration and EleConcentration	0
4		Element.setEqConcentration		transEleType and Eq. Conc.	assign cE to elements (allows for mixed fractions)	0
5		WaterLevels.setWaterLevels		water level properties at t=0	transient water level properties at t=0	0
7		Estuary.initialiseDispersionGraph		D, dExt	ReachGraph to define connectivity and dispersion	0
8		Advection.initialiseTransients		Q, qln, qOut	Initialise transient properties and flow graph for each type of advection	0
		Advection.initialiseRiverGraph				
		Reach.setReach				
10			Reach.setReachProps	(reachEleID)	assign reach id to elements (for plotting)	0
11				Moving volume and area, WLS	HW/LW volume, area, prism, csa, uriv,	3,5
12		Advection.setTidalPumping		RiverGraph, (qtp, qtpln, qtpOut)		
13			Advection.getTidalPumpingDischarge	h, ws", xi, avCSA, uriv, +props	qtp, qtp0 for each reach	3,5,11
14		Interventions.initialiseInterventions		user defined changes in area and volume for any given year	Unique years and array of change for each element (dv and ds)	0
15		Saltmarsh.initialiseMarshDepthConc		EleType,cE,ws,S2,M2,zhw,zlw,	MarshDepthConc	0
16			Saltmarsh.singleBioSettlingRate	dmx,bmx,wsa,wsb,Bc, (depth,aws1,aws2)	ws'	0
17			Saltmarsh.BiomassCoeffs	dmn, dmx, bmx	Bc	0
25		Element.setEleAdvOffsets		Offset flags, n, B and dd	eqAdvOffSet advection offset correction	40+41(using 7&8)
9			ASM_model.setDQmatrix	cE, kCel	DQ and dqln	7,8,12
45			Advection.getAdvection Flow	uses advection graphs	Q, qln and qOut for each type of	
18		ASM_model.asmitaEqFunctions		alpha, beta, EleType, Prism, tr	EqVolume, ve, EqSA, se	3,11
			ASM_model.asmitaEqVolume	alpha, beta, EleType, Prism, tr	EqVolume, ve, EqSA, se	
			ASM_model.asmitaEqHypsometry	alpha, beta, EleType, Prism, tr	EqVolume, ve, EqSA, se	
			Saltmarsh.EqDepths	dmn,dmx,dslr,tr,cE,sm,vm,(ws')	Deq'	3,6
19			Saltmarsh.BioSettlingRate	ws, sa,vm,ve,bmx,wsa,wsb,(Bm)	ws'	3
20			Saltmarsh.Biomass vm, sa, (Bc)		Bm	3
21			Saltmarsh.Morris kbm, qm, dslr, (Bc)		Deq	0
22			Saltmarsh.BiomassCoeffs	dmn, dmx, bmx	Bc	0
26		Element.setEqScalingCoeffs		initial volumes, (EqVolume)	eqScaling coefficient	3
18			ASM_model.asmitaEqFunctions	alpha, beta, EleType, Prism	EqVolume based on initial prism	3,11
			Saltmarsh.EqDepths+subfunctions	dmn,dmx,dslr,tr,cE,sa,vm,(ws')	Deq'	3,6
27		Element.setEquilibrium		eqScaling, eqAdvOffSet	EqVolume, ve, corrected for any scaling	2, 12
18			ASM_model.asmitaEqFunctions	alpha, beta, EleType, Prism	EqVolume, ve	3,11
19			Saltmarsh.EqDepths+subfunctions	dmn,dmx,dslr,tr,cE,sa,vm,(ws')	Deq'	3,6
24		ASM_model.setVertExch		vm,vf,Vo,ve,cb,n,(ws'),(B),(dd)	ws''	9,19,40+41(using 7&8)
20			Saltmarsh.BioSettlingRate	ws, sa,vm,ve,bmx,wsa,wsb,(Bm)	ws'	3
28		Element.setEleConcentration			assign concentrations, conc, to elements	0
29			ASM_model.asmitaConcentrations	vm, ve, sa, ws", n, D, dExt, Q, cE, qln, kCel	array of concentrations, conc	3,18,24
30		AsmitaModel.CheckStability		vm, D, Q, cnc, NumSteps, TimeStep, outInt	Runsteps, delta, outInt	0, 5, 15
		ASM_model.MassBalance		delta, Time, dExt, cE, cb, cl, cnc, qln,	dWLVOLUME, SedMbal, WatMbal	E0
31		AsmitaModel.PostTimeStep		outInt		
32			Results.setResults	WLS and vm, vf, ve, cnc, dWL	initialise Results if not instantiated, populate Year, RunData, EleData	3,6,11,18
D	AsmitaModel.InitTimeStep					
5		WaterLevels.setWaterLevels		jt, delta, Time, DateTime	iStep, Time, DateTime	main
33		Interventions.setAnnualChange		water level properties at t=t(i)	transient water level properties at t=t(i)	-
34		Estuary.updateDispersionGraph	for dynamic elements allowing horizontal exchange to be varied	Year,dV,dS,vm,vf,sa,sf	vm,vf,sm,sf	-
41		Advection.updateAdvectionGraphs	update if timeseries input	tsyear, AdvGraph	updated river or flow graph	
11		Reach.setReachProps		Moving volume and area, WLS	HW/LW volume, area and prism at t=t(i)	5,33,34,8
		Element.setEleWLchange		dHWchange,dLWchange, EleType	assign WL change based on element	-
12		Advection.setTidalPumping		RiverGraph, (qtp, qtpln, qtpOut)		8
13			Advection.getTidalPumpingDischarge	h, ws", xi, avCSA, uriv, +props	qtp, qtp0 for each reach	11+E-1
27		Element.setEquilibrium		eqScaling, eqAdvOffSet	EqVolume, ve, corrected for any scaling	11+E-1
18			ASM_model.asmitaEqFunctions	alpha, beta, EleType, Prism	EqVolume, ve	11+E-1
19			Saltmarsh.EqDepths+subfunctions	dmn,dmx,dslr,tr,cE,sa,vm,(ws')		6+E-1
37		Element.setEqConcentration	Not implemented: only needed if cE is varied during run			
9		ASM_model.setDQmatrix		cE, kCel	DQ and dqln	34,41
45			Advection.getAdvection Flow	uses advection graphs	Q, qln and qOut for each type of	
24		ASM_model.setVertExch	update if timeseries input	AdvGraph updated for tsyear		41
			Advection.getSubVarMatrix	vm,vf,Vo,ve,cb,n,(ws'),(B),(dd)	ws''	9,E-1,40+41(using E-1)
20			Saltmarsh.BioSettlingRate	ws, sa,vm,ve,bmx,wsa,wsb,(Bm)	ws'	E-1
28		Element.setEleConcentration			assign concentrations, conc, to elements	0
29			ASM_model.asmitaConcentrations	vm, ve, sa, ws", n, D, dExt, Q, cE, qln, kCel	array of concentrations, conc	3,18,24
E	AsmitaModel.RunTimeStep					
38		ASM_model.asmitaVolumeChange		dwl, vm, vf, ve, sa, n, cb	assign transient values of vm,vf,vb,conc to elements	6,E-1
29			ASM_model.asmitaConcentrations	vm, ve, sa, ws", n, D, dExt, Q, cE, qln, kCel	conc	E-1,24
40			ASM_model.BddMatrices	sa,ws",n,Q,cE,(DQ,dqln)	B and dd	E-1
39			Saltmarsh.BioProduction	kbm,(Bm)	Bk	
21			Saltmarsh.Biomass vm, sa, (Bc)		Bm	E-1
			getEleSpeciesProduct	SpeciesProduct	kbm	
		ASM_model.asmitaAreaChange		dwl, sa, se, n, cb	assign transient values of sa	
			ASM_model.FeeMatrices	sa,ws",n,Q,cE,(DQ,dqln)	F and ee	E-1
42			ASM_model.MassBalance	delta, Time, dExt, cE, cb, cl, cnc, qln, qOut, Vo, vf, vm, vb, shw, dwl	dWLVOLUME, SedMbal, WatMbal	E
F	AsmitaModel.PostTimeStep					
32		Results.setResults		iStep, outInt		D
				WLS and vm, vf, ve, cnc, dWL	initialise Results if not already instantiated, populate Year, RunData,	E
H	AsmitaModel.runModel					
43		dstable		Year, EleData, RchData		32
44				sedMbal, WatMbal	message to user	42

Appendix B – Erratum

Erratum to flow equation in paper by Townend et al, 2016, Part 1.

With some re-arrangement it can be shown that $dV/dt \Rightarrow 0$ if the equilibrium volume is factored by:

$$\alpha = \left(\frac{k \cdot q_r + \delta_E}{q_o + \delta_E} \right)^{1/n} \quad (\text{eq.42a})$$

The adjusted equilibrium is then given by:

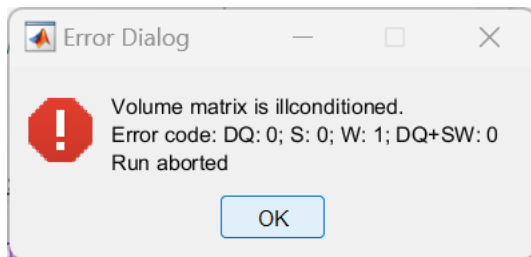
$$\frac{dV}{dt} = C \left\{ (q_o + \delta_e) \left(\frac{\alpha \cdot V_e}{V} \right)^n - (k \cdot q_r + \delta_e) \right\} \quad (\text{eq.42b})$$

The drift expression is correct.

Appendix C – Matrix Condition Checks

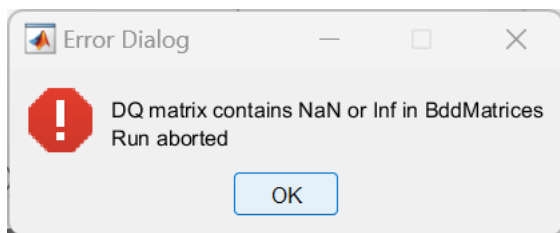
Checks are made when computing the matrices that are set up in ASMInterface class methods BddMatrices, FeeMatrices and asmitaConcentrations (see Townend et al, 2016, Part 1 for equations used).

Check whether matrices are ill conditioned for inversion. Checks the dispersion/flow matrix, DQ, the surface area, S, the vertical exchange, W and the combined DQ+SW matrix. Values are true if the matrix is ill-conditioned. For surface areas the matrices are the same but amended for depth as detailed in Appendix 1 of Townend et al (2016, Part 2).



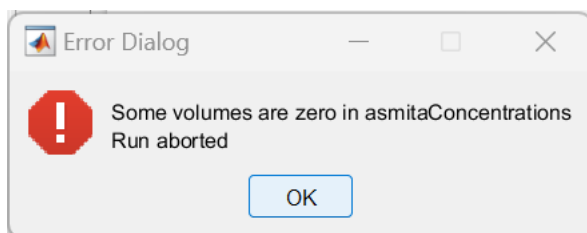
Usually means that some values in the matrix have grown to be very large, very small or both. For surface area the issue can also be connected to the element depths that are being used to define the matrices.

Check whether the Dispersion/Flow matrix contains infinite or Nan entries.



One possible cause is when Advection.getDriftFlow fails to converge.

Check whether $G_{am} = (v_e/v_m)^n$ are infinite or NaN in asmitaConcentrations.



Possibly means one or more elements have zero volume.