

## Preface

ChannelForm is a Matlab™ App that provides a framework for studying estuary and inlet morphological landforms in the context of the surrounding landscape (valley and flood plain). A set of tools allow digital terrain models (DTMs) to be created from models or imported. The models/data can define the channel form and the surrounding or antecedent landscape. A set of grid manipulation tools allow the grids to be manipulated and combined to generate composite DTMs of the channel in the landscape. Several idealised models are provided that use some combination of the plan form and cross-section to generate a 3D form, or generate the form using parameters that are not a function of the estuary itself (i.e. they are exogenous). In both cases, the water level surface can be defined using plane surfaces, or derived from a quasi-analytical hydrodynamic model that accounts for tides and river flows. Using channel and valley forms, marine transgression as a function of sea level rise can also be examined. This identifies the horizontal transgression of the initial form to achieve a net mass balance when any exchange with the external environment is accounted for. The output includes the gross properties of the channel, various well-known parameter relationships, and a time series of DTMs.

## Requirements

The model is written in Matlab™ and provided as Open Source code (issued under a GNU General Public License) and runs under v2016b or later. ChannelForm uses the `muitoolbox` and `dstoolbox`. To include hydrodynamic water surfaces, the `CSTmodel` App also needs to be installed.

## Resources

The ChannelForm App, `CSTmodel` App and the two toolboxes (`muitoolbox` and `dstoolbox`) can be downloaded from [www.coastalsea.uk](http://www.coastalsea.uk).

*Cite as:*

Townend, I.H., 2021, ChannelForm manual, CoastalSEA, UK, pp84, [www.coastalsea.uk](http://www.coastalsea.uk).

## Bibliography

Townend I H and Pethick J, 2002, Estuarine flooding and managed retreat. *Phil.Trans.R.Soc.Lond.A*, 360 (1796), 1477-1495.

Townend I H, 2010, An exploration of equilibrium in Venice Lagoon using an idealised form model. *Continental Shelf Research*, 30 (8), 984-999.

Townend I H, 2012, The estimation of estuary dimensions using a simplified form model and the exogenous controls. *Earth Surface Processes and Landforms*, 37, 1573-1583.

Townend I H, Zhou Z, Guo L and Coco G, 2021, A morphological investigation of marine transgression in estuaries. *Earth Surf. Process. Landf.*, 46, 626–641, <https://doi.org/10.1002/esp.5050>.

## Acknowledgements

The ChannelForm App makes use of several functions from the Matlab™ File Exchange Form, including:

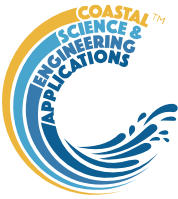
InterX - NS (2010). <https://www.mathworks.com/matlabcentral/fileexchange/22441-curve-intersections>

xy2sn and sn2xy –Bart Vermeulen (2022). Cartesian to Curvilinear coordinate forward and backward transformation. <https://www.mathworks.com/matlabcentral/fileexchange/55039-cartesian-to-curvilinear-coordinate-forward-and-backward-transformation>



## Revision history

Version	Date	Changes
3.4	Aug 2024	Added inlet form model to represent rectangular and elliptical tidal inlets.
3.3	Jan 2024	Solver for the CKFA model updated. Plot outputs modified to include additional information.
3.2	July 2023	Added option to plot 2/3D lines (e.g., centre-lines)
3.1	Oct 2022	Added functionality to use real DTM grids alongside idealised model grids. Added option to edit grid position properties
3.0	Dec 2021	Ported to work as an App using dstoolbox and muitoolbox. Added range of grid tools and options to include a shoreface and meanders
2.1	Jan 2020	Completed addition of transgression model
2.0	July 2019	Updated to be compatible with ModelUI v2. Added grid import export facilities. Improved compatibility between form options. Added statistical comparisons and additional plot options
1.0	Jan 2019	First release via email distribution



## Contents

1	Introduction .....	1
2	Getting started .....	1
2.1	Configuration.....	1
2.1.1	Installing the toolboxes .....	1
2.1.2	Installing the App .....	1
2.2	Model Set-up .....	2
3	Application Menus .....	4
3.1	File.....	4
3.2	Tools.....	4
3.3	Project.....	4
3.4	Setup.....	5
3.4.1	Form Parameters.....	5
3.4.2	System Parameters .....	8
3.4.3	Run Parameters.....	11
3.4.4	Import Data.....	11
3.4.5	Grid Tools.....	12
3.4.6	Form Properties and Model Constants .....	13
3.5	Utilities .....	13
3.5.1	Multi-feature composition .....	14
3.6	Run .....	15
3.7	Analysis .....	17
3.7.1	Plotting .....	17
3.7.2	Statistics.....	19
3.7.3	Transgression Plots.....	24
3.8	Help .....	24
3.9	Tabs .....	24
3.10	UI Data Selection .....	25
3.11	Form and Transgression Model Output Table Content .....	26
3.11.1	Form Table .....	26
3.11.2	Hypsometry Table .....	26
3.11.3	SectionProps Table.....	27
3.11.4	Plan Table.....	27
3.11.5	GrossProps Table .....	28
3.11.6	WaterLevels Table .....	28
3.11.7	Transgression Table .....	29
3.12	Accessing data from the Command Window .....	29



4	Background to Form Models.....	31
4.1	Basis of the Exponential Form model .....	32
4.1.1	Longitudinal plan form.....	32
4.1.2	Low water channel .....	32
4.1.3	Intertidal flat profile .....	33
4.2	Basis of the Power Law model.....	34
4.3	Basis of the ‘CKFA’ model.....	37
4.3.1	Longitudinal Plan form .....	38
4.3.2	Low water channel .....	38
4.3.3	Tidal flat profile.....	38
4.3.4	Idealised 3D form.....	40
4.3.5	Dimensions of a tidal system.....	40
4.3.6	Gross properties.....	43
4.3.7	Influence of waves.....	43
4.4	Basis of the Tidal Inlet model .....	45
4.5	Adding variations in along-channel water level surface .....	45
4.6	Valley Model.....	46
4.7	Combined channel and valley form.....	46
4.8	Shoreline form.....	47
4.9	Meanders .....	48
5	Background to Transgression Model.....	49
5.1	Model Formulation.....	50
5.2	Water Levels.....	52
5.2.1	Tidal Amplitude .....	52
5.2.2	Sea Level Rise .....	52
5.2.3	Hydrodynamics .....	53
5.3	Sediment Flux.....	53
5.4	Shoreline Change .....	54
5.5	Constraints.....	54
5.6	Model Configuration .....	55
5.6.1	System dimensions and properties .....	55
5.6.2	Hydraulic parameters .....	55
5.6.3	Minimum data requirements for models .....	55
5.6.4	Kinematic and Quasi-dynamics models .....	56
5.7	Model Convergence .....	56
5.8	Study Workflow .....	56
5.8.1	Quick view .....	57



---

5.8.2	Valley model .....	57
5.8.3	Create conformal channel to extract channel properties .....	58
5.8.4	Create Form model.....	58
5.8.5	Holocene Transgression .....	58
6	Supporting Information .....	60
6.1	Derive Output.....	60
6.1.1	Calling an external function .....	61
6.1.2	Input and output format for external functions.....	61
6.1.3	Pre-defined functions .....	63
7	Program Structure.....	65
8	Bibliography.....	69
	Appendix 1 – Sample Output for demo.mat file .....	73
	Appendix 2 – Grid model property error estimates.....	75
	Appendix 3 – Channel form solver equations .....	78

## 1 Introduction

ChannelForm App provides a simplified form model that uses an idealized 3D description of the channel and intertidal flats based on the following components, all of which derive from ‘energy flux equilibrium’ arguments of one sort or another:

- (i) a longitudinal form described by exponentially converging cross-sectional area
- (ii) a low water channel described by a parabolic variation in depth, based on the most probable distribution of transverse slope
- (iii) an intertidal profile that has uniform bottom shear stress across the profile as the tide varies.

By combining the exponential along estuary variation of width, with the descriptions of the channel and intertidal, a 3D description of the estuary is obtained. The detailed formulation is given in Townend (2010) and the application to UK estuaries in Townend (2012).

Channel Form includes tools to combine the estuary form within an idealised valley, or valley DTM (if relatively straight). Various tools to examine the dimensions of a regime river channel and the magnitude of the sediment flux with the outside environment are incorporated into the App. These various inputs are used in the Transgression model to examine the concept of estuary migration landwards under sea level rise as proposed by Allen (1990) and presented in Townend et al. (2020).

## 2 Getting started

### 2.1 Configuration

ChannelForm is installed as an App and requires `multoolbox` 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](#)

`multoolbox`: [multoolbox.mltbx](#)

The App file: [ChannelForm.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 App 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:

```
>> ChannelForm;
```

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

Documentation can be viewed the Help menu option in the App, or from 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 Model Set-up

The main steps in a typical workflow are outlined below. Further details about the input menus can be found in Section 3 and the background to the model and supporting information about model configuration are provided in Sections 4 and 5.

*File>New* to create a new project space.

To create a basic channel morphology the following parameters need to be defined:

*Setup>Form Parameters>Exp Form Parameters*: channel and intertidal form

Or *Setup>Form Parameters>Power Form Parameters*: channel and intertidal form

And

*Setup>System Parameters>Hydraulic Parameters > Tidal Forcing*: water levels at boundary

*Setup>System Parameters>Sediment Parameters*: sediment parameters for estuary and river

*Setup>Run Parameters>Grid Parameters*: model grid definition

To add a water surface variation along the channel requires:

*Setup>System Parameters>Hydraulic Parameters > Hydraulic Model*: tidal propagation

To create a valley form requires:

*Setup>Form Parameters>Power Form Parameters*: dimensions of valley

And *Sediment Parameters*, *Grid Parameters*, *Hydraulic Model* as above

The transgression model requires a Channel Model and a Valley model to have been created (or a grid model loaded for the valley) and in addition:

*Setup>System Parameters>Transgression Parameters*: settings for the transgression model.

*Setup>Run Parameters>Run Time Parameters*: model time step parameters

In addition, fixed adjustments to the idealised model form can be imposed using the parameters defined from *Setup>System Parameters>Morphological Modifications*.

Create a Channel Form or a Valley Form:

*Run>Exponential form model* - run the exponential model,

OR *Run>Power form model* - run the power model,

OR *Run>CKFA form model* - run the CKFA model.

*Run>Valley form model* - run the Valley model.

Once a Channel and Valley Form have been created they can be combined using *Utilities>Add Form to Valley*. Other utilities allow the influence of the various parameter settings to be examined (such as valley slope, morphological time scale, flood plain area, etc).

Marine transgression is then examined using:

*Run>Transgression model* - run the transgression model. Prompts to select source channel and valley models to use.

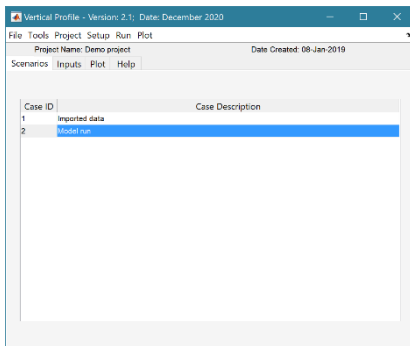
Note: changing the Input parameters that determine the initial channel and valley forms (using the Setup UIs) has no effect when running the transgression model. To change the settings, new initial forms need to be created as inputs to the transgression model. However, some Input parameters settings do not influence the initial form and these can be modified, when prompted, at run time. These include, sea level rise, tidal range cycles, and the coefficients used in the sediment flux calculations.

The run is listed on the *Cases* tab, the resultant DTM can be viewed on the *Q-Plot* tab and a range of summary parameters and plots can be viewed on the *Form-Props* tab.

### *Plot>Plot menu*

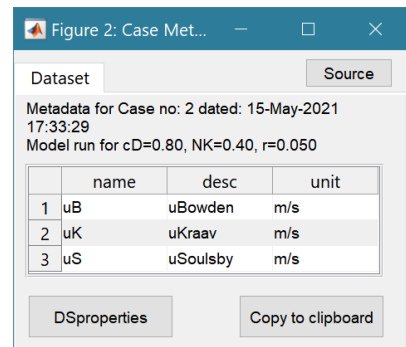
The results from a run can be selected and plotted. By using the Add button additional model runs can be included on the plot, allowing different Cases to be compared.

The meta-data associated with a particular case can be viewed by clicking the mouse on the Case ID of interest on the *Cases* tab. This displays a summary of output variables, with buttons to access the data source(s) and data set properties (DSproperties), and copy the tabulated data to the clipboard.



### Default User Interface

Left click mouse on a Case ID to see the associated meta-data (as shown on the right)





### 3 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.

#### 3.1 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.

#### 3.2 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.

#### 3.3 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 (eg 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 (eg text file, Excel, etc), use the *Project>Cases>Edit Data Set* option to make a selection and then use the ‘Copy to Clipboard’ button to paste the selection to the clipboard.

### 3.4 Setup

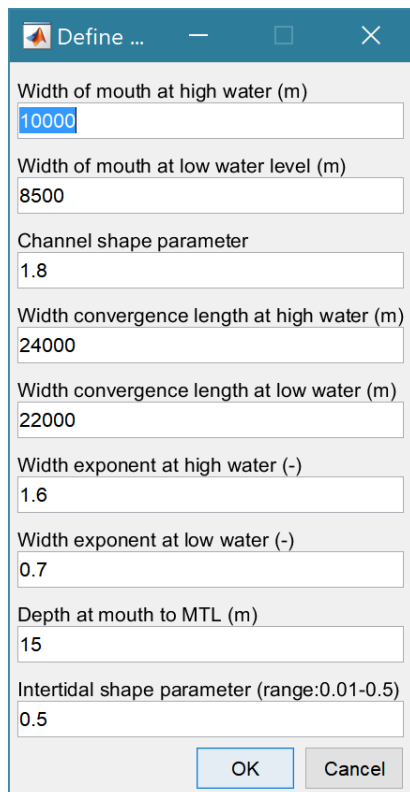
The setup menu provides a series of menus to enable different components of the model to be defined.

#### 3.4.1 Form Parameters

*Setup>Form Parameters:* enter and edit the specified model parameters.

*Note: the CKFA model input parameters are derived from the water level, hydrodynamic and sediment parameter settings.*

*Setup>Form Parameters>Exp Form Parameters:* UI to input the parameters needed for the exponential form model are as follows:



Widths for high and low water at the mouth

A shape parameter for the channel cross-section. A value of 2 is a parabola ( $>0$ ,  $1 \rightarrow$ linear,  $>5 \rightarrow$ rectangular)

Exponential plan form convergence lengths at high and low water

Power law plan exponents at high and low water\*

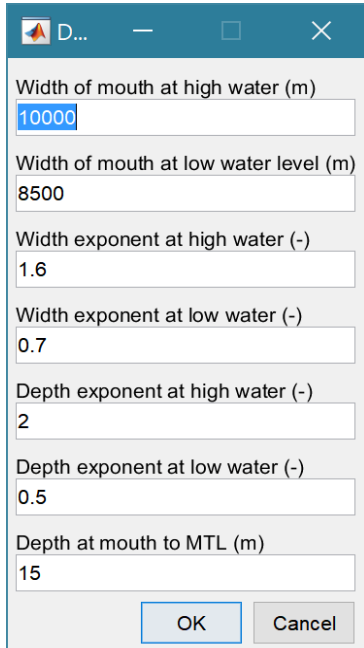
Depth of bed in centre of channel at the mouth measured from mean tide level.

Intertidal shape parameter (for L&M intertidal profile)

[\*this only applies to the plan form as an alternative to the exponential plan form (set to 0 if not required). The Power Form model uses both width and along channel depth exponents].

Note: some additional parameters need to be defined for the model to run. These include the distance to tidal limit, which is set using *Setup>System Parameters>Hydraulic Parameters>Hydraulic Model*, and tidal amplitude and mean sea level, which are set using *Setup>System Parameters>Hydraulic Parameters>Tidal Forcing*. A grid must be defined using *Setup>Run Parameters>Grid Parameters*. If a marsh is to be included the marsh parameters need to be defined using *Setup>System Parameters>Sediment Parameters*. Similarly, if a river channel is to be included, the river discharge is defined using *Setup>System Parameters>Hydraulic Parameters>Hydraulic Model*, and the river bed properties (grain size and critical shear stress) using *Setup>System Parameters>Sediment Parameters*.

*Setup>Form Parameters>Power Form Parameters*: UI to input the parameters needed for the power form model.



Widths for high and low water at the mouth

Power law width exponents at high and low water

Power law depth exponents at high and low water

Depth of bed in centre of channel at the mouth measured from mean tide level.

Note: some additional parameters needed to be defined for the model to run. These include:

- the distance to estuary/river switch and distance to tidal limit which are set using *Setup>System Parameters>Hydraulic Parameters > Hydraulic Model*, and
- tidal amplitude and mean sea level, which are set using *Setup>System Parameters>Hydraulic Parameters > Tidal Forcing*.
- A grid must be defined using *Setup>Run Parameters>Grid Parameters*.
- If a marsh is to be included the marsh parameters need to be defined using *Setup>System Parameters>Sediment Parameters*.
- If a river channel is to be included the river discharge is defined using *Setup>System Parameters>Hydraulic Parameters>Hydraulic Model*, and the river bed properties (grain size and critical shear stress) using *Setup>System Parameters>Sediment Parameters*.



*Setup>Form Parameters>Inlet Form Parameters*: UI to input the parameters needed for the inlet form model.

Widths for high and low water at the mouth

Maximum width of basin

Width convergence length at low water

Width power law exponent at low water

Depth at mouth inlet to mean tide level

Length of mouth inlet (spit/island or jetty constrained) from mouth

Length of the low water channel from the mouth

Length of basin at high water from the mouth

Power law depth exponents at high and low water

Shape parameter for the channel cross-section. A value of 2 is a parabola ( $>0$ ,  $1 \rightarrow$ linear,  $>5 \rightarrow$ rectangular)

Intertidal shape parameter (for L&M intertidal profile)

Width of mouth at high water (m)	2400
Width of mouth at low water level (m)	1800
Width of basin at high water (m)	19000
Width convergence length at low water (m)	3000
Width exponent at low water (-)	0.5
Depth at mouth to MTL (m)	14
Length of mouth inlet (m)	1700
Length of LW channel (m)	12000
Length of basin at high water (m)	17000
Channel shape parameter	2
Intertidal shape parameter (range: 0.01-0.5)	0.5

*Setup>Form Parameters>Valley Parameters*: UI to input the parameters needed for the valley form model.

Width of the valley at the mouth

Maximum depth of the valley at the mouth

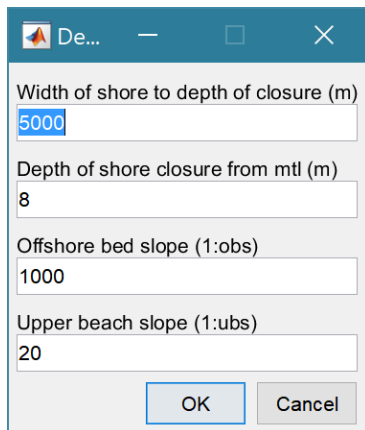
Elevation of the water level at the tidal limit

The total length of the valley

The elevation at the head of the valley

Maximum elevation to be displayed

Valley width at mouth (m)	20000
Valley depth at mouth (mAD)	-15
Distance to tidal limit (m)	140000
Elevation of tidal limit (mAD)	3
Distance to head of valley (m)	300000
Elevation of head of valley (mAD)	300
Cut-off valley elevation (mAD)	12



*Setup>Form Parameters>Shore Parameters:* UI to input the parameters needed for the shore profile form model.

shore normal width to depth of closure (m)

closure depth of profile from mtl (m)

offshore bed slope - defines bed alope of channel

upper beach slope - defines slope above 0mAD

### 3.4.2 System Parameters

*Setup>System Parameters>Hydraulic Parameters > Tidal Forcing:*

UI to input water level parameters including sea level rise and any cyclic variations in tidal range.

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

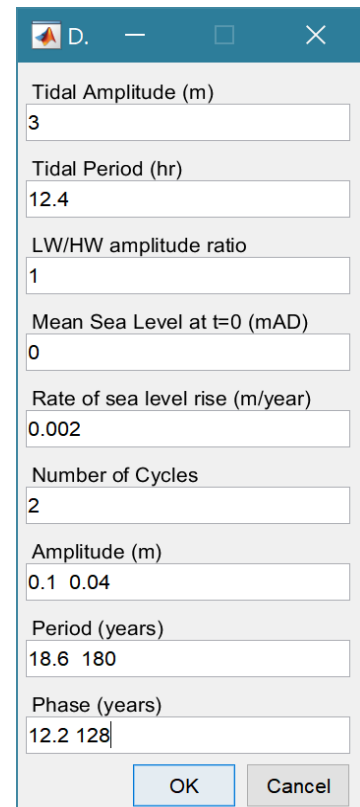
The LW/HW amplitude ratio accounts for any difference in the amplitude to low water and high water relative to the define mean sea level<sup>1</sup>. Used to adjust influence of changes in tidal range.

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

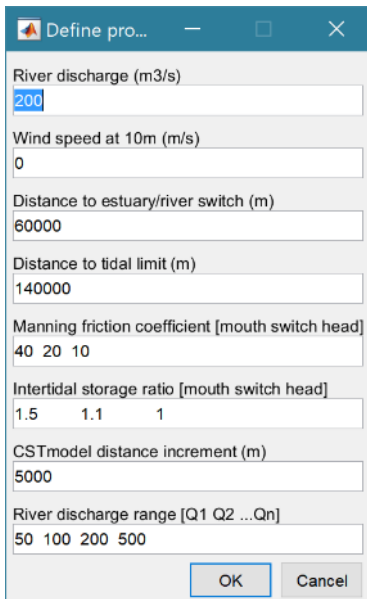
Rate of sea level rise as a linear rate (positive or negative). This can also be used to define an exponential rate of slr or to call a user function. See Section 5.2.2 for the details of how to use these options.

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 have to agree with the number of cycles specified. The phase defines a shift in the cycle to align with the Julian calendar. This is only or relevance if the model is being used to hindcast change and compare volume changes with observed changes (e.g. Townend et al., 2007).

Note: the Water Level UI has a second page for entering certain tidal constituents but these are **not** used in the current version of ChannelForm.



<sup>1</sup> Option included because of observed bias (eg on 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>System Parameters>Hydraulic Parameters > Hydraulic Model:* UI to input parameters needed for the CST hydraulic model.

River discharge

Wind speed at a 10m elevation above mean sea level (to include wave effects in CKFA profile)

Distance from mouth to points at which the properties below are adjusted. These are supplemented by the mouth and head distances so the number of the following properties should be number of distances+2.

Manning friction coefficient as a set of linear along channel variations

The storage width ratio is the ratio of the width at high and low water and is also taken to vary linearly based on the defined values at the switch points.

The x-axis interval used in the CSTmodel. Reducing the value increases computation time and can make the model convergence over sensitive. Some trials may be needed depending on the size of

the system being modelled in ChannelForm and the grid resolution being used.

The river discharge range is used to run the CSTmodel for each value to create a look-up table which is then used to display the hydraulics for any discharge value within the range on the *Hydro-Props>Hydraulics* tab.

*Setup>System Parameters>Sediment Parameters:* UI to input parameters needed to determine external sediment exchange.

Characteristic grains size in the estuary channel<sup>2</sup>

Critical bed shear stress in the estuary channel

Mass erosion rate constant

Sediment density that determines the equilibrium condition in the sediment flux model

The wet or bulk density of the estuary channel bed

The transport coefficient used in the sediment flux model<sup>3</sup>

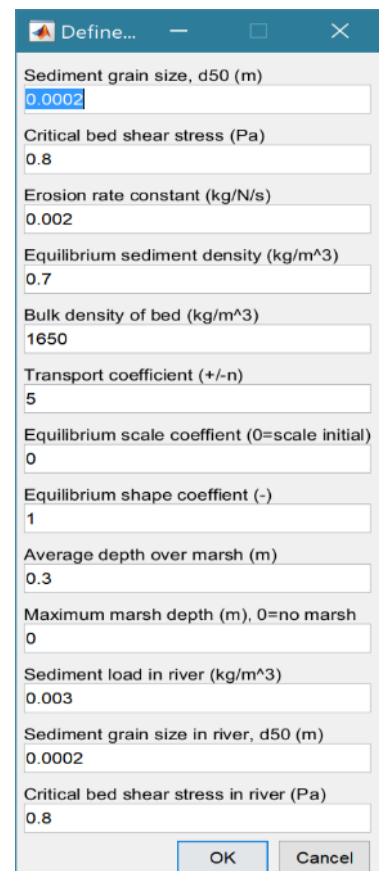
The shape,  $\alpha$ , and scale,  $\beta$ , coefficients define the equilibrium volume in the flux model, as a function of the tidal prism ( $\alpha.P^\beta$ )

Average depth of marsh is used to add the marsh platform and the maximum depth determines the seaward extent across the intertidal profile. No marsh is added if the maximum depth is zero.

Concentration density of the rivers sediment load.

Characteristic grains size in the river channel<sup>4</sup>

Critical bed shear stress in the estuary channel.

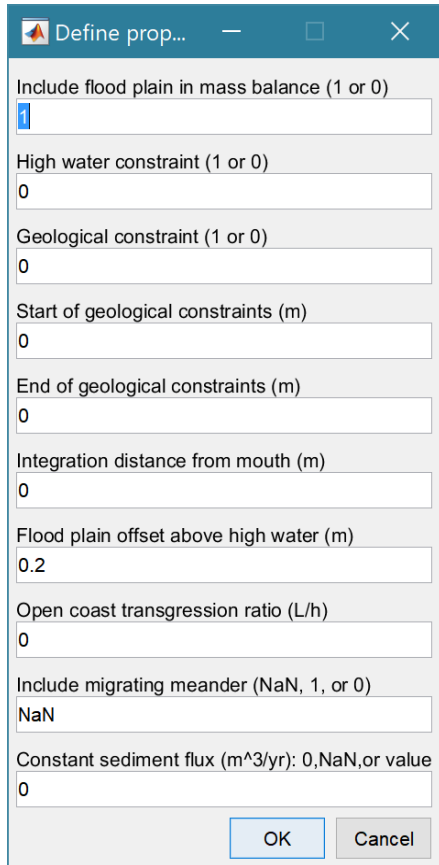


<sup>2</sup> Size, shear stress and erosion rates are used in CKFA model to estimate erosion/deposition balance.

<sup>3</sup> When suspended load transport dominates a value of 3 is representative, whereas a value of 5 is more representative when bed load transport dominates sediment exchanges.

<sup>4</sup> Size and shear stress are used in estimation of regime river channel dimensions.

**Setup>System Parameters>Transgression Parameters:** UI to set inclusion/exclusion of various constraints, computation parameters and open coast erosion settings.



Flag to include the flood plain in the mass balance calculation to estimate the transgression distance.

Flag to exclude the flood plain and prevent any erosion of the plan form at high water.

Flag to include geological constraints to prevent erosion over selected lengths of the channel (accretion is still possible).

Start distance from mouth for any bed constraints (*can be multiple*)

End distance from mouth for any bed constraints (*must be same number as Start constraints*)

Distance measured from mouth to calculate sediment mass balance. Uses tidal limit if distance is zero.

The ratio of the distance from the shoreline to the closure depth, L, to the closure depth, h. When multiplied by the rate of sea level rise this give a rate of shoreline retreat.

Option to exclude (NaN) or include a meander, if defined, allow the meander to translate (1) or be stationary (0), see Section 4.9 for more details.

Option to impose a constant sediment flux\*. If 0, sediment settings used to calculate flux at each time step. If >0 constant value is used. If NaN the kinematic model is used (see Section 5.6.4).

\* *can be modified at run time.*

**Setup>System Parameters>Morphological Modifications:** UI to input the x,y,z definitions of the, one or more, boxout forms that can be imposed on model forms. To include these modifications in a Channel Form use **Utilities>Add Modifications**.

The start distance(s) for the morphological modification(s)

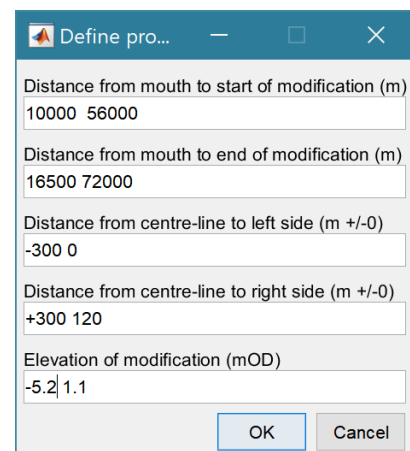
The end distance(s) for the morphological modification(s)

The distance(s) measured from the centre-line (0) to the right and left sides of the modification (checks that left distance is greater than right). This allows the modification to be offset to one side.

The elevation of the modification(s).

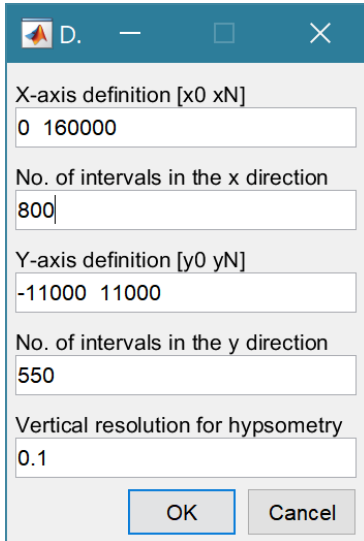
**Notes:**

1. This form of modification is **ONLY** included in the Transgression Model if it has already been added to the Channel Form that is used to define the initial channel using the **Utilities>Add Modifications** option.
2. Defines modifications that are spatially fixed. i.e. they are not subject to transgression.





### 3.4.3 Run Parameters



*Setup>Run Parameters>Grid Parameters:* The grid is defined by x and y ranges and intervals for both axes.

Upper and lower limits for the x co-ordinates

Grid spacing along the x-axis

Upper and lower limits for the y co-ordinates

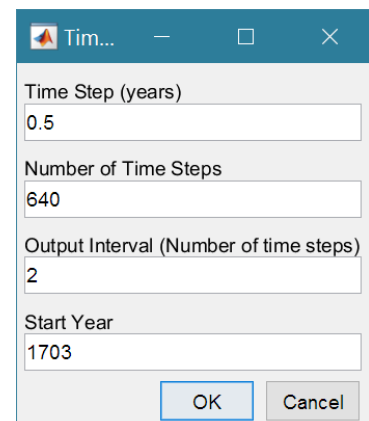
Grid spacing along the y-axis

The vertical interval to be used when computing the channel hypsometry.

*Setup>Run Parameters>Run Time Parameters:* The time step parameters comprise the time step (years or parts of a year) and the number of time steps. 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.

**Note:** The 'Output Interval' variable is not used in the ChannelForm model.

**Note:** the Start Year and format for the time in ChannelForm 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.

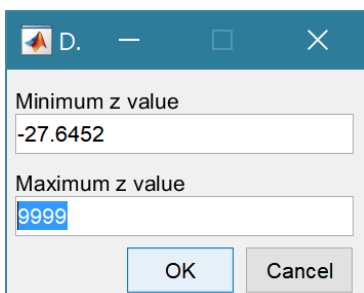


### 3.4.4 Import Data

*Setup>Import Data:* dialog with sub-menu options to Load, Add, Delete, Quality Control.

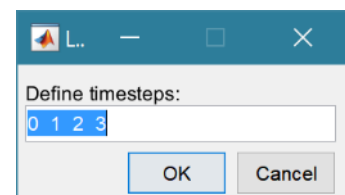
When the data has been loaded, the user is prompted to provide a description of the data set (Case) and this is listed on the *Cases* tab. The source file(s) can be checked by selecting the first column of a Case on the *Cases* tab and then the 'Source' button on the Meta-data table, to display a listing of the input files used.

*Setup>Import data> Load data:* Select one or more files to load. The first prompt is for the date-stamp of the file(s) being loaded. For a single file the default of 0 years can be used. For multiple files the default vector is 0 1 2 ...N and should be



edited to define the time sequence of the grids (assumed to be in years). The next prompt presents the maximum and minimum Z values in the grid and the

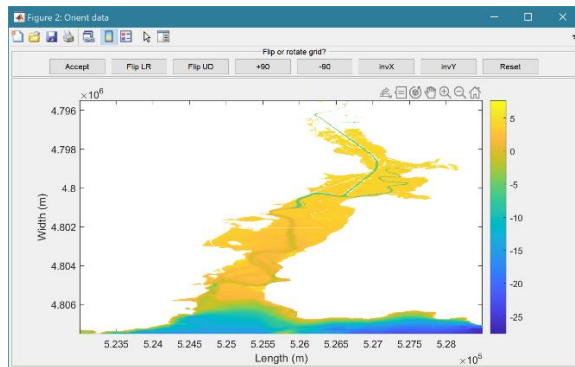
option to limit the range of data to be loaded (values outside the range are set to NaN). The grid is then plotted on a figure with multiple options to rotate or flip the grid, and invert the direction of the X and Y axes. This is illustrated below for the case of the Oka estuary in northern Spain. In (b) the orientation of the grid and axes



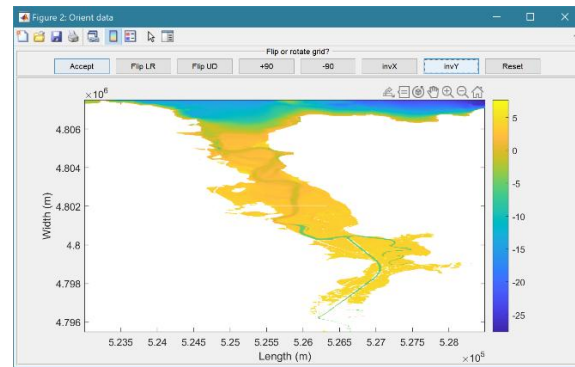


are geographically correct, whereas in (c) the estuary channel has been aligned with the X-axis, whilst keeping the grid coordinates correct relative to the channel (X-axis is descending). This is required if the Inlet Property tools for hypsometry and gross properties are to be used as these assume the channel is aligned to the X-axis.

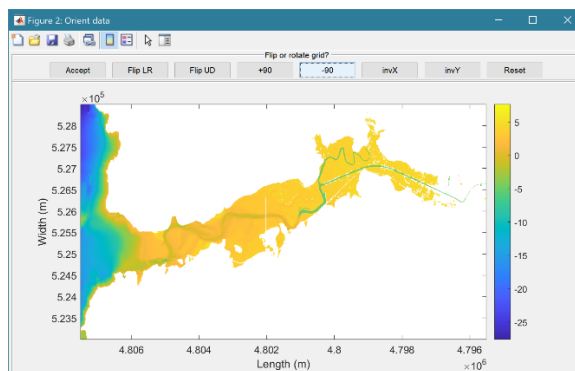
(a) raw data loaded from file



(b) grid flipped UD and Y-axis inverted to give correct geographical orientation



(c) grid flipped UD and rotated -90 to align estuary channel with x-axis (note X-axis is descending but correct geographically)



#### Button options:

Accept – uses the current settings to load grid  
 FlipLR – flips the grid left to right (horizontally)  
 FlipUD – flips the grid up-down (vertically)  
 +90 – rotates grid clockwise  
 -90 – rotates grid anti-clockwise  
 invX – reverses the direction of the X-axis  
 invY – reverses the direction of the Y-axis  
 Reset – restores the settings as loaded from file

*Setup> Import data> Gridded Data > Add:* prompts for file to be added (only one file at a time can be added) and the Case to use (if more than one Case). The user assigns timestep, range and adjust orientation as explained for loading grids above, Only files with the format used to create the data set can be used to Add data to a data record.

*Setup> Import data > Delete data:* prompts for Case from which some part of the data is to be deleted.

*Setup> Import data > Data QC:* runs a series of checks on the data. This is only available if defined for the specific data format.

### 3.4.5 Grid Tools

#### *Setup > Grid Tools*

A range of tools to manipulate cartesian grids.

*Setup>Grid Tools>Translate Grid:* interactively translate grid x-y coordinates.

*Setup>Grid Tools>Rotate Grid*: interactively flip or rotate grid.<sup>5</sup>

*Setup>Grid Tools>Re-Grid*: re-grid a gridded dataset to match another grid or to user specified dimensions.

*Setup>Grid Tools>Sub-Grid*: interactively define a sub-grid and save grid as a new Case.

*Setup>Grid Tools>Combine Grids*: superimpose one grid on another based on maximum or minimum set of values.

*Setup>Grid Tools>Add Surface*: add horizontal surface to an existing grid.

*Setup>Grid Tools>To curvilinear*: map grid from cartesian to curvilinear coordinates.

*Setup>Grid Tools>From curvilinear*: map grid from curvilinear to cartesian coordinates.

*Setup>Grid Tools>Display Dimensions*: display a table with the dimensions of a selected grid.

*Setup>Grid Tools>Difference Plot*: generate a plot of the difference between two grids.

*Setup>Grid Tools>Plot Sections*: interactively define section on a grid and plot as sections.

*Setup>Grid Tools>Digitise Line*: interactively digitise a line (with option to add elevations) using selected grid as base map.

*Setup>Grid Tools>Export xyz Grid*: select a Case and export grid as xyz tuples.

### 3.4.6 Form Properties and Model Constants

*Setup>Add Properties*: add form properties to a gridded data set. The user is prompted to input levels for HW, MTL and LW, or provide a file<sup>6</sup> to define along channel water levels. If there are multiple grids loaded for the selected Case, there is the option to use the same water levels for all grids, or specify for each grid. *Note: the grid must have the channel oriented along the x-axis and sections are taken normal to the x-axis. This can produce strange results for the section properties, such as width and cross-sectional area, if the channel meanders.*

*Setup>Delete Properties*: delete ALL property tables associated with a selected gridded data set.

*Setup> Edit Inlet Definition*: provides access to the definition of the position of the head of the inlet, distance from the x-axis origin to the mouth and any definition of the channel centre-line (if used).

*Setup>Input Data>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.

## 3.5 Utilities

*Utilities>Hydraulic Model* - runs the CSTmodel (if CSTmodel App is installed) for the range of inputs specified in *Setup>*

*Utilities>Add Form to Valley* - combines channel and valley grids (see Section 4.7).

*Utilities>Add Meander* - adds a user defined meander to an idealised channel form model. After selecting a model, the user is prompted to load a text file containing the x-y co-ordinates of the channel centreline. These are used to map the straight channel into curvilinear co-ordinates that follow

<sup>5</sup> Rotation re-orientates the z grid and swaps x and y axes when rotating  $\pm 90^\circ$ . However, it does not change the order of the x and y axes. Consequently, rotating  $+90^\circ$  or flipping left-right reverses the co-ordinate values on the x-axis and rotating  $-90^\circ$  or flipping up-down reverses the co-ordinate values on the y-axis.

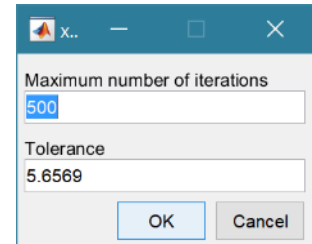
<sup>6</sup> File format is: single header line followed by 4 columns of numerical values for distance, high water level, mean tide level and low water level. The distance must cover the extent of the grid being used.

the centreline and then interpolate the results onto the cartesian model grid. Subsequent prompts are to include an offset distance from the mouth to the start of the meander centreline definition (see Section 4.9 for further details).

A prompt for the convergence parameters used in the mapping transformation, requires:

The number of iterations to be used in trying to fit the grid to the specified centre-line.

The distance tolerance to be used to test for solution convergence



*Utilities>Add Shoreline>Model Shore* - add a shoreline strip to a channel or valley model. This extends the grid seawards by the distance to the closure depth specified in *Setup>Form Parameters>Shore Parameters*. This comprises a beach profile using Dean's equilibrium model on the open coast and the defined offshore bed slope across the mouth of the channel (see Section 4.8 for further details).

*Utilities>Add Shoreline>Extrapolate Shore* - add a shoreline strip to a grid (imported or model). The user is prompted to define the width of the extension, the seaward or offshore depth and an exponent. An exponent of 1, gives linear extrapolation between the grid shore values and the defined offshore value, a value greater than one gives a shore that is concave down and a value less than one gives a shore that is convex up.

*Utilities>Add Modifications* - adds defined modifications to a grid.

*Utilities>Add Thalweg to Valley* - uses an xyz definition of a channel thalweg, to create a valley base in an existing model or imported grid. This is used to modify an existing valley grid, to extend the valley form down to the pre-Holocene surface (i.e. below the existing channel bed) using the defined the thalweg data. User is prompted for a thalweg file. This has a header of %f %f %f, followed by rows of x, y, z values that define the pre-Holocene valley bed. The upper cut-off, offshore mask width and valley base width must then be defined. The upper cut-off is used to mask out the channel in the valley grid in order to introduce the new channel base (thalweg). Typically values just above high water and below the deepest depth in the valley grid channel should suffice. The offshore mask width defines the width either side of the channel centreline to be adjusted to seaward of the existing estuary channel mouth. The valley base width replicates the thalweg depths for the defined width about the thalweg (i.e. +/- width/2).

*Utilities>River Dimensions* - calculates the river dimensions based on the current input parameters.

*Utilities>Valley Thalweg* - plots the valley centre-line along the x-axis.

*Utilities>Area of Flood Plain* - calculates the area of the flood plain.

*Utilities>CKFA Channel Dimensions* - calculates the governing dimensions that would be used in the CKFA model based on the current input parameters.

*Utilities>Morphological Timescale* - calculate the morphological timescale using the ASMITA model parameterisation and the current input parameters.

*Utilities>Channel-Valley Sub-Plot* - create a plot showing the channel, valley and combined form as 3 sub-plots.

*Utilities>Channel-Valley Sub-Plot* - create a plot of a selected x,y or x,y,z file that is being used to define a centre-line for a valley or channel (e.g. for use in curvilinear transformations).

### 3.5.1 Multi-feature composition

The options to combine any of the channel forms with a valley, meander, shoreline and modification can be used in any combination. However the order in which they are added can affect the outcome. For example, if a shoreline is added to a channel and then a valley is added, the valley extends to the

most seaward point of the shore. Whereas, if the channel and valley are combined and then the shore is added, the shoreline extends seaward of the whole model domain.

When a meander is being added to a channel with a shoreline, the location of the mouth relative to the x-axis grid origin forms part of the grid definition. Any offset to the meander centreline definition is added from the mouth and not the grid origin. Hence, there may not be a need to include an offset to the meander centreline unless the initial curvature is particularly severe.

### 3.6 Run

When model run has completed all models prompt for a Case description, which is added to the listing on the *Cases* tab.

*Run>Exponential form model* - run the exponential model. Prompts to select type of water surface (see below) and additional form selection properties to use.

The Form Selection properties available are:

*Plan Form*: Exponential or Power Law;

*Channel Form*: Parabolic or Rectangular;

*Intertidal Form*: Linear, Rectangular, Stepped, Uniform shear, and L&M muddy shore, where Uniform shear is based Friedrichs (1996) and L&M refers to the Lee and Mehta (1994; 1997).

See *Setup>Form Parameters>Exp Form Parameters* in Section 3.4.23.4.1, for details of input requirements.

*Run>Power form model* - run the power model. Prompts to select type of water surface to use (see below). See *Setup>Form Parameters>Power Form Parameters* in Section 3.4.23.4.1, for details of input requirements.

*Run>CKFA form model* - run the CKFA model. Prompts to select type of water surface to use (see below). This model requires:

- the tidal amplitude, tidal period and mean sea level to be defined using *Setup>System Parameters>Hydraulic Parameters>Tidal Forcing*;
- the distance to tidal limit which is set using *Setup>System Parameters>Hydraulic Parameters>Hydraulic Model*;
- the sediment grain size and critical bed shear stress must be set using *Setup>System Parameters>Sediment Parameters*;
- a grid must be defined using *Setup>Run Parameters>Grid Parameters*;
- if a marsh is to be included the marsh parameters need to be defined using *Setup>System Parameters>Sediment Parameters*;
- if a river channel is to be included the river discharge is defined using *Setup>System Parameters>Hydraulic Parameters>Hydraulic Model*, and the river-bed properties (grain size and critical shear stress) using *Setup>System Parameters>Sediment Parameters*.

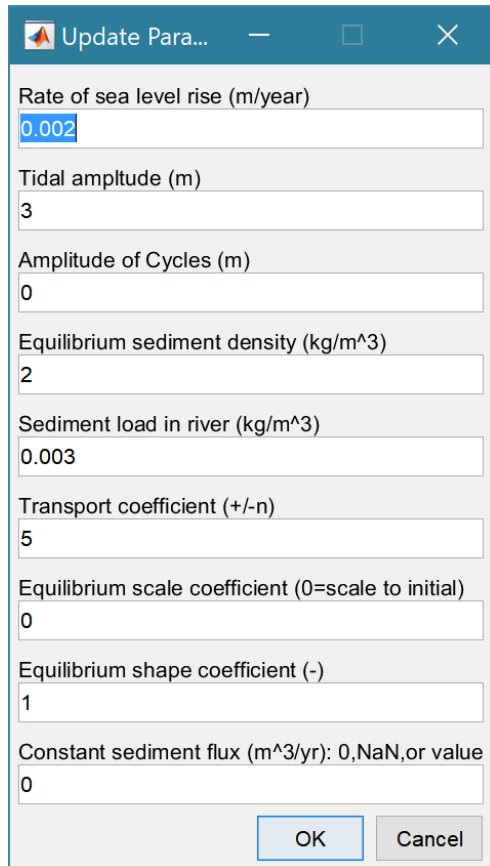
Water surface options are as follows:

- (i) the CSTmodel to define a kinematic water surface that takes account of tidal propagation and river flow;
- (ii) constant HW and a linear increasing LW (ie to meet HW at the tidal limit); and
- (iii) constant HW and LW.

*Run>Valley form model* - run the Valley model.

*Run>Transgression model* - run the transgression model. Prompts to select source channel and valley models to use. The transgression model is initialised with a selected channel and valley model and

copies the run parameters used to created the selected forms. A sub-set of these run parameters can be modified at run time, as follows:



Rates of sea level rise following one of the various options available as explained in Section 5.2.2.

The amplitude of any cycles to be included (Section 3.4.2, Tidal Forcing).

Sediment density that determines the equilibrium condition in the sediment flux model (see Section 3.4.2, Sediment Parameters).

Sediment contribution from the rivers (see Section 3.4.2, Sediment Parameters).

Transport coefficient: ~3 for coarse sediments and ~5 for fine sediments (see Section 3.4.2, Sediment Parameters).

The scale, a, and shape, b, coefficients define the equilibrium volume in the flux model, as a function of tidal prism ( $a.P^b$ ). If the scale coefficient = 0, the equilibrium volume is determined from the initial values of volume and prism, with a shape coefficient of 1.

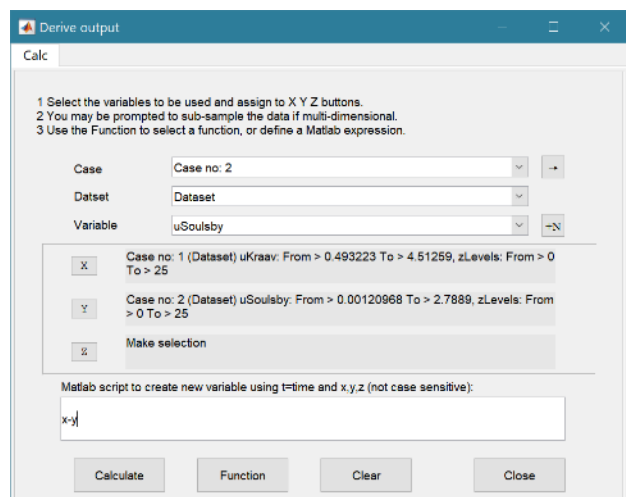
Option to impose a constant sediment flux\*.

*\*If set to 0, the sediment parameter settings are used to calculate the sediment flux at each time step to reflect the changing morphology and hydrodynamics. If >0 a constant value is used. If set to NaN a kinematic version of*

*the model is run. This uses a single time step based on the step duration x number of time steps (see Section 3.4.3 for Run Parameters settings and section 5.6.4 for an explanation of the difference between the kinematic and quasi-dynamic models).*

**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 3.10 for details of how this works.





An equation is then defined in the text box below using the x, y, z or t variables<sup>7</sup>. 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.

Adding the comment %time or %rows, allows the the row dimension to be added to the new dataset. For example if x and y data sets are timeseries, then a Matlab<sup>TM</sup> expresion, or function call, call can be used to create a new time series as follows:

```
x^2+y %time
```

The output from function calls can be figures or tables, a single numeric value, or a dataset to be saved (character vectors, arrays or dstables). External functions should return the table RowNames (e.g., time or location) as the first variable (or an empty first variable), followed by the other variables to be saved.

If there is no output to be passed back the function should return a string variable.

If varout = 'no output'; this suppresses the message box, which is used for single value outputs. For expressions that return a result that is the same length as one of the variables used in the call, there is the option to add the variable to the input dataset as a new variable. In all there are three ways in which results can be saved:

1. As a new dataset;
2. As an additional variable(s) to one of the input datasets;
3. As an additional variable(s) to some other existing dataset.

For options 2 and 3, the length of the new variables must be the same length (numbere of rows) as the existing 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 (plus time if defined for a selected variable) can be selected but they are defined in the call using dst, for example:

```
[time,varout] = myfunction(dst, 'usertext', mobj);
```

```
dst = myfunction(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 number of variables. When a dstable, or struct of dstables is returned, it is assumed that the dsproperties have been defined in the function called and dstables are saved without the need to define the meta-data manually.

Some further details on using this option and the '**Function**' library available are provided in Section 6.1.

## 3.7 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 3.10.

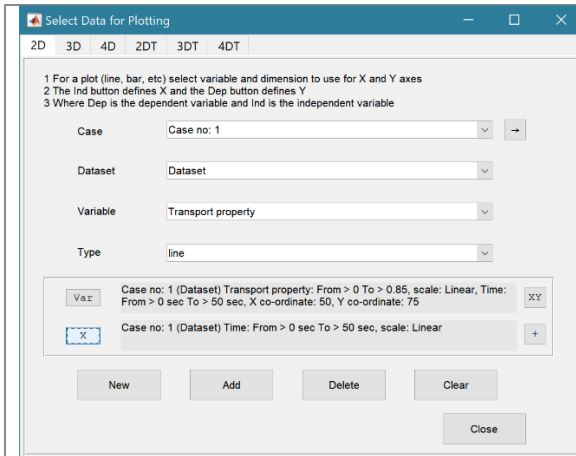
### 3.7.1 Plotting

*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

---

<sup>7</sup> 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.

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 and 4D plots. The following figures illustrate the options available.



## 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)

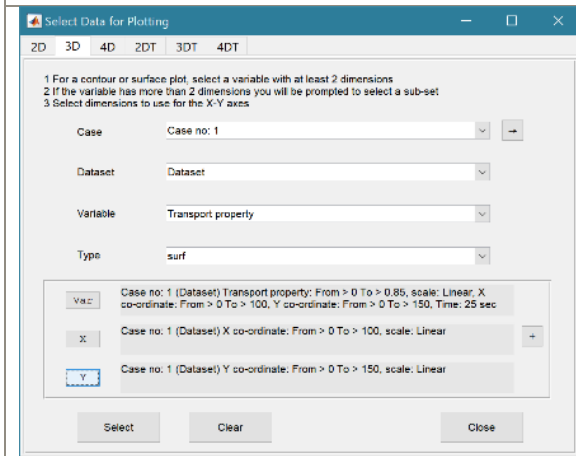
### Control Buttons:

→ : updates the list of Cases

XY : swaps the X and Y axes

+ : switches between cartesian and polar plot type

*If polar selected then Ind assumed to be in degrees.*



## 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.

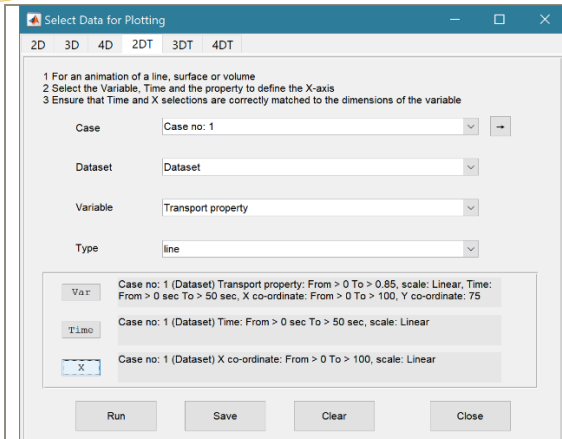
For all plot types, when the data has more dimensions than the plot or animation the user is prompted to sub-select from the data (by selecting sampling values for the dimensions that are not being used).

Animations follow a similar workflow. There are buttons at the bottom of each tab to:

**Run** the selection and create an animation,

**Save** the animation to a file (the animation needs to have been run first) . There is also an option to save on the bottom left of the animation figure.

**Clear** the current selection.



### 2DT animation

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

> Assign a variable, or a dimension, to the Var, Time and X buttons.

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

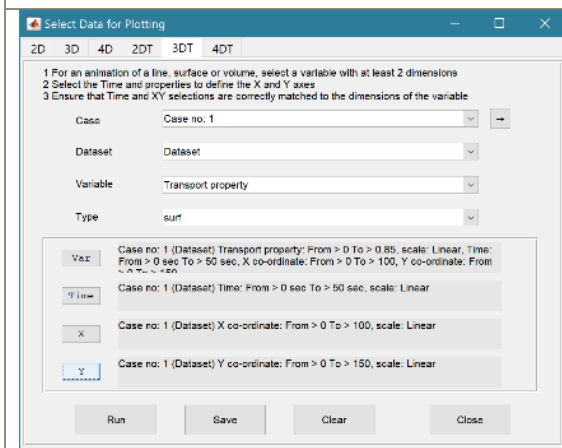
> Select plot type (line, bar, scatter, stem, etc)

#### Control Buttons:

→ : updates the list of Cases

+ : switches between cartesian and polar plot type

*If polar selected, then X assumed to be in degrees and when prompted select Polar and NOT Rose.*



### 3DT animation

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

> Assign selections to the Var, Time, X and Y buttons

Take care to ensure that the assignments to Time, 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 2DT plot above.

### 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 pfunctions 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.

### 3.7.2 Statistics

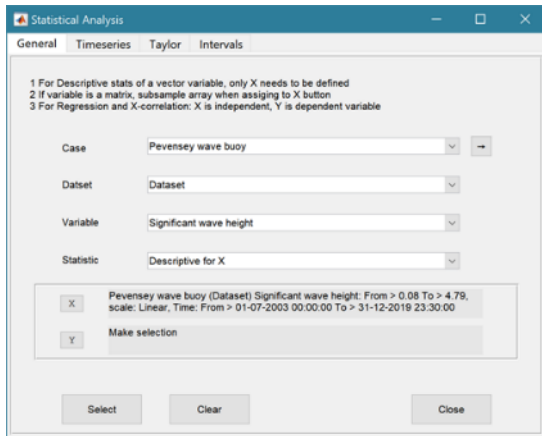
**Analysis > Statistics:** several statistical analysis options have been included within the Statistical Analysis GUI. The tabs are for **General** statistics, **Timeseries** statistics, model comparisons using a **Taylor** Plot, and the generation of a new record based on the statistics over the **Intervals** defined by another timeseries.

#### General tab

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

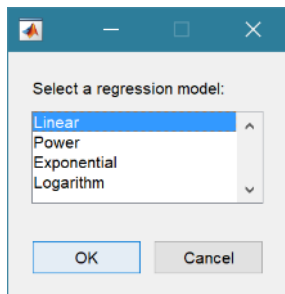


- 1) **Descriptive for X:** general statistics of a variable (mean, standard deviation, minimum, maximum, sum and linear regression fit parameters). Only X needs to be defined. The range of the variable can be adjusted when it is assigned to the X button (see Section 3.10). If the variable being used 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.



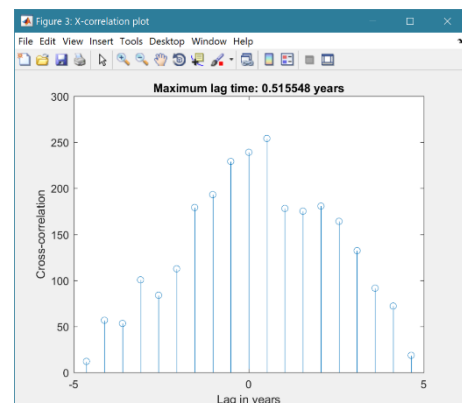
sum and linear regression fit parameters). Only X needs to be defined. The range of the variable can be adjusted when it is assigned to the X button (see Section 3.10). If the variable being used is a multi-dimensional matrix ( $>2D$ ), the user is prompted to define the range or each additional dimension, 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.

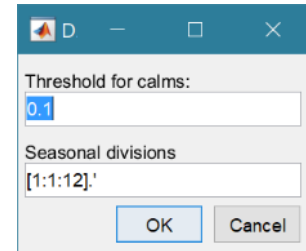
### Timeseries tab

The Timeseries tab allows the user to select a single Timeseries variable and apply any of the following statistics:

- 1) **Descriptive:** general statistics of a variable (mean, standard deviation, minimum, maximum, sum and linear regression fit parameters). The results are tabulated in a new window and can be copied to the clipboard for use in other applications.

Various ‘seasonal’ sub-divisions can be defined. The required option is selected from the table in the UI, by selecting a Syntax cell and then closing the UI.

The next UI prompts for a threshold for calms (values below threshold are deemed to be “calm” conditions) and allows the selected ‘seasonal’ divisions to be changed (if the desired option is not in the default list), or edited. The divisions can be expressed in several ways, as detailed below:



Script	Result
1	Descriptive statistics for the full-time series
[1:1:12].'	Descriptive statistics for the full-time series and monthly values (the .' creates a column vector).
[12,1,2; 3,4,5; 6,7,8; 9,10,11]	Descriptive statistics for the full-time series and seasons based on groupings – Dec-Feb, Mar-May, Jun-Aug, Sep-Nov shown.

When seasonal statistics are produced with more than 2 seasons a plot is generated. This can be a cartesian or polar plot of the mean values with error bars used to depict +/- one standard deviation. The polar plot maps the year as one revolution.

- 2) **Peaks:** generates a new timeseries of peaks over a defined threshold. There are three methods that can be selected:

- 1 = all peaks above the threshold;
- 2 = the peak value within each up-down crossing of threshold; and
- 3 = peaks that have a separation of at least ‘*tint*’ hours.

For option 3, the separation between peaks (‘*tint*’) is also be defined in the pop-up gui. This can be used to try and ensure that peaks are independent. The peaks are marked on a plot with the defined threshold. If rejected, new values can be defined. If accepted a new timeseries is added. This has the class of the Data Type that was used as the source timeseries but is not appended to that timeseries because the date/times are a subset of the source.

- 3) **Clusters:** The selection process is similar to peaks, where the user defines a threshold, selection method and time between peaks (for method 3). In addition, the cluster interval is defined in days. This is the period of time separating two peaks for them to be no longer considered part of a cluster (e.g. if a sequence of storms occurs every few days they will form a cluster. If there is then a gap of, say, 31 days to the next storm, with a cluster time interval of 30 days this would be considered as part of the next cluster). Once a selection has been made, a plot is generated that shows the peaks for each cluster with a different symbol. The user can either choose a different definition, or accept the definition. Once accepted, the results are added as a new timeseries, with the class of the Data Type that was used as the source timeseries. Two values are stored at the time of each peak in the clusters: the magnitude of the peak; and the number of the cluster to which it belongs (numbered sequentially from the start). This allows the data for individual clusters to be retrieved, if required.
- 4) **Extremes:** The selection process is similar to peaks, where the user defines a threshold, selection method and time between peaks (for method 3). A figure is generated with two plots. The left-hand plot shows the peaks for the defined threshold and the right hand plots shows the mean excess above the threshold (circles), the 95% confidence interval (dotted red lines) and the number of peaks (vertical bars + right hand axis) as a function of threshold. This plot can be used to help identify a suitable threshold for the peak-over-threshold extremes analysis method. The user can either choose a different definition, or accept the definition. Once

accepted, the user is prompted to select a plot type. Options are: None; Type 1 – a single return period plot; Type 2 – a composite plot showing the probability, quantile, return period and density plots. See Coles (2001) for further details of the method used and the background to these plots. The results are tabulated on the *Stats/Extremes* tab and can be copied to the clipboard for use in other applications.

- 5) **Poisson Stats:** user is prompted to select a threshold, method and peak separation (see Peaks above) and the function generates a plot of the peak magnitude, time between peaks (interarrival time) and the duration above the threshold for each peak. The plot shows a histogram of each variable and the exponential pdf derived from the data, along with the  $\mu$  value for the fit.
- 6) **User:** calls user\_stats.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 psfunctions folder and illustrates the workflow to produce a clusters plot. Some code in the header (commented out) shows how to get a time series using the handles passed to the function (obj and mobj). This code would get the same timeseries as the one passed to the function. However, by modifying the 'options' variable it is possible to access other timeseries variables.

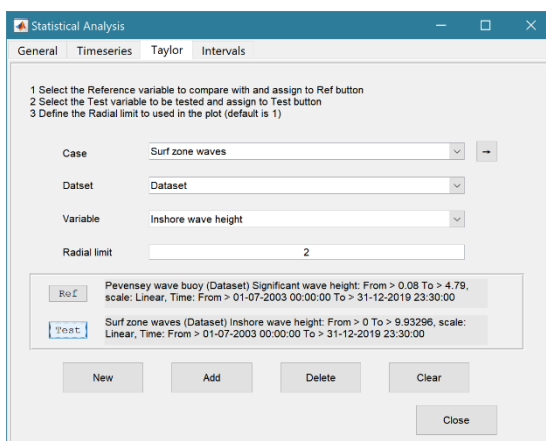
### Taylor tab

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).

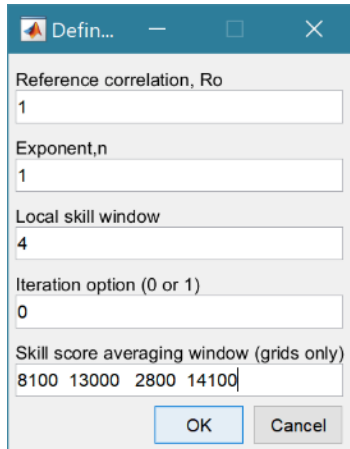
*[The ModelSkill App provides additional tools to test data and the ModelSkill App manual provides further details of the methods used.]*

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.



Defin...

Reference correlation,  $R_o$   
1

Exponent,  $n$   
1

Local skill window  
4

Iteration option (0 or 1)  
0

Skill score averaging window (grids only)  
8100 13000 2800 14100

OK Cancel

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

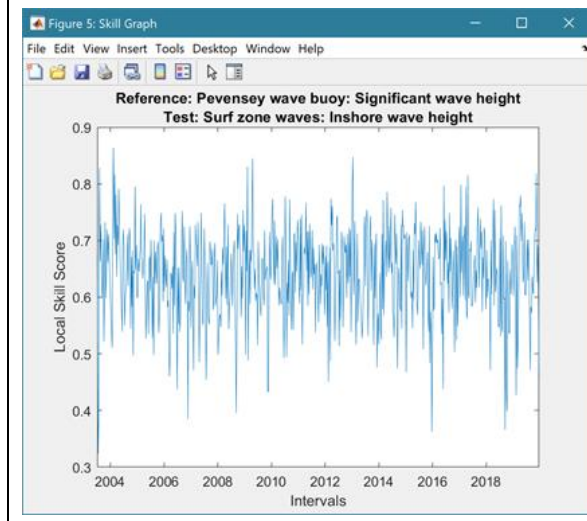
Exponent used in computing the skill score (see ModelSkill manual for details).

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.

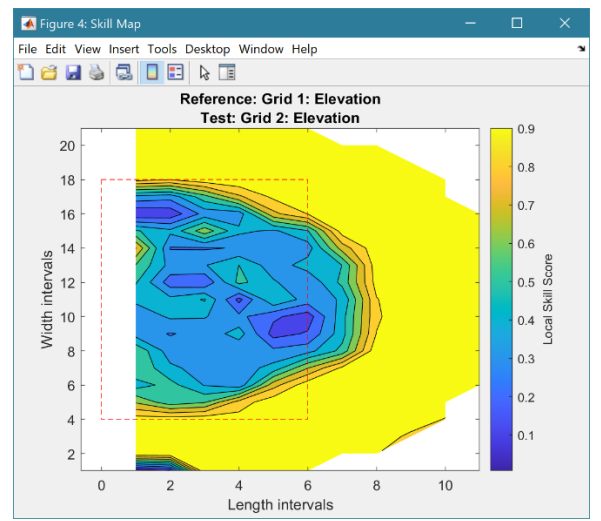
Local skill score is computed for window around every grid cell ( $=1$ ), or computes score for all non-overlapping windows ( $=0$ )

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

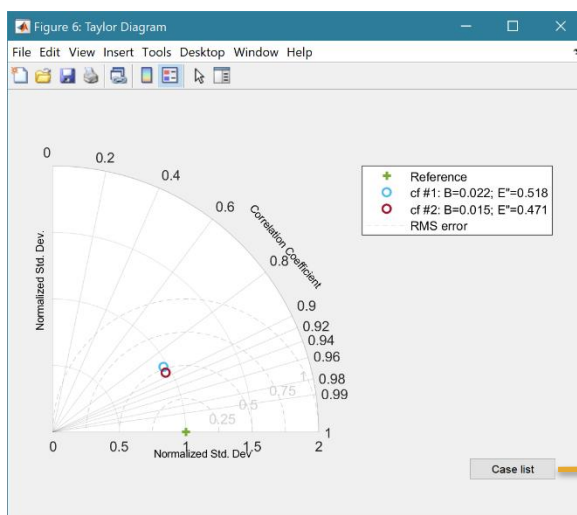
(a) time series skill score plot



(b) grid skill score plot

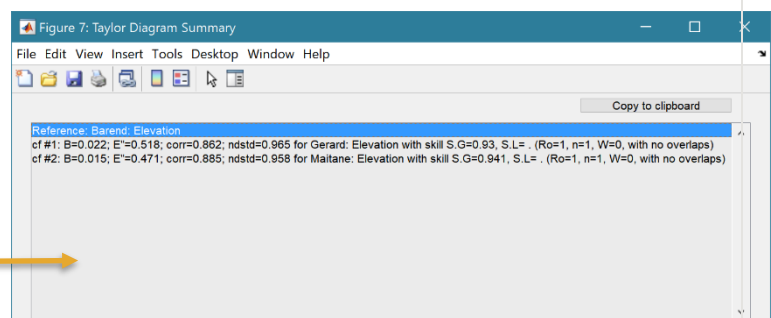


The Taylor Plot shows the Reference point as a green cross and the Test points as coloured circles. The legend details the summary statistics and the Case List button generate a table figure listing all the results. These can be copied to the clipboard.



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,  $S_g$ , and the average local skill score,  $S_l$ .

### 3.7.3 Transgression Plots

Several standard output plots are available for the Transgression model, together with an option to export the GrossProps and Transgression Tables to a text or spreadsheet file. Details of the output table content are given in Section 3.11.

*Analysis>Transgression Plots>Grid Plot* - plot initial grid, new grid and difference.

*Analysis>Transgression Plots>Change Plot* - plot volume difference over the model run and the variation of sediment exchange with transgression distance.

*Analysis>Transgression Plots>Section Plot* - plot a set of along-channel cross-sections and the channel centre-line (thalweg) for the initial and final grid.

*Analysis>Transgression Plots>Transgression Plot* - composite set of plots to show results from Transgression table.

*Analysis>Transgression Plots>Export Table* - write GrossProps and Transgression output tables from the Transgression model to a file.

## 3.8 Help

The help menu provides options to access the App documentation in the Matlab™ Supplemental Software documentation, or the App manual.

## 3.9 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).

**Form:** summary of the properties that have been defined

- **Exponential** - tabulates the parameters required for the exponential form model.
- **Power** - tabulates the parameters required for the power law form model.
- **Valley** - tabulates the parameters required for the valley form model.
- **Shore** - tabulates the parameters required for the shore profile form model.

**Settings:** tabulates the current model settings (display only).

- **Forcing** - tabulates the water level and hydrodynamic parameters.
- **Sediments** - tabulates the sediment parameters.
- **Transgression** - tabulates the additional parameters needed for the transgression model.
- **Modifications** - tabulates constant modifications to the channel form.
- **Run Parameters** - tabulates the key run parameters, conditions and equilibrium coefficients).

**Hydro-Props**

- **Water Levels** - plots the water levels as a function of time.
- **Hydraulics** - plots the along channel variation of water levels.

**Form-Props:** tabulates the gross properties together with a range of property plot.

**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).

### 3.10 UI Data Selection

Functions such as Derive Output (3.6), Plotting (3.7.1) and Statistics (3.7.2) 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 (e.g., 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:

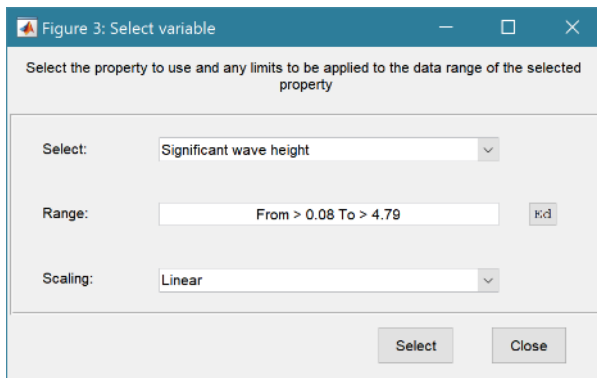


Figure 3: Select variable

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.

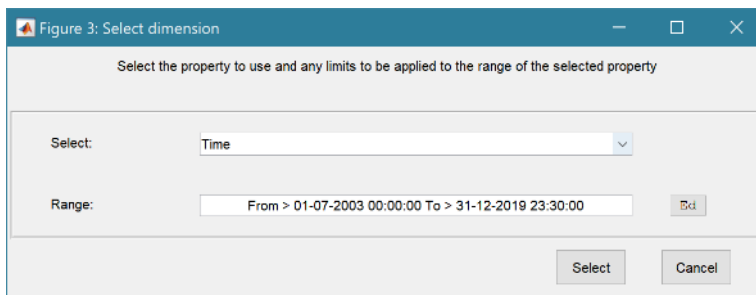


Figure 3: Select dimension

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]

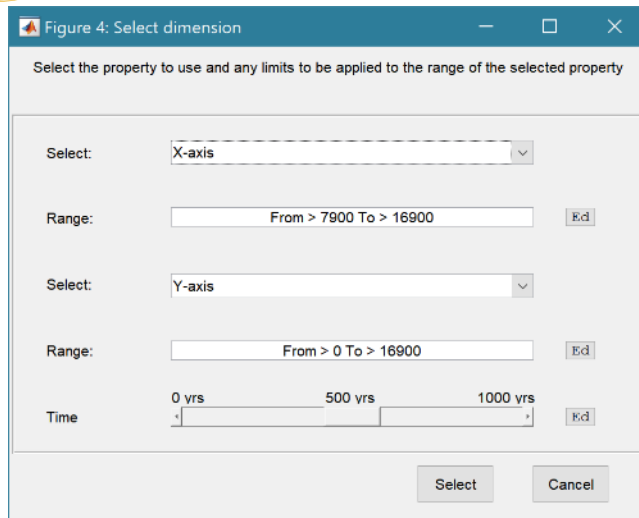
[Select] [Cancel]

#### Step 2 - Variable only has dimension of time.

No selection to be made.

Edit range if required.





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

Select the 1<sup>st</sup> variable to use as a dimension.  
Edit range if required.

Select the 2<sup>nd</sup> 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.

### 3.11 Form and Transgression Model Output Table Content

The following tables are generated when a Form model is created or the Transgression model is run.

#### 3.11.1 Form Table

Elevations of grid along with the x and y dimensions, with the following variables:

- Z - Elevation (mAD)

Dimensions of time (table rows), X and Y.

*UserData.ishead* - orientation of x-axis relative to mouth (true if minimum x is at the head of the estuary/inlet, default is false).

*UserData.xM* - distance from grid origin to mouth of estuary/inlet (default is 0).

*UserData.cline* – struct for x and y coordinates of meander centre-line.

*Description* – user assigned Case description.

*Source* - source file or model class.

*MetaData* - details any manipulation eg type of model or grid rotation.

#### 3.11.2 Hypsometry Table

Variation of surface area and volume as a function of elevation, with the following variables:

- Volume - Volume (m<sup>3</sup>)
- SurfaceArea - Surface area (m<sup>2</sup>)
- SAfreq - Surface area frequency

*Dimensions* of time (table rows), and Z.

*Description* – user assigned Case description.

*Source* - source file or model class.

*MetaData* - details any manipulation eg type of model or grid rotation.

### 3.11.3 SectionProps Table

Variation of the width and cross-sectional area along the x-axis derived from the gridded data, with the following variables:

- Whw - High Water Width (m)
- Wmt - Mean Tide Width (m)
- Wlw - Low Water Width (m)
- CSAhw - High Water Cross-sectional Area (m<sup>2</sup>)
- CSAmt - Mean Tide Cross-sectional Area (m<sup>2</sup>)
- CSAlw - Low Water Cross-sectional Area (m<sup>2</sup>)
- Dhwh - High Water depth (m)
- Dmt - Mean Tide depth (m)
- Dlw - Low Water depth (m)
- PrA - Tidal Prism (using CSA) (m<sup>3</sup>)
- Shw - High Water surface area (m<sup>2</sup>)
- Smt - Mean Tide surface area (m<sup>2</sup>)
- Slw - Low Water surface area (m<sup>2</sup>)
- Vhw - High Water volume (m<sup>3</sup>)
- Vmt - Mean Tide volume (m<sup>3</sup>)
- Vlw - Low Water volume (m<sup>3</sup>)
- PrV - Tidal Prism (using hypsometry) (m<sup>3</sup>)
- Gamma - Dronkers' Gamma (-)
- Vs - Storage Volume (m<sup>3</sup>)
- Vc - Channel Volume (m<sup>3</sup>)
- amp - Tidal amplitude at mouth (m)
- hyd - Hydraulic depth of channel (m)

*Dimensions* of time (table rows), and X.

*Description* – user assigned Case description.

*Source* - source file or model class.

*MetaData* - details any manipulation eg type of model or grid rotation.

### 3.11.4 Plan Table

Widths along the x-axis, specified in the model, or based on an exponential fit to the gridded data, with the following variables:

- Whw - High Water Width (m)
- Wmt - Mean Tide Width (m)
- Wlw - Low Water Width (m)

*Dimensions* of time (table rows), and X.

*Description* – user assigned Case description.



*Source* - source file or model class.

*MetaData* - details any manipulation eg type of model or grid rotation.

### 3.11.5 GrossProps Table

Various summary properties of the form related to length, area, volume, rate of convergence, etc, with the following variables:

- Shw - High Water Surface Area ( $m^2$ )
- Slw - Low Water Surface Area ( $m^2$ )
- Vhw - High Water Volume ( $m^3$ )
- Vlw - Low Water Volume ( $m^3$ )
- PrA - Tidal Prism using cross-sectional areas ( $m^3$ )
- PrV - Tidal Prism using hypsometry volumes ( $m^3$ )
- Gamma - Dronkers' Gamma (-)
- Vs - Storage Volume ( $m^3$ )
- Vc - Channel Volume ( $m^3$ )
- Wm - Mean Tide Width at mouth (m)
- Am - Mean Tide Cross-sectional Area at mouth ( $m^2$ )
- Dm - Depth at mouth to MTL (m)
- amp - Tidal amplitude at mouth (m)
- hyd - Hydraulic depth of channel (m)
- aoh - Amplitude to Depth ratio (-)
- VsoVc - Storage to Channel Volume ratio (-)
- PrvAm - Prism to Cross-sectional Area ratio at mouth (m)
- SflShw - Intertidal to Basin Area ratio
- Lw - Width convergence length
- La - Cross-sectional Area convergence length

*Dimensions* of time (table rows).

*Description* – user assigned Case description.

*Source* - source file or model class.

*MetaData* - details any manipulation eg type of model or grid rotation.

### 3.11.6 WaterLevels Table

Variation in water levels at high water, mean tide and low water along the x-axis, with the following variables:

- zhw - High Water level (mAD), where mAD = metres above Datum
- zmt - Mean Tide level (mAD)
- zlw - Low Water level (mAD)

*Dimensions* of time (table rows), and X.

*Description* – user assigned Case description.

*Source* - source file or model class.

*MetaData* - details any manipulation eg type of model or grid rotation.

### 3.11.7 Transgression Table

When the Transgression model is run as additional table is included in the model output tables containing a set of output properties that summarise the changes of form over time, with the following variables:

- delX - Unadjusted transgression distance (m)
- estdX - Adjusted transgression distance (m)
- cstdX - Open coast transgression distance (m)
- dSLR - Cumulative sea level rise (m)
- Lt - Distance to tidal limit (m)
- FPA - Flood plain area (m)
- waterVol - Water volume change due to changes in HW (m<sup>3</sup>)
- sedVol - Cumulative sediment flux (+ve = sediment import) (m<sup>3</sup>)
- diffv - Volume change for [0, estdX, delX] (m<sup>3</sup>)

*Dimensions* of time (table rows).

*Description* – user assigned Case description.

*Source* - source file or model class.

*MetaData* - details any manipulation eg type of model or grid rotation.

### 3.12 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:	<b>Purpose</b>
Inputs: [1×1 struct]	A struct with field names that match all the model parameter input fields currently
Cases: [1×1 muiCatalogue]	muiCatalogue 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.

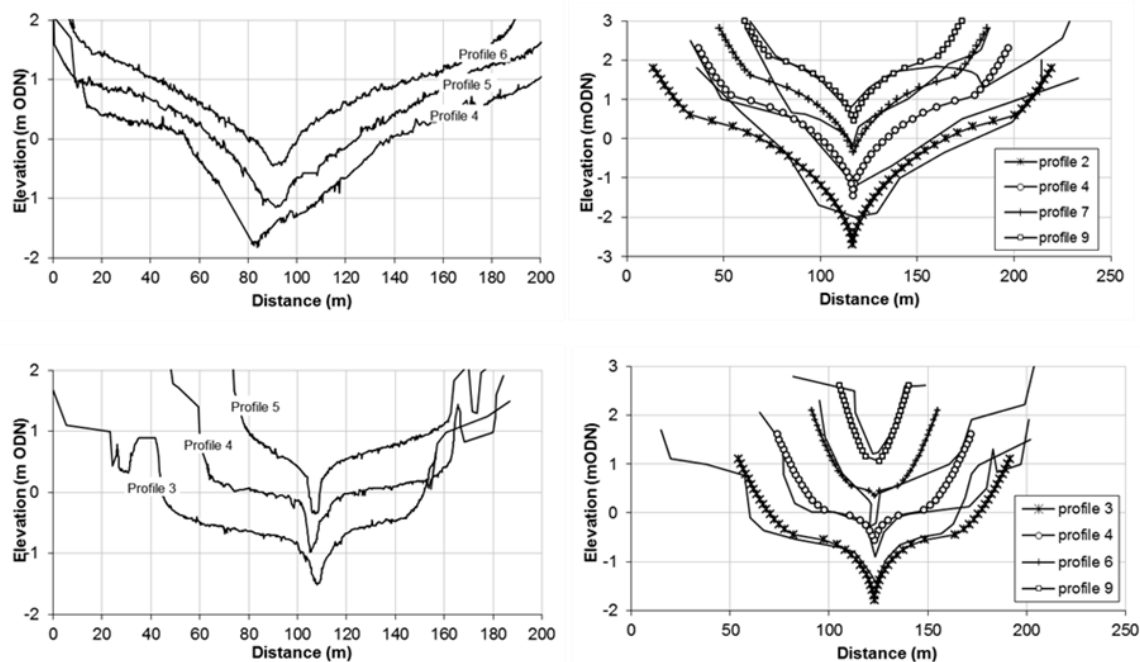
## 4 Background to Form Models

Many channels and tidal creeks are observed to have a relatively simple compound form. See for example the sections on the left-hand side of Figure 1, where two creeks in the Solent on the south coast of the UK are illustrated. Both creeks are above low water at their lowest point. Although subtly different, both creeks comprise an upper concave-up slope and a lower convex-up slope. Estuary channels are invariably more complex because of the presence of ebb and flood channels and shoals, the presence of geological controls and sea walls. Nonetheless they can be represented in cross-section by a simple parametric relationship for the channel and intertidal. Large channels, particularly those with a significant river influence, would appear to have additional points of inflection, most likely due to the influence of waves on the intertidal area and the combined influence of tidal and fluvial flows in widening the base of the channel.

A 3D form can be obtained by specifying the variation in any two planes (i.e. x-y and x-z, or y-z and y-x, etc). One can therefore define the variation in width and depth as functions of distance along the estuary, or alternatively specify the variation of the cross-section together with the along-estuary variation in either width or depth. Thus by specifying an upstream exponential decay in width and, say a parabolic cross-sectional form (Cao & Knight, 1997), a simple 3D form can be generated. To more closely represent a tidal channel, the cross-sectional description can be extended to include the intertidal by, for example, combining the parabolic form of the channel with a trigonometric relationship for the intertidal (the latter having been suggested by Friedrichs & Aubrey, 1996)

By specifying several key estuary dimensions, a 3D form can be generated and a range of estuary “geometric properties” can be derived from the form. This is the basis of the so called “Form” model. An alternative is to define the controls/constraints on the system and the forcing (or exogenous) conditions, as explored in Townend, 2010. This is the basis of the so called “CKFA” model. The following sections explain the formulation of these two models in greater detail.

Figure 1- Selected cross-sections for (a) Forton Lake and (b) Causeway Lake. Left plots show original data with profiles offset vertically in 0.5 m intervals for clarity. Right plots show smoothed measured profiles compared with model sections denoted by the symbols



## 4.1 Basis of the Exponential Form model

The Exponential Form model is a geometric construction comprising a channel cross-section an intertidal cross-section and a plan form that defines the along channel variation in width at high and low water. Several options are provided for each of these components.

### 4.1.1 Longitudinal plan form

#### *Exponential convergence*

An idealised estuary has been shown to have a width convergence that is exponential (Pillsbury, 1956). The variation in width is therefore prescribed as exponential along the length of the estuary.

$$W_m(x) = W_{m0} \exp\left(\frac{-x}{L_W}\right) \quad (4.1.1)$$

where  $W_{m0}$  is the width at mean tide level and at the mouth,  $L_W$  is the rate of width convergence and the subscript m refers to some reference elevation (e.g. mtl). For many systems this describes the outer part of the system well but the tidal-river has a more prismatic section. For this reason, a slightly modified form of exponential convergence is adopted:

$$W_m(x) = W_{mr} + (W_{m0} - W_{mr}) \exp\left(\frac{-x}{L_W}\right) \quad (4.1.2)$$

where  $W_{mr}$  is the cross-sectional width of the river reach.

#### *Power law*

The power law option defines the variation in plan form (ie W) as a power law defined by:

$$W_m(x) = (W_{mL} - W_r) \cdot \left(\frac{x}{L}\right)^n + W_r \quad (4.1.3)$$

where  $x$  is distance along the channel,  $W_m$  is the width along the channel,  $W_{mL}$  is the width at the mouth, the length of the channel is  $L$  and the suffix m defines an elevation. The plan form is determined by defining the variations in width at high and low water. To allow for a river channel at the landward end of the model the offset width,  $W_r$ , is included and applied uniformly along the length of the channel.

### 4.1.2 Low water channel

#### *Parabolic low water channel*

A close correspondence between the equilibrium form relationships for both estuaries and rivers suggests that “estuaries do not substantially differ from alluvial rivers in terms of morphology” (Savenije, 2012). The low water channel cross-section is therefore based on an approach originally derived for river channels (Cao & Knight, 1997). This is illustrated in *Figure 2*, which also defines some of the key dimensions.

Using entropy concepts Cao & Knight (1997) derive a parabolic equation for channel form:

$$d(y) = \frac{\mu W_{lw}}{2n_{bk}} \left[ 1 - \left( \frac{n_{bk} y}{W_{lw}} \right)^n \right] \quad (4.1.4)$$

Where  $d$  is the depth at a distance  $y$  from the centreline of a channel of width,  $W_{lw}$ . The factor  $n_{bk}$  has a value of 2 for a channel that is symmetric about the centreline and 1 if the width relates to a half channel. The parameter  $\mu$  is the submerged static friction factor taken to be equal to  $\tan \phi$ , where  $\phi$  is the angle of repose of the sediment. The value of  $\mu$  largely determines the aspect ratio. The parabolic form given by Cao & Knight is defined with an exponent of  $n=2$ . More generally, treating the exponent as a variable determines the shape of the channel, the larger the value the closer the form is to a rectangular channel (i.e. equivalent to the Rectangular low water channel as defined below).

### Rectangular low water channel

This channel section has a constant depth across the width of the low water channel. For the depth at the mouth to be comparable with the parabolic channel case, the hydraulic depth,  $h_d$ , of the parabolic channel is used to define the depth at the mouth. This is given by (Cao and Knight, 1997):

$$h_d = \mu W_{lw}/2/(n+1), \text{ or } h_d = n.h_c/(n+1), \text{ with } \mu = 4.n.h_c/W_{lw}$$

where the parameters are as defined for Equation 4.1.4.

### 4.1.3 Intertidal flat profile

Four different forms can be defined for the intertidal: linear plane slope, a rectangular section, section based on the flow only form that is consistent with uniform shear stress and a similar form proposed by Lee and Mehta (1997) for muddy shores.

#### Linear

The profile has a plane slope between the high and low water widths defined by the plan form.

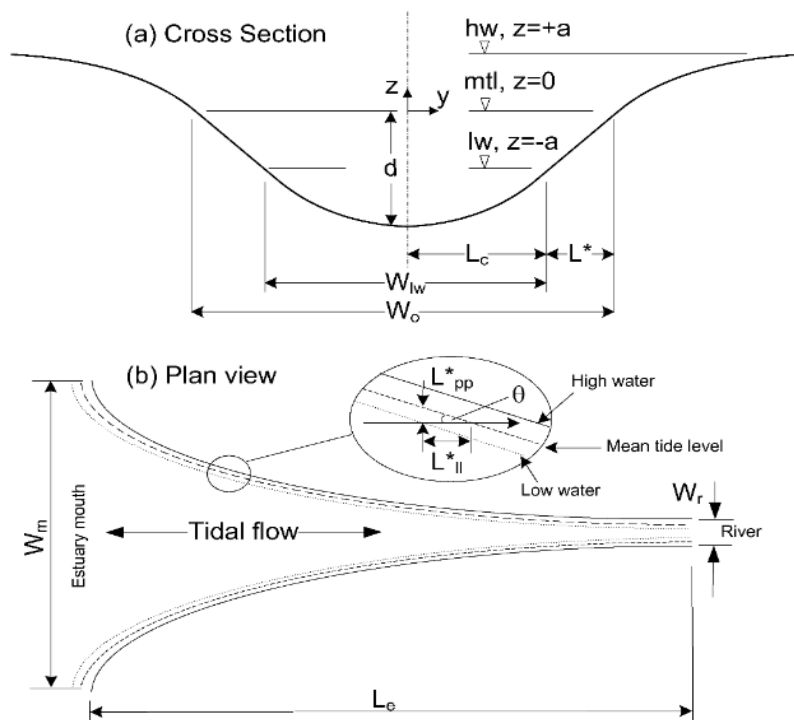
#### Rectangular

The profile has a constant elevation at the mean tide level.

#### Uniform shear

The equilibrium form of the intertidal is based on the uniform dissipation of energy to describe the intertidal profile, based on the maximum depth averaged velocity as a proxy for the maximum bottom shear stress. Friedrich & Aubrey (1996) use a kinematic approach to determine the magnitude of the tidal velocity over the tidal flat. They show that a uniform distribution of tidal velocity is obtained over a linear slope below mean tide level but the profile needs to be non-linear above mean tide level and is suitably represented by a trigonometric function.

Figure 2 - Definition sketch for equilibrium form of channel and intertidal (a) cross-section; (b) plan view



With reference to the definitions in *Figure 2*, the intertidal profile on the basis of uniform distribution of tidal velocity is described by (Friedrichs & Aubrey, 1996, eq.(11) and (15)):

$$\begin{aligned} z &= a \cdot \left( \frac{y'}{L^*} - 1 \right) \quad \text{for } y' \leq L^* \\ z &= a \cdot \sin \left( \frac{y'}{L^*} - 1 \right) \quad \text{for } y' > L^* \end{aligned} \quad (4.1.5)$$

where  $y' = y - L_c$ . If we assume a symmetric channel it follows that  $W' = W - W_{lw}$  and  $L^* = (W_o - W_{lw})/2$ . Whence:

$$\begin{aligned} z &= a \cdot \left( \frac{2y - W_{lw}}{W_o - W_{lw}} - 1 \right) \quad \text{for } \frac{W_{lw}}{2} > y \leq \frac{W_o}{2} \\ z &= a \cdot \sin \left( \frac{2y - W_{lw}}{W_o - W_{lw}} - 1 \right) \quad \text{for } y > \frac{W_o}{2} \end{aligned} \quad (4.1.6)$$

#### *L&M muddy shore*

Lee and Mehta (1994; 1997) description of a muddy intertidal profile is derived by considering the wave energy dissipation across the shore. They introduce a wave attenuation coefficient averaged over the profile length,  $k_i$ , which can be related to the properties of the bed. Equation 9 from their paper defines the depth in terms of the intertidal width,  $y_o$ , and the depth at the seaward end of the profile,  $h_o$ , as follows:

$$h = h_o \exp(4k_i y) \left[ 1 - \frac{y}{y_o} \right]^2 \quad (4.1.7)$$

One problem with this formulation is that it cannot handle very large intertidal widths as the exponential term grows too quickly unless the value of  $k_i$  is adjusted spatially along the length of the estuary. For simplicity, I have therefore made the exponential factor non-dimensional by dividing it by the intertidal width and adjusting the value of  $k_i$  to suit. If we now re-pose the equation in this non-dimensional form, and in terms of width and elevation, as used for the other models, the intertidal width is  $L_i = (W_{hw} - W_{lw})/2$ ; the  $y$  co-ordinate for the intertidal is transformed as  $y_i = y - W_{lw}/2$  for  $y > W_{lw}/2$  and the depth of the profile at the seaward end of the intertidal is twice the tidal amplitude,  $2a$ . The elevation  $z$  about the mean tide level is then given by:

$$z = a \left\{ 1 - 2 \cdot \exp \left( 4 \cdot \hat{k}_i \frac{y_i}{L_i} \right) \cdot \left[ 1 - \frac{y_i}{L_i} \right]^2 \right\} \quad (4.1.8)$$

Where  $\hat{k}_i$  is a modified version of  $k_i$  that is about  $10^2$  larger and dimensionless. For this application it is simply treated as a shape coefficient in much the same way as  $n$  and  $m$  are in the first model and values used are in the range 0.01 to 0.5. Low values result in a convex-up profile, similar to the F&A intertidal profile, whereas higher values produce an intertidal profile that is concave-up. The former is compatible with a tidally dominated profile and the latter with a wave dominated profile.

## 4.2 Basis of the Power Law model

This model uses power law descriptions for both the plan and cross-sectional variations in form. This has the advantage that it can represent a wide range of different possible forms and is able to reproduce funnel and bell-shaped plan forms, in conjunction with two-stage concave or convex cross-shore forms. By using power laws, it is possible to readily derive relationships for various gross properties, such as tidal prism, plan area and volume and to also relate the form to the hydraulics using the analytical solutions suggested by Evangelisti (1955), 1955 and Prandle & Rahman (1980).



In its simplest form, this relationship can be considered as a rectangular section with varying width and depth, where the depth is equivalent to the hydraulic depth. This estuary form can then be used to derive an analytical solution for tidal wave propagation in estuaries (Evangelisti, 1955; Prandle & Rahman, 1980). The width at mean tide level and the hydraulic depth at any given section is described as follows:

$$W = W_L \cdot \left( \frac{x}{L} \right)^n \quad \text{and} \quad H = H_L \cdot \left( \frac{x}{L} \right)^m \quad (4.2.1)$$

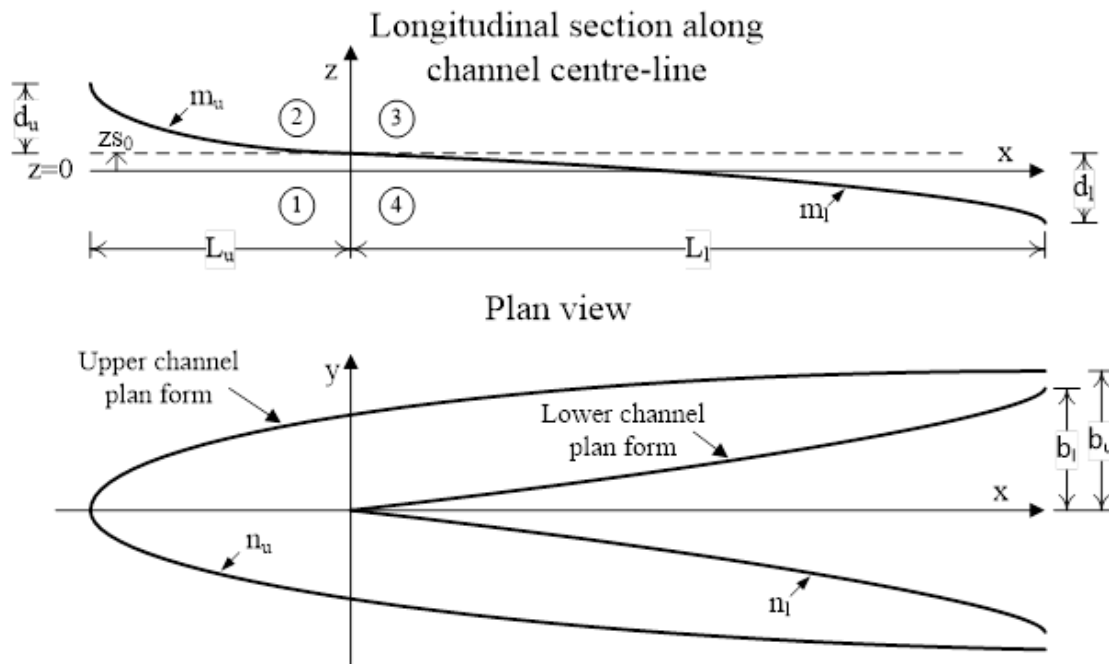
Where  $W$  is the width,  $H$  is the hydraulic depth and  $L$  is the length over which the variation occurs.  $W_L$  and  $H_L$  are the width and depth at  $x=L$ . The values of the shape parameters,  $n$  and  $m$ , determine whether the form is a funnel ( $>1$  = convex to centre-line) or bell shaped ( $<1$  = concave to centre-line). There is, however, no requirement for the cross-section to be rectangular; the solution simply requires that the hydraulic depth varies as specified in equation (4.2.1).

A slightly more complex form, capable of reproducing the key characteristics seen in creeks and estuaries, can be obtained by mirroring the form defined by equation (1) about some horizontal datum. The model is based on a longitudinal thalweg (total depth and not hydraulic depth) defined about a zero datum in  $x$  and  $z$  (which will later be referred to as the ‘shoulder’). As  $x$  becomes negative so the profile inverts, to give the longitudinal section shown in *Figure 3*. To define the plan form, the origin at any given level,  $z$ , is moved to be coincident with the position ( $x$ -chainage) of the thalweg at that level. The plan form is only calculated for positive  $x$  values, relative to the transformed origin. Depths can then be calculated from the following relationship:

$$z(x, y) = d_L \cdot \left( \frac{x}{L} - \left( \frac{y}{b_L} \right)^{1/n} \right)^m \quad (4.2.2)$$

where  $d_L$  and  $b_L$  are the depth and half-width at length  $L$  respectively.

Figure 3 - Definition sketch of channel shape





The inversion about the shoulder means that this produces a complex form but is rather inflexible because the same values are being used to define the form above and below the zero datum in the vertical. A more flexible form can be obtained by allowing the depth, width and shape parameters  $m$  and  $n$  to vary above and below this plane. The relevant parameters are defined in *Figure 3*. Also shown denoted on the longitudinal section are the four domains of positive/negative  $x$  and  $z$ . Domain 1 does not need to be considered because there is no form in this domain. Domain 3 is a horizontal extension of domain 2. The solutions for domains 2 and 4 can then be obtained by simple manipulation of equations (1) and the origin of solution for each elevation to give:

Domains 2 and 3:

$$z(x, y) = d_u \cdot \left[ 1 - \left( \frac{L_t}{L_u} \left( \frac{x}{L_t} - \left( \frac{y}{b_u} \right)^{1/n_u} \right) \right) \right]^{m_u} + z_{s_0} \quad \text{where} \quad L_t = L_u + L_l \quad (4.2.3)$$

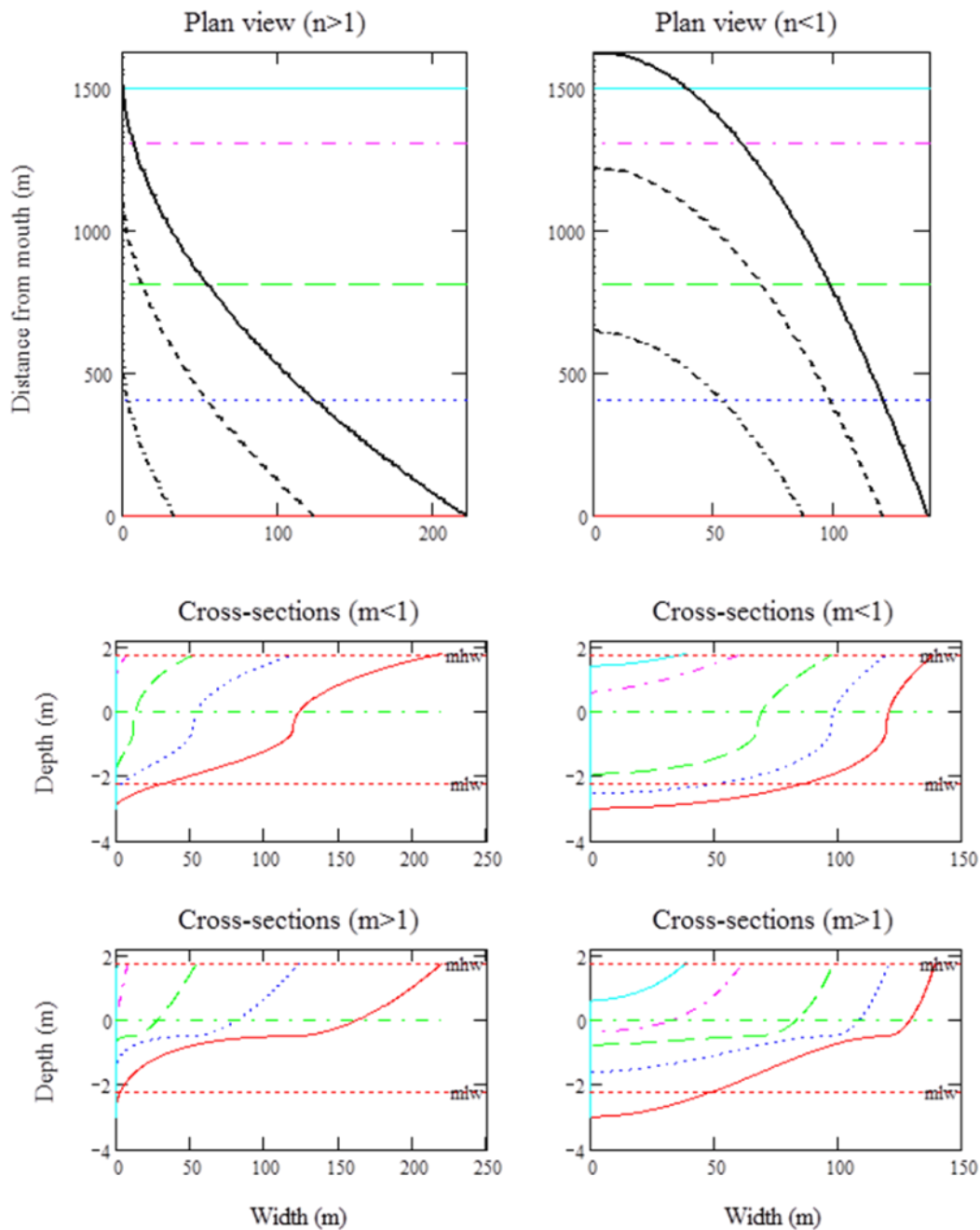
Domain 4:

$$z(x, y) = d_l \cdot \left( \frac{x_0}{L_l} - \left( \frac{y}{b_l} \right)^{1/n_l} \right)^{m_l} + z_{s_0} \quad \text{where} \quad x_0 = x - L_u \quad (4.2.4)$$

Simply varying the values of  $m$  and  $n$  alters the shape of both the plan and cross-sectional form as illustrated in *Figure 4 - Plan and cross-sections for forms generated using different combination of  $n$  and  $m$* . By altering the values that scale the form (the depth, width and length values) and control the shape ( $m$  and  $n$ ) quite subtle variations in form can be achieved, although care must be taken to ensure there is no discontinuity at the zero datum.

This was tested on two small tidal creeks that almost dry at low water and found to provide a good description of the form as illustrated in *Figure 1*. These results suggest that the formulation works well for creeks and other forms with a single inflection. The model has also been tested on larger estuaries, such as the Humber. Whilst a fit can be achieved, this requires considerable additional adjustment, presumably due to other influences, such as the river flow, waves on the intertidal and complex shoals and ebb-flood channels.

Figure 4 - Plan and cross-sections for forms generated using different combination of  $n$  and  $m$



#### 4.3 Basis of the 'CKFA' model

The CKFA model is so called because it derives the channel cross section from the river channel form suggested by Cao and Knight (1997) and an intertidal form based on the tidal flow only form suggested by Friedrichs and Aubrey (1996). Hence CKFA. The plan form used is exponential.

The form model is constructed from the following components:

- A plan form described by exponentially converging width and cross-sectional area;
- A low water channel that is parabolic in shape, based on the most probable distribution of transverse slope; and
- An intertidal flat profile that has uniform bottom shear stress across the profile as the tide varies.

The model is detailed in Townend (2010; 2012) and the components are explained in more detail below.

#### 4.3.1 Longitudinal Plan form

The along estuary variation in cross-sectional area,  $A$ , at mean tide level (mtl) is assumed to be exponentially convergent (Savenije, 2012):

$$A(x) = A_m \cdot \exp\left(\frac{-x}{L_A}\right) \quad (4.3.1)$$

Where  $A_m$  is the area at the mouth of the estuary,  $x$ , is the distance along the estuary from the mouth and  $L_A$  is the e-folding length<sup>8</sup> for cross-sectional area, as distinct from the length of the estuary,  $L_e$ . Similarly, the width and associated average depth at mean tide level,  $h_s$ , can be expressed as:

$$W(x) = W_m \cdot \exp\left(\frac{-x}{L_W}\right) \quad \text{and} \quad h_s(x) = h_{sm} \cdot \exp\left(\frac{-x}{L_h}\right) \quad (4.3.2)$$

Where  $W_m$  and  $h_{sm}$  are the width and depth at the mouth, and  $L_W = L_A$  if  $h_s$  is constant along the length of the estuary to the tidal limit. Otherwise, since  $A = Wh_s$  the e-folding lengths are related as follows:

$$L_A^{-1} = L_W^{-1} + L_h^{-1} \quad (4.3.3)$$

#### 4.3.2 Low water channel

A close correspondence between the equilibrium form relationships for both estuaries and rivers suggests that “estuaries do not substantially differ from alluvial rivers in terms of morphology” (Savenije, 2012). Given that the focus in this paper is on gross properties, without regard for detailed features such as flood and ebb channels, a simple description should suffice. The form of the low water channel cross-section is therefore based on an approach originally derived for river channels Cao & Knight (1997). This is illustrated in *Figure 2(a)*, which also defines some of the key dimensions.

Using entropy concepts Cao & Knight (1997) derive a parabolic equation for channel form:

$$d(y) = \frac{\mu W_{lw}}{2n_{bk}} \left[ 1 - \left( \frac{n_{bk} y}{W_{lw}} \right)^2 \right] \quad (4.3.4)$$

Where  $d$  is the depth at a distance  $y$  from the centreline of a channel of width,  $W_{lw}$ . The factor  $n_{bk}$  (number of banks) has a value of 2 for a channel that is symmetric about the centreline and 1 if the width relates to a half channel. The parameter  $\mu$  is the submerged static friction factor taken to be equal to  $\tan\phi$ , where  $\phi$  is the angle of repose of the sediment. The value of  $\mu$  largely determines the aspect ratio. More generally, one might consider the exponent of 2 to be a variable,  $n$ . This determines the shape of the channel, the larger the value the closer the form is to a rectangular channel.

#### 4.3.3 Tidal flat profile

The equilibrium form of the intertidal profile has been examined using arguments based on energetics in relation to wave activity (Dean, 1991; Lee and Mehta, 1994) and tidal propagation, or the combination of the two (Friedrichs and Aubrey, 1996). Alternative approaches include the use of detailed process models (Roberts et al., 2000; Woolnough et al., 1995), or consideration of the basin

<sup>8</sup> The e-folding length is a measure of the rate of convergence, being the length over which the variable (in this case area) changes by a multiple equal to the exponential function, ‘e’ (or its inverse when negative).

hypsonymetry, which has been used to examine the overall form of creeks and estuaries (Boon III and Byrne, 1981; Townend, 2008; Wang et al., 2002).

Here use is again made of entropy arguments and the uniform dissipation of energy to describe the intertidal profile, based on the maximum depth averaged velocity as a proxy for the maximum bottom shear stress. Friedrichs & Aubrey (1996) use a kinematic approach to determine the magnitude of the tidal velocity over the tidal flat. They show that a uniform distribution of tidal velocity is obtained over a linear slope below mean tide level but the profile needs to be non-linear above mean tide level and is suitably represented by a sine relationship. Although this assumes that the tidal flow is one dimensional, it does not require the currents to be perpendicular to the contours. Only that the currents flow at a constant angle to the bathymetry (Friedrichs & Aubrey, 1996). Within an idealised estuary, the flow is along the axis of the estuary and, because the banks of the system converge, the flow travels across the tidal flat at some small angle to the shore,  $\theta$ , *Figure 2(b)*. Using this simplified description to derive the intertidal dimensions in the direction of the flow, the component normal to the shore contours, or normal to the channel centreline, can then be readily obtained using the relevant trigonometric relationship.

For a channel that converges exponentially the shore is, strictly speaking, lobate in character. The intertidal profile can still be determined for this case as set out by Friedrich & Aubrey. However, for simplicity we assume that the estuary is sufficiently long that the curvature is small and that the shore can be treated as locally straight. This assumption is likely to be weakest near the mouth of estuaries with rapid convergence, or locally in systems with large meanders.

The width of the lower intertidal (from low water to mean tide level) in the along channel direction is given by  $L_{*ll}^* = u_T/\omega$ , where  $\omega$  is the tidal frequency and  $u_T$  is the tidal velocity. Resolving this normal to the channel centre-line and hence to the width on one bank of an estuary cross-section, one obtains  $L_{*pp}^* = L_{*ll}^* \tan(\theta)$  where  $\theta$  is the angle with the shoreline. The variation in width along the channel is defined by equation (7.2) and the angle of the shore to the centre-line is given by:

$$\theta(x) = \frac{d(-W(x)/2)}{dx} \quad \text{and} \quad \theta_{av} = \frac{W_m}{2L_e} \left( 1 - \exp\left(\frac{-L_e}{L_w}\right) \right) \quad (4.3.5)$$

Where  $\theta_{av}$  is the average value for an estuary of length,  $L_e$ .

With reference to the definitions in Figure 1, and writing  $L_{*pp}^* = L^*$ , a description of the intertidal form has been proposed by Friedrichs & Aubrey (1996, eq(11) and (15)) on the basis of uniform distribution of tidal velocity (in the flow direction):

$$\begin{aligned} z &= a \cdot \left( \frac{y'}{L^*} - 1 \right) \quad \text{for} \quad y' \leq L^* \\ z &= a \cdot \sin\left(\frac{y'}{L^*} - 1\right) \quad \text{for} \quad y' > L^* \end{aligned} \quad (4.3.6)$$

where  $y' = y - L_c$ . If we assume a symmetric channel it follows that  $W' = W - W_{lw}$  and

$L^* = (W_o - W_{lw})/2$ , or more generally  $L^* = (W_o - W_{lw})/n_{bk}$ , where  $n_{bk}$  is the number of banks and  $W_o$  is the width at  $z=0$ . A description of the estuary cross-section is then obtained by combining the intertidal and low water channel descriptions over three elevation intervals, as illustrated in *Figure 2*:

$$\begin{aligned}
 z &= a \cdot \sin \left( \frac{n_{bk} y - W_{lw}}{W_o - W_{lw}} - 1 \right) & \text{for } a \geq z > 0 \\
 z &= a \cdot \left( \frac{n_{bk} y - W_{lw}}{W_o - W_{lw}} - 1 \right) & \text{for } 0 \geq z > -a \\
 z &= -a - \frac{\mu W_{lw}}{2n_{bk}} \left[ 1 - \left( \frac{n_{bk} y}{W_{lw}} \right)^2 \right] & \text{for } z \leq -a
 \end{aligned} \tag{4.3.7}$$

#### 4.3.4 Idealised 3D form

The 3D form is obtained by combining equations (4.3.2) and (4.3.7) and results in a funnel shaped channel with a narrowing intertidal along the length of the estuary. In order to obtain the dimensions of this 3D form, for a given system, it is necessary to know the amplitude, frequency, and velocity of the tide, as well as the rates of cross-sectional area and width convergence and the overall length of the estuary or inlet. These can be obtained from the hydraulic and sediment balance relationships. In addition, the width of the system needs to be dimensioned and this is obtained by noting that the channel converges to meet the river and the dimensions of the river provide an independent constraint on the system.

#### 4.3.5 Dimensions of a tidal system

Equilibrium is defined by the state in which there is a balance of erosion and deposition over a suitable averaging period. The state is conditioned by the hydraulic forcing. This sort of balance has been extensively explored as the basis for defining the equilibrium cross-section (as noted above), in box models such as ASMITA and in “transport concentration” models (Di Silvio, 1999). What is novel in the model presented here is that the sediment balance equation is related to a 3D form that is itself a function of the hydraulics. In summary, there are three inter-related equations:

1. A hydraulic equation that gives the velocity in terms of the depth and the rate of estuary convergence (a key form parameter);
2. A sediment balance equation that defines a relationship between velocity and depth for given sedimentological conditions; and
3. A relationship between the prism as given by the hydraulics and the “equilibrium” 3D form, which links the rate of convergence with the depth.

Substituting the third equation into the second leaves two equations in terms of velocity and depth, which can be solved iteratively for the given boundary conditions.

The formulation is for convergent channels at low water (and hence is potentially applicable to inlets and estuaries) as developed in work by (Friedrichs and Aubrey, 1994; Savenije, 2012; Dronkers, 2016). In such formulations a distinction is made between the hydraulic depth at mean tide level and the hydraulic depth of the stream or channel width (Dronkers refers to these as the conveyance and propagation depths, respectively).

$$h_m = \frac{A_m}{W_m}; \quad h_s = \frac{A_m}{W_s}; \quad \text{assuming } W_s = \beta W_m; \quad h_s = h_m / \beta, \text{ where } \beta \approx 0.7 \quad (\text{Dronkers, 2017})$$

where subscript m refers to mean tide level values and  $W_s$  is the channel or stream flow width.

##### 4.3.5.1 Channel hydraulics

The one-dimensional hydraulic equations to describe tidal propagation in a channel have been extensively explored (Evangelisti, 1955; Friedrichs and Aubrey, 1994; Hunt, 1964; Jay, 1991; Lamb, 1932; Pillsbury, 1956; Prandle and Rahman, 1980). Useful summaries are provided of the various

conditions that can be applied and the different solutions that result in the texts by Dronkers (2016) and Savenje (2012). To obtain the tidal velocity amplitude the solution of the governing equations for strongly convergent estuaries is adopted, where frictional damping is balanced by the convergence of tidal energy as it propagates landwards. An inherent property of such channels is that there is a progression in tidal phase along the channel, accompanied by a relative phase difference between elevation and velocity of up to 90°. For the case of constant depth,  $h$ , the first-order solutions for elevation and velocity are given by:

$$\begin{aligned}\zeta &= a \cdot \cos(\omega t - kx) \\ u &= -U \cdot \sin(\omega t - kx)\end{aligned}\quad (4.3.8)$$

Where  $\zeta$  is the tidal elevation,  $a$  is the amplitude of tidal elevation,  $\omega$  is the tidal frequency,  $k$  is the tidal wave number,  $x$  is the distance along estuary from the mouth,  $u$  is the tidal velocity and  $U$  is the amplitude of the tidal velocity, which is given by (Friedrichs and Aubrey, 1994):

$$U = \frac{a \cdot \omega \cdot L_A}{h_s} \quad (4.3.9)$$

In this schematisation  $h_s$  is the average depth at mean tide level including the influence of the tidal flats, taken as a tidal average and  $L_A$  is the e-folding length for the convergence of cross-sectional area<sup>9</sup>.

#### 4.3.5.2 Balance of erosion and deposition

The rate of erosion and the rate of deposition (Partheniades, 1965; Whitehouse *et al.*, 2000), viz:

$$\begin{aligned}\frac{dm}{dt} &= m_e (\tau - \tau_{cr}) \quad \text{for } \tau > \tau_{cr} \\ \text{and } \frac{dm}{dt} &= -w_s \cdot \rho_c\end{aligned}\quad (4.3.10)$$

where  $w_s$  is the settling velocity,  $\rho_c$  is the mass sediment concentration,  $m_e$  is the erosion constant,  $\tau$  is the bed shear stress and  $\tau_{cr}$  is the critical bed erosion value.

The first equation of can be used to obtain the average erosion rate, by integrating  $u$ , as given by equation over the interval when the flow exceeds the critical threshold value. Assume that flow is symmetric, and if there is a balance on the flood this will also be the case for the ebb. Hence, integrate over a half tidal cycle and, i.e.:

$$\left\langle \frac{dm_{ero}}{dt} \right\rangle = \frac{2}{T} \int_{t_1}^{t_2} \frac{dm_{ero}}{dt} dt \quad \text{where } t_1 = \frac{1}{\omega} \sin^{-1} \left( \sqrt{\frac{\tau_{cr}}{\tau_a}} \right) \quad \text{and } t_2 = \frac{\pi}{\omega} - t_1$$

$$u = -U \cdot \sin(\omega t - kx); \quad \tau = \rho \cdot C_d \cdot u^2; \quad \text{at } x=0 \quad \tau = \rho \cdot C_d \cdot U^2 \cdot \sin^2(\omega t)$$

$$\text{With } \tau_a = \rho \cdot C_d \cdot U^2 \quad \text{and } \tau = \tau_a \cdot \sin^2(\omega t)$$

where  $C_d$  is the drag coefficient due to flow or combined flow and waves (Soulsby and Clarke, 2005).

$$\left\langle \frac{dm_{ero}}{dt} \right\rangle = \frac{2}{T} \int_{t_1}^{t_2} m_e (\tau_a \cdot \sin^2(\omega t) - \tau_{cr}) dt = \frac{m_e}{\pi} \left\{ \left( \tau_a - 2\tau_{cr} \right) \cdot \left( \frac{\pi}{2} - \sin^{-1} \left( \sqrt{\frac{\tau_{cr}}{\tau_a}} \right) \right) + \sqrt{\tau_{cr} (\tau_a - \tau_{cr})} \right\}$$

<sup>9</sup> Whereas,  $h$  is the average depth within the channel

$$\left\langle \frac{dm_{dep}}{dt} \right\rangle = \frac{2}{T} \int_0^{T/2} -w_s \cdot \rho_c dt = -w_s \cdot \rho_c$$

#### 4.3.5.3 Hydraulic and plan form tidal prism

Form-estimate of tidal prism:

$$P = 2a \cdot \cos(\varphi) \int_0^{L_e} W dx = 2a \cdot \cos(\varphi) \cdot W_m L_w \left( 1 - \exp\left(\frac{-L_e}{L_w}\right) \right) \quad (4.3.11)$$

$$P = 2a \cdot \cos(\varphi) \cdot W_m L_w \varepsilon_{L_w}; \quad \varphi = \varepsilon_A = k \cdot L_A; \quad \varepsilon_{L_w} = \left( 1 - \exp\left(\frac{-L_e}{L_w}\right) \right)$$

$$k = \frac{\omega C_r L_A}{g h_m}; \quad C_r = \frac{8}{3\pi} \frac{C_d U}{h_s}$$

The expression for  $k$  follows from Eq.(17) and (20) in F&A'94.

For the hydraulic estimate the discharge through the mouth is integrated over half a tidal cycle (assumes a symmetric tide).

The propagation of the tidal wave is described by:

$$\eta = a \cdot \cos(\omega t - kx); \quad u = -U \cdot \sin(\omega t - kx - \varepsilon_A)$$

At the mouth,  $x=0$ :  $\eta = a \cdot \cos(\omega t)$ ;  $u = -U \cdot \sin(\omega t - \varepsilon_A)$ ;  $\varepsilon_h = a/h_s$

$$P = \int_{\frac{\varepsilon_A}{\omega}}^{\frac{\varepsilon_A + \pi}{\omega}} U \sin(\omega t - \varepsilon_A) \cdot A_m \left( 1 + \frac{a \cdot \cos(\omega t)}{h_s} \right) dt = \frac{U A_m}{2\omega} (4 - \pi \varepsilon_h \cdot \sin(\varepsilon_A)) \quad (4.3.12)$$

Equating the solutions for prism form and prism hydraulics, gives:

$$2 \cdot \cos(k \cdot L_A) \cdot W_m L_w \varepsilon_{L_w} = \frac{L_A}{h_m} \frac{A_m}{2} (4 - \pi \varepsilon_h \cdot \sin(k \cdot L_A))$$

$$\cos(k \cdot L_A) \cdot \varepsilon_{L_w} = \frac{L_A}{L_w} (1 - \varepsilon_H \cdot \sin(k \cdot L_A)); \quad \varepsilon_H = \frac{\pi \varepsilon_h}{4} = \frac{\pi a \beta}{4 h_m}$$

$$L_w \cdot \varepsilon_{L_w} = L_A \left( \frac{1 - \varepsilon_H \cdot \sin(k \cdot L_A)}{\cos(k \cdot L_A)} \right) \quad (4.3.13)$$

Solve for  $L_w$ , noting that this is a function  $L_A$ .

#### 4.3.5.4 Model solution

To complete the dimensioning of the system it is necessary to know what the estuary is converging to. In this case the river channel at the tidal limit is used to define the width and cross-sectional area that the converging channel has to match at this point. This is achieved by using the river discharge,  $Q_r$ , and energy slope,  $s_r$ , at the tidal limit in a suitable river regime equation (eg. Cao and Knight, 1996) to determine the width and hydraulic depth of the river channel,  $W_r$ , and  $h_r$ . For simplicity the river discharge is assumed given (or based on the surrounding catchment in some way) and the energy slope is taken as the known valley slope at the tidal limit or the ratio of the tidal range and the distance from the mouth to the tidal limit,  $L_e$ , if this is not known.



Once the width e-folding length and river width are known, equation (4.3.2) can be used to estimate the mouth width,  $W_m$ . The mouth cross-sectional area is then simply the depth,  $h_s$  x  $W_m$ .

Putting  $L_A = L_W$  so that  $h$  is constant, and  $\varepsilon_{LW} = 1$ :

$$\cos(k \cdot L_A) = (1 - \varepsilon_H \cdot \sin(k \cdot L_A)) \quad (4.3.14)$$

Which has a solution:

$$L_A = L_W = \frac{1}{k} \tan^{-1} \left( \frac{2\varepsilon_H}{1 + \varepsilon_H^2}, \frac{1 - \varepsilon_H^2}{1 + \varepsilon_H^2} \right) \rightarrow \frac{1}{k} \tan^{-1} \left( \frac{2\varepsilon_H}{1 - \varepsilon_H^2} \right) \quad (4.3.15)$$

In the model Eq.4.3.15 is used to obtain  $L_W$  and  $L_A = L_W \cos(\varepsilon_A)$ ;  $\varepsilon_A = k \cdot L_A$  is solved iteratively, given  $L_W$ . Retaining  $\varepsilon_{LW}$  is more complex and the equations used are detailed in Appendix 3.

#### 4.3.6 Gross properties

The resulting 3D form model allows several integral properties to be derived as measures of the overall size of the estuary. For instance, simply integrating equations (4.3.1) and (4.3.2) provides the volume and surface area at mean tide level:

$$V_0 = A_{mtl} \cdot L_A \cdot \left[ 1 - \exp \left( \frac{-L_e}{L_A} \right) \right] \quad \text{and} \quad S_0 = W_{mtl} \cdot L_W \cdot \left[ 1 - \exp \left( \frac{-L_e}{L_W} \right) \right] \quad (4.3.16)$$

Using equation (4.3.7), estimates can also be obtained for the volume and area of the low water channel and intertidal flats. The slope of the lower intertidal,  $m_{eq}$  - see Figure 3, is defined as:

$$m_{eq} = \frac{L_{pp}^*}{a} \quad \text{and we define the variable} \quad m_{eqS} = \frac{L_{pp}^* \cdot L_e}{a} \quad (4.3.17)$$

Where, if  $\theta_{av}$  is used from equation (4.3.5), these are average values for the estuary as a whole. The relationships in equation (4.3.7) then provide a means of calculating the widths, or equivalently the surface areas, for the channel and intertidal. By integrating equation (4.3.7) over the intervals to which they apply, cross-sectional areas are obtained. This leads to the following simple expressions for surface areas,  $S$ , and volumes,  $V$ , of the channel and intertidal flats:

$$\begin{aligned} S_{lw} &= S_0 - n_{bk} \cdot a \cdot m_{eqS} & V_{lw} &= V_0 - \frac{a}{2} \cdot (S_{lw} + S_0) \\ S_{fl} &= \left( 1 + \frac{\pi}{2} \right) \cdot n_{bk} \cdot a \cdot m_{eqS} & V_{fl} &= \frac{3 + \pi}{2} n_{bk} \cdot a^2 \cdot m_{eqS} \end{aligned} \quad (4.3.18)$$

#### 4.3.7 Influence of waves

The wave field itself is based on a wind speed,  $U_w$ , blowing over a fetch  $F$ , using some form of parametric model. In this case the TMA spectrum (Bouws et al., 1985; Hughes, 1984) has been used because this includes the influence of depth limitation on the wave spectrum. The largest fetch was assumed to occur at  $\sim 45^\circ$  to the shore, which was estimated as a function of the width at any point as  $\sqrt{2}$  x width. Estimates of the significant wave height,  $H_s$ , and the mean zero crossing period,  $T_z$ , were obtained from the spectral moments, with the peak wave period,  $T_p$ , obtained using the empirical relationship proposed for wind-waves by Donelan *et al.* (1985).

A similar argument to that used for tidal flows can be used to determine the critical depth for given wave conditions. This is again based on a hydraulic equation and a sediment balance equation. Representing the wave using linear wave theory, the balance of erosion and deposition can be obtained by integrating over a wave cycle, using the maximum wave orbital velocity given by:

$$u_w = \frac{1}{2} \sqrt{\frac{g}{d_e}} H \quad (4.3.19)$$

Where  $H$  is the root mean square wave height ( $\sim H_s/\sqrt{2}$ ), this being a more appropriate value for estimating the bed shear stress due to wave action (Whitehouse et al., 2000). Making use of

$\tau_0 = \rho c_d u_w^2$  (where  $c_d$  equates to the wave friction factor,  $f_w/2$ ) with the above expression for the wave orbital velocity, an iterative solution can be used to obtain an estimate of depth at which waves will begin to influence the profile for a given tidal stage,  $d_e$ . The magnitude of the drag coefficient,  $c_d$ , is defined by the maximum of the respective wave friction factors for smooth or rough turbulent flow conditions (Soulsby and Clarke, 2005), which making use of equation (21) can be expressed as:

$$f_{ws} = 0.095 \left( \frac{gTH^2}{\nu d} \right)^{-0.187} \quad \text{or} \quad f_{wr} = 0.237 \left( \frac{\sqrt{gd}TH}{4\pi k_s d} \right)^{-0.52} \quad (4.3.20)$$

The form of the wave generated profile is obtained by considering the conservation of energy across the profile (Dean, 1991; Friedrichs and Aubrey, 1996) to obtain a profile that has a uniform distribution of wave orbital velocity and is defined by:

$$d = d_e \left( 1 - \frac{y}{L} \right)^{2/3} \quad (4.3.21)$$

Given the depth for the initiation of wave influence on the channel profile, the length of the profile,  $L$ , is given by (Friedrichs and Aubrey, 1996):

$$L = \frac{3\pi}{4c_d} \frac{d_e^2}{H} \quad (4.3.22)$$

The shear stress due to waves is  $\tau_w = \rho \cdot C_{dw} \cdot u_w^2$  and balancing erosion and deposition gives the same equation as used for flow:

$$\frac{m_e}{\pi} \left\{ (\tau_w - 2\tau_{cr}) \cdot \left( \frac{\pi}{2} - \sin^{-1} \left( \sqrt{\frac{\tau_{cr}}{\tau_w}} \right) \right) + \sqrt{\tau_{cr} (\tau_w - \tau_{cr})} \right\} - w_s \cdot \rho_c = 0 \quad (4.3.23)$$

For the model presented here it is assumed that whilst waves will influence the profile at all stages of the tidal cycle, the predominant influence on the *form* will occur during the high and low water stands, when the extended time at a fixed elevation increases the probability that waves will be able to leave a signature that tends towards the equilibrium profile. At high and low water estimates are made of the fetch based on the width of the estuary, and with the average depth of the estuary and the prevailing wind speed, the wave height and period are obtained using the TMA spectrum.

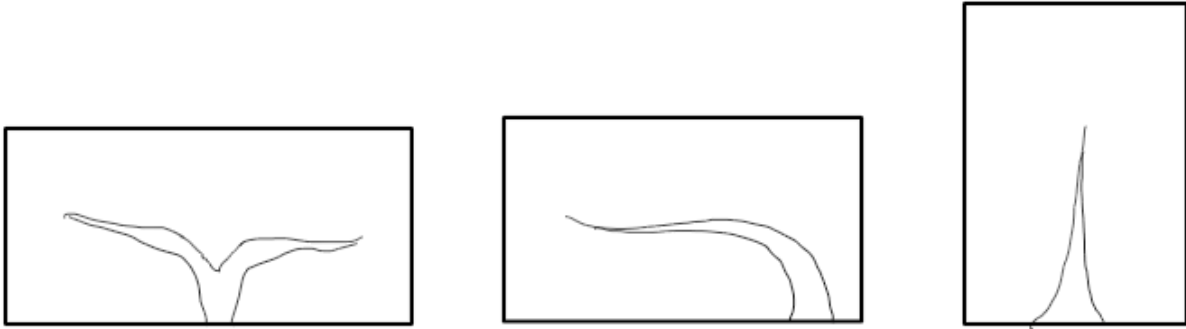
Modifying the concentration (specified at mean tide level) to take account of water depth changes, equation (4.3.9) is used with equations (4.3.16) and (4.3.17) to provide the wave profile depths at high and low water, from which the width of the two profiles follows from equation (4.3.20) and the elevations across the profile can be determined using equation (4.3.19). As these two profiles can now be defined relative to known water levels (high and low water) and points on the flow-only-profile

where the depth equals the equilibrium depth for the wave profiles, the overall cross-section can be modified to include the wave profiles.

#### 4.4 Basis of the Tidal Inlet model

The Tidal Inlet model is a geometric construction comprising a channel cross-section, an intertidal cross-section, a plan form that defines the along channel variation in width at low water, and a basin plan form that defines the along channel variation at high water. Several options are provided for each of these components. The construct is similar to that used for the Exponential Form model (Section 4.1), the primary differences being the inclusion of an inlet channel and the definition of high water as either a rectangle, or an ellipse. In this model the inlet is represented by a single straight low water channel and a flanking tidal flat. For this the landward tidal length ( $L_t$ ) > lateral width ( $W_t$ ). Where the topographic controls force the basin to be wider than the length ( $W_t > L_t$ ) this is assumed to be equivalent to a basin with the length and width reversed, as illustrated in Figure 5.

Figure 5 – bifurcating and offset inlets are represented by a straight low water channel and equivalent intertidal area by swapping basin width and length.



The inlet channel over the inlet length,  $L_i$ , is defined by the widths and depth specified at the mouth with a channel cross-section based on Eq. 4.1.4 and an intertidal profile based on the selected form, as described in Section 4.1.3. Eq. 4.1.4 is also used to define the river channel landward of the tidal limit if present. Over the channel reach within the inlet, the low water channel can be defined by an exponential convergence, or a power law in plan (see Section 4.1.1) and a cross-sectional form based on Eq. 4.1.4. The high water limit over this reach is defined either as a rectangle, with a width extending to the defined basin width,  $W_t$ , or as an ellipse defined by:

$$y_i = \sqrt{\frac{4b_t^2}{L_t - L_i} x_i \left( 1 - \frac{x_i}{L_t - L_i} \right)} \quad \text{where } b_t = W_t/2 \quad (4.4.1)$$

The intertidal profile is again based on the selected form, as described in Section 4.1.3. In this implementation the profile is “stretched” to fit between the defined low water and high water widths. Beyond the end of the low water channel, the landward slope is defined in the same manner.

#### 4.5 Adding variations in along-channel water level surface

When creating a channel form from one of the 3 options available the user is prompted to select a method to define the along-channel water level surfaces. These options use:

- (i) the CSTmodel to define a kinematic water surface that takes account of tidal propagation and river flow;
- (ii) constant HW and a linear increasing LW (i.e. rising to meet HW at the tidal limit); and
- (iii) constant HW and LW.

The initial values of mean sea level and tidal amplitude defined in *Setup>System Parameters>Hydraulic Parameters > Tidal Forcing* are updated based on any changes in sea level or tidal amplitude (see Section 5.2.2).

The first option is provided to capture the influence of tidal propagation in relatively long estuaries. In many instances the resulting water level surfaces are not horizontal. In addition, the river discharge can alter the mean tide elevation and the levels of high and low water about the mean tide level. These variations are neatly captured by the model of Cai et al (2016). This has the merit that it includes tidal amplitude and river discharge as forcing conditions but only requires some basic information about the dimension of the system: width and area at the mouth and at the head, rates of convergence, extent of the storage area and the bed friction.

It should be noted that the addition of the water surfaces, to set the levels for the channel form surroundings (i.e. the flood plain) and redefine the level of key change points in cross-sections along the channel (i.e. HWL, MTL and LWL), is a simple geometric superposition of water level surfaces. There is no inference that the resultant sections are dynamically linked to the hydrodynamics, although the assumptions underpinning the idealised channel forms being used means that they should reflect the prevailing hydrodynamics.

#### 4.6 Valley Model

This model generates an independent valley form, typically to represent the antecedent morphology prior to the advancement of a tidal channel into the fluvial valley due to marine transgression in response to sea level rise. The along valley and cross-valley profiles are presumed to vary exponentially (to be consistent with the uniform energy and minimum work arguments proposed by Leopold and Langbein (1962)). The longitudinal profile is defined by an exponential with the rate of convergence determined by the water level (high water) at the tidal limit,  $z_r$ , the level at the head of the valley,  $z_H$ , and the level of the base of the Holocene at the mouth,  $z_0$ . This two-dimensional form (x-z) is made three-dimensional by defining a lateral profile (y-z). The same reasoning is applied to the lateral dimension - that elevations adjust (subject to any imposed constraints) to do minimum work - and an exponential form is adopted. The valley model therefore uses a longitudinal profile given by transforming elevations to  $\Delta z_r = z_r - z_0$  and  $\Delta z_H = z_H - z_0$ :

$$z_v(x) = z_{m0} \left( \exp \left( \frac{x}{L_v} \right) - 1 \right) + z_0 \quad \text{where} \quad z_{m0} = \frac{\Delta z_H}{\exp(x_H/L_v) - 1} = \frac{\Delta z_r}{\exp(x_r/L_v) - 1} \quad (4.5.1)$$

The total length from the estuary mouth to the tidal limit is  $L_e$ . The bed elevation at the tidal limit is  $z_{TL}$  and at the mouth is  $z_{m0}$ . The lateral bed profile for one side of the valley is then given by:

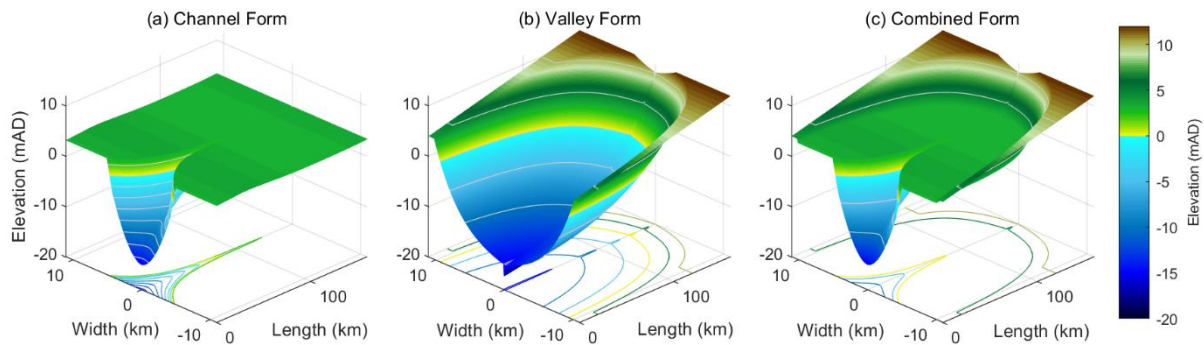
$$z(x, y) = z_{m0} \left( \exp \left( \frac{y}{L_s} \right) - 1 \right) + z_v(x) \quad \text{where} \quad L_s = \frac{B_{mV}}{\log \left( (z_{hw} + z_{m0} - z_0) / z_{m0} \right)} \quad (4.5.2)$$

where the half-width of the valley at the mouth is  $B_{mV}$ , at the level of high water. Equation (4.5.2) generates a 3-D form that converges along the length of the river (see Figure 6), which is a simplification in the absence of more detailed knowledge of estuary valley forms.

#### 4.7 Combined channel and valley form

By taking the maximum elevation at each grid point, the channel and valley forms can be combined to give the composite morphology as illustrated in Figure 6 (*Utilities> Add Form to Valley*: see Section 3.5). Typically this results in grid points above high water being taken from the Valley model, whereas everything at or below high water is based on the Channel form model.

Figure 6 - illustration of the combination of an estuary form (a) and a valley form (b) to give an estuary and surrounding flood plain embedded in a valley (c).



#### 4.8 Shoreline form

A shoreline strip can be added to Channel, Valley or Combined form models (*Utilities>Add Shoreline*: see Section 3.5). There are two options:

- (i) *Utilities>Add Shoreline>Model Shore*: generates a beach profile using Dean's equilibrium model on the open coast and the defined offshore bed slope across the mouth of the channel. The resulting grid is extended seawards by the distance to the closure depth specified in *Setup>Form Parameters>Shore Parameters*.

The idealised beach profile is based on a linear slope (upper beach slope) between the beach crest level and mean tide level (SWL=0mOD) and an equilibrium profile out to deep water. This uses the Dean equation (Dean, 1977):

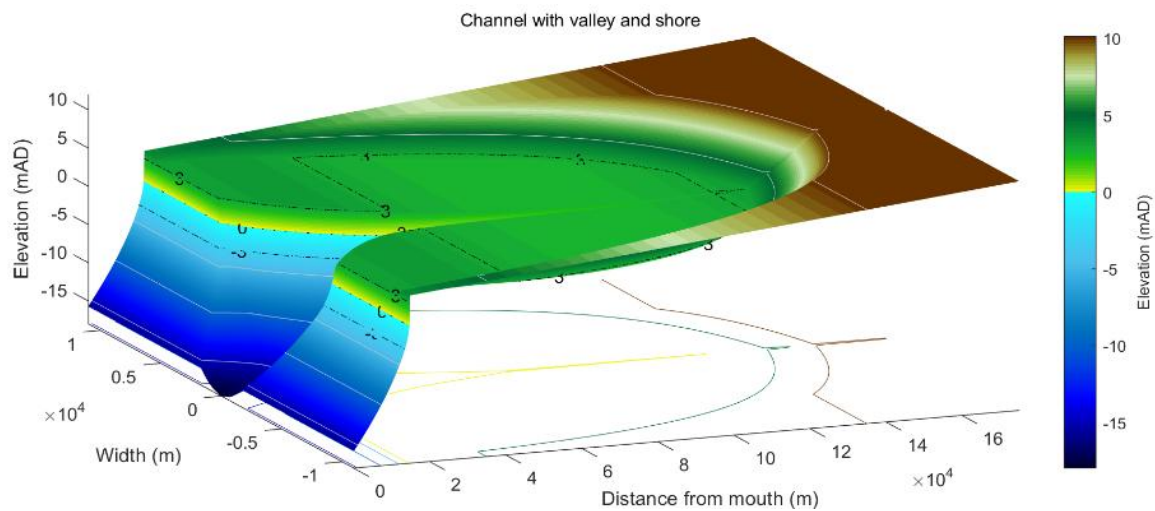
$$h = Ay^{2/3}$$

In the Dean formulation, A is a function of sediment fall velocity. Here it is derived from the depth and distance of the closure depth specified (see deanbeachprofile.m for details). The resulting combined channel+valley+shore model is shown in Figure 7.

- (ii) *Utilities>Add Shoreline>Extrapolate Shore*: allows the user to define a shore width, offshore depth and an exponent. This fits a power law between the levels along the seaward boundary of the source grid and the specified offshore depth. An exponent of 1, gives linear extrapolation between the grid shore values and the defined offshore value, a value greater than one gives a shore that is concave down and a value less than one gives a shore that is convex up.



Figure 7 – Illustration of a form comprising channel, valley and shoreface components.

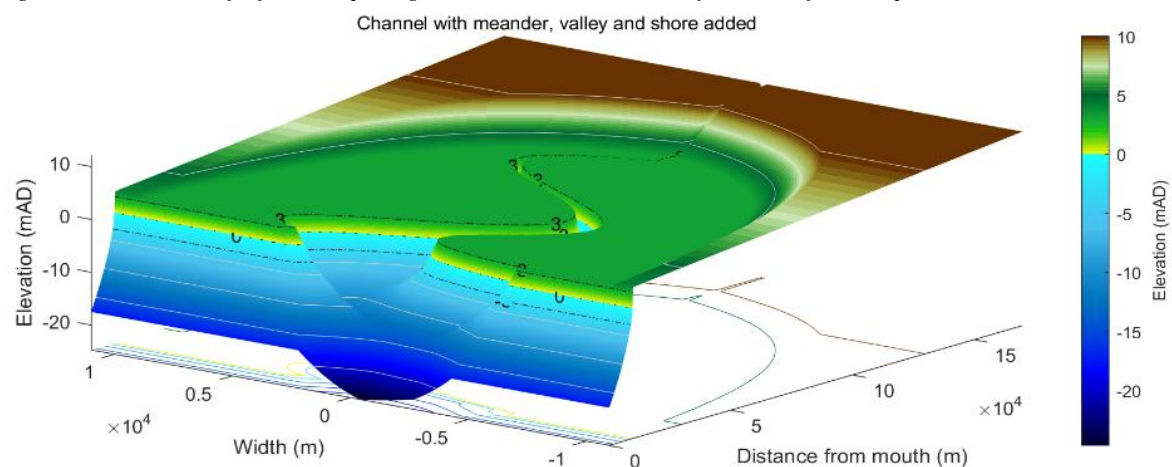


## 4.9 Meanders

The idealised channel form generated by the model uses a straight convergent channel. A meander can be imposed on this form by defining the required centreline (*Utilities>Add Meander*: see Section 3.5). This is loaded as an ascii text file with a one-line header that defines the data format, usually '%f%f', followed by two columns of data that define the x- and y-co-ordinates of the thalweg or centreline of the channel. The x-y co-ordinates have a zero at the central point of the mouth (thalweg) with x in the mean direction of the channel and y defined as +/- distances either side of the mean transect. Once added the centreline is saved along with the grid and can be used in the transgression model.

It is sometimes necessary to introduce an offset at the mouth to avoid slithers being created when the grid is transformed to curvilinear and interpolated back to cartesian co-ordinates. When adding a meander to a channel form the user is prompted to define an offset. This should be a distance equal to several grid cells (typically 5-8 seems to suffice for the cases tested so far). When a meander is being added to a channel with a shoreline, any offset is added from the mouth and consequently an offset may not be required unless the initial curvature is particularly severe.

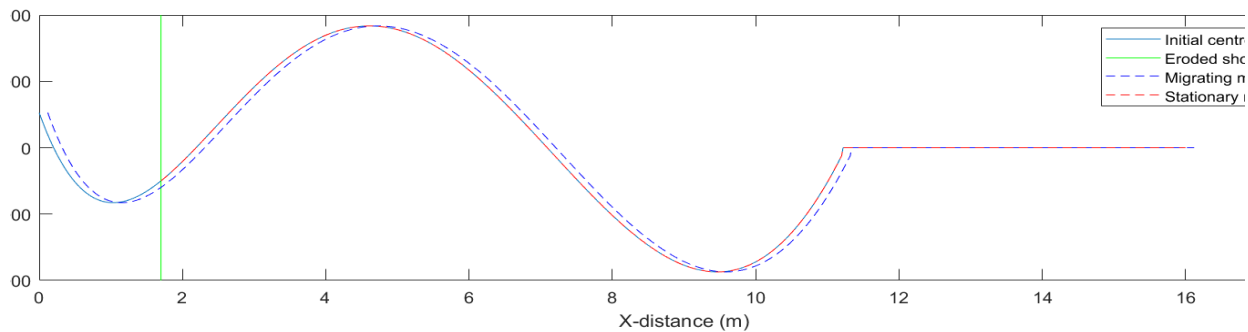
Figure 8 – Illustration of a form comprising channel with meander, valley and shoreface components.



The transgression model only includes a meander if the source channel has a meander and the option in *Setup>System Parameters>Transgression Parameters* has been set to 1 or 0. A value of NaN excludes the meander when the channel is updated. A value of 1 migrates the meander coordinates landward at the rate of the open coast (i.e., the open coast erosion rate, whereas a value of 0 keeps the

meander coordinates fixed in their initial position within the model domain. The two meander migration options are illustrated schematically in

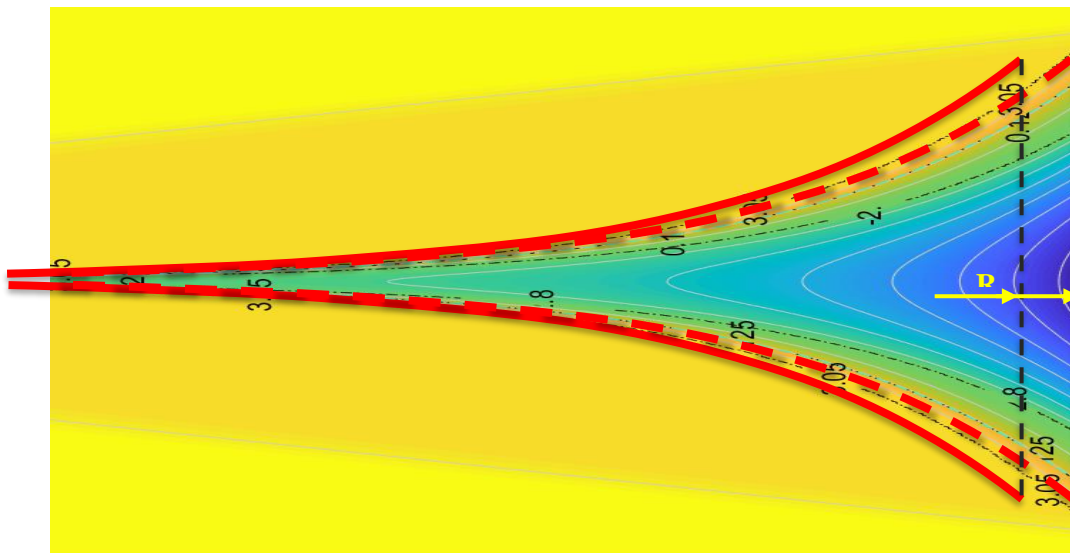
Figure 9 – illustration of static and dynamic meander migration options, with the latter translating at the same rate as the shoreline erosion.



## 5 Background to Transgression Model

The concept examined with the Transgression Model is that, for a defined amount of sea level rise, the system can adjust by moving landward and requires zero sediment exchange to do so (i.e. no sediment import or export). This is achieved by the form moving upwards and landwards to maintain its form relative to the mean tide level. The computation considers a Lagrangian estuary form and seeks the distance that the form needs to move for there to be no net volume change (R in Figure 10). This is measured over the area of the translated form, which means there is a volume to seaward that is not considered (see Figure 10). What happens to this volume will be a function of how the open coast adjacent to the estuary responds to sea level rise.

Figure 10 – Illustration of marine transgression





- (ii) where the adjustment takes account of the additional sediment demand of the flood plain.

The transgression distance and pattern of change using option (i) are consistent with those reported by Townend & Pethick (2002).

## 5.1 Model Formulation

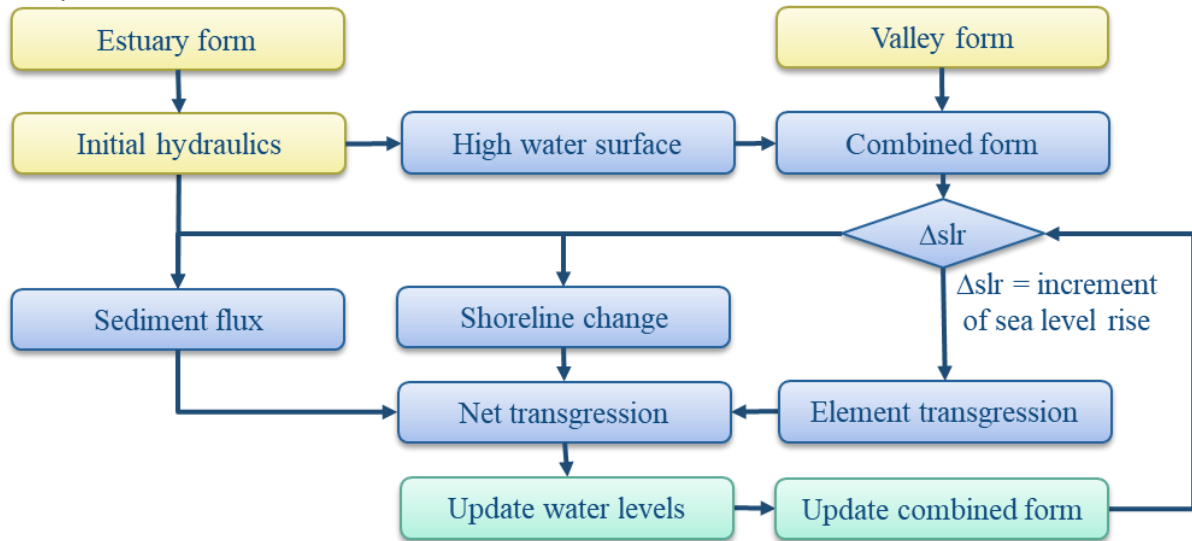
The main components of the model are:

- (i) an idealised form model to define a channel morphology,
- (ii) a model, or imported, DTM to define the antecedent valley (i.e. the morphology prior to the Holocene transgression).
- (iii) the variation of water levels at the mouth of the estuary.
- (iv) a hydraulic model to compute the high and low water surfaces along the length of the channel. In the current version of the model a constant river discharge is assumed.
- (v) a function to find the transgression distance as a function of sediment import/export.
- (vi) a simple representation of open coast shoreline retreat based on the Bruun relationship.

The computation procedure is illustrated in Figure 11. Once a combined form is created from the estuary form model, with a high water surface defined using the hydraulic model and the form of the valley superimposed, an estimate of the landward transgression for zero sediment import/export can be made, based on a defined change in sea level (the static case). By introducing a rate of sediment import/export from the surrounding environment an iterative loop is introduced (Figure 11). At each time step (order of one decade in this model), the mean water level is raised by the sea level rise over the time increment. Simply raising the existing form vertically by an amount equal to the sea level rise increment and computing the volume between the two surfaces, over the area of the estuary and flood plain, defines the volume of sediment required for the form to remain *in situ*. Progressively moving the raised form landward reduces the volume required, as internal erosion provides some of the sediment required to raise the form to the new level, Figure 12. The greater the landward movement of the form the greater the erosion such that after a certain distance the system would need to export sediment to maintain the form. The volume change is computed for a range of landward transgression distances using the combined form and any imposed constraints (e.g. include/exclude the flood plain). The sediment flux (import/export from sea and river) is then used to interpolate the landward transgression distance,  $\delta X$ , for zero net volume change. The transgression distance is used to update the widths, cross-sectional area and depth at the mouth. This allows the hydraulics to be updated and a new channel form to be defined with an updated length to the tidal limit that takes account of changes in tidal propagation and any change in the intersection with the river valley. Summing the incremental change,  $\delta X$ , over time gives the total transgression distance.

Figure 11– Model flowchart: yellow components define inputs, blue components denote computations and the green component is the output for each iteration. Element transgression refers to the distance moved by the initial form with zero external exchange and Net transgression adjusts the distance moved to account for the net sediment flux in or out of the

estuary.



If the coast is also subject to marine transgression (i.e. eroding) then the situation is slightly more complicated, as summarised in Figure 13. As well as the estuary transgression distance,  $estdX$ , there is a coast or shoreline transgression distance,  $cstdX$ . The seaward extension of the estuary form (or contraction if the difference is negative) is then given by difference between these two values (i.e.  $edX = estdX - cstdX$ ). The estuary length is altered by the amount of coastal transgression,  $cstdX$  and any change in tidal propagation and hence change in the distance to the tidal limit. The width and  $csa$  at the mouth will also change depending on the amount of seaward extension, or landward contraction, of the original form at the mouth,  $edX$ .

Figure 12 – Example of net change due to marine transgression with zero net volume change. In the example shown, a sea level rise of 0.2m results in a transgression distance for zero volume change of 1610m. The dashed line in (a) indicates the high-water level after sea level rise. The difference plot highlights how the more horizontal surfaces accrete and the more vertical surfaces (e.g. sides of the channel) erode.

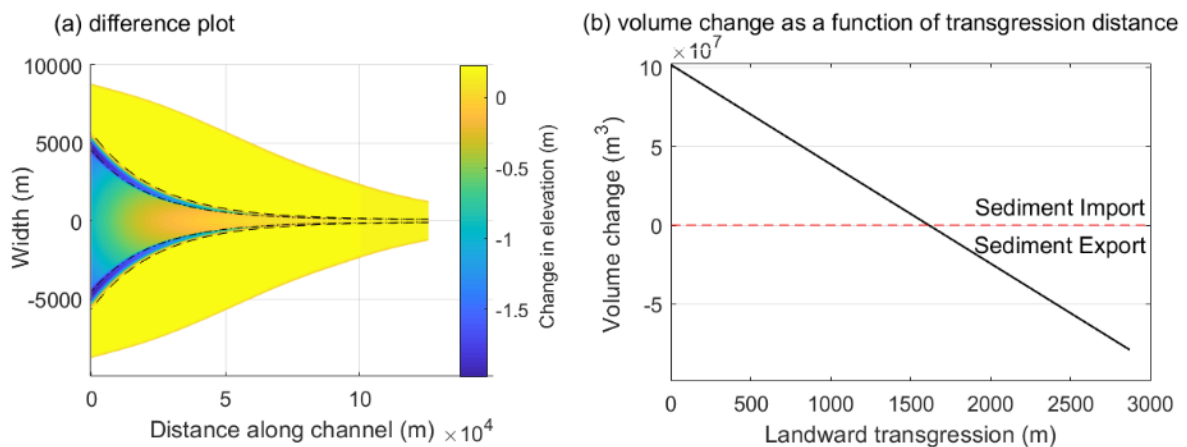
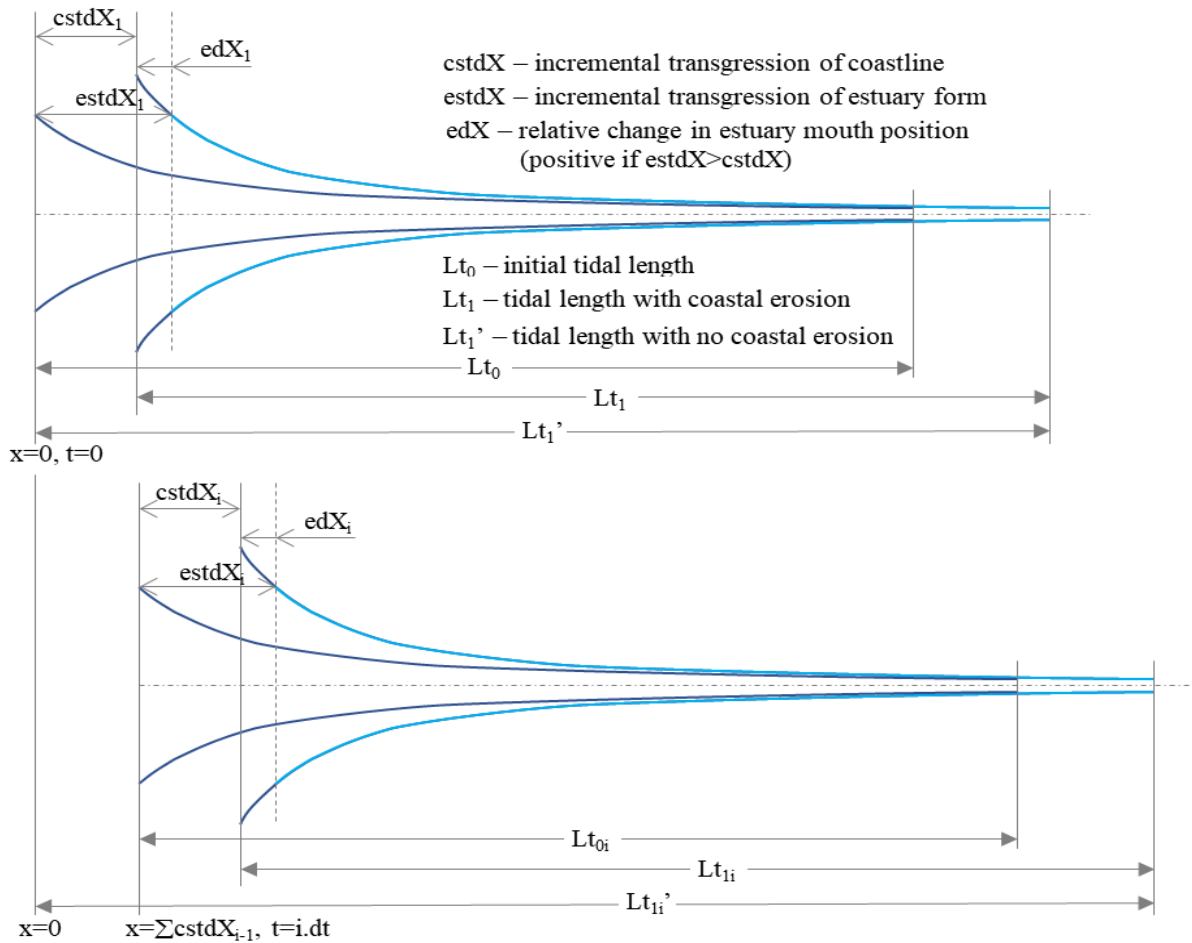


Figure 13 – Definition sketch of transgression distances and estuary length to the tidal limit when both estuary and shoreline are subject to transgression. If there is no coastal erosion, the mouth does not move landwards and the plan form extends to

$x=0$  (not shown for clarity).



## 5.2 Water Levels

### 5.2.1 Tidal Amplitude

The water level variation at the mouth is defined using the *Setup>System Parameters>Hydraulic Parameters > Tidal Forcing* UI. A symmetric tide is assumed given by the mean tide level at the mouth and the tidal amplitude. The tidal range can be varied by specifying one or more cycles. 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. Townend et al., 2007). These cycles can be used to represent decadal variations such as the nodal tidal cycle and longer-term variations.

### 5.2.2 Sea Level Rise

The rate of sea level rise is defined in *Setup>System Parameters>Hydraulic Parameters > Tidal Forcing* 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:

- a linear rate of sea level rise - *slr* input as a single positive value in m/yr.
- an exponential rate of sea level rise. The exponential rate increase (m/yr) is input as a negative value. The start of the exponential increase is set to 1900 and the rate prior to 1900 is taken as 0.001 m/yr.

- (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 [0.011,1900,0.001] gives the same as option (ii).
- (iv) extends option (iii) to include the Holocene. 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 provides a good approximation against data for the Humber estuary. The input would be: [0.011, 1900, 0.001, 1.32, 2400, 0.6].
- (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 3 in sealevelrise.m.

For example, by defining a rate  $\leq -0.01$  m/year, 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). [Note: for this to be used in the model run, the Start Year (set in Setup>Run Parameters>Time Step) must be prior to 1900.] However, this function can be replaced by any function that returns the amount of sea level rise at model run time from  $t=0$ .

### 5.2.3 Hydrodynamics

To define the water surfaces the model uses the solution proposed by Cai, Savenije and Toffolon (Cai, 2014). This model accounts for the influence of river discharge on tidal wave propagation and hence damping to give the along channel variation of water levels (high, mean, low). The analytical solutions were obtained by solving a set of four implicit equations, i.e. phase lag equation, scaling equation, celerity equation, and damping equation, derived from the conventional 1-dimensional conservation of mass and momentum equations (Savenije, 2012). In addition to the tide and river flow conditions, the variations in width and cross-sectional area along the channel must also be defined. Hence, the model is well matched to the idealised forms being examined here. The resultant residual water level is accounted for using an iterative procedure to account for non-linear friction. The model reproduces the main tidal dynamics (i.e. tidal amplitude, velocity amplitude, wave celerity, and phase lag between elevation and velocity) along the estuary axis. This is referred to here as the CST model. For a full description of the model, readers should refer to Cai et al. (2014; 2016) and related papers.

### 5.3 Sediment Flux

The transgression model seeks a landward distance that gives a sediment mass balance. However, this will also be modified by the supply or demands imposed on the estuary system by the surrounding environment. Whilst some systems are known to export sediment (e.g. Gao, 2007), others have been shown to import sediments (e.g. Rees, 2006). There is therefore a need to adjust the geometric mass balance by the volume imported or exported over a given time interval. This changes the simple transgression model from a time-independent geometric shift, to a time dependent evolution of the volume (assuming that the estuary form is maintained relative to the mean tide level). To provide an estimate of the sediment import/export an aggregated sediment balance model – ASMITA (Stive and Wang, 2003) was used. For this application, a single element representation was adopted for the combined channel and intertidal volume. An estimate of the rate of import/export of sediment was obtained from the rate of change of the element volume and this requires the characteristic rates of sediment exchange and the background (equilibrium) sediment concentration to be defined (see Townend et al., 2016 for details). The equation for a single element is as follows:

$$\frac{dV}{dt} = \frac{1}{c_b} \cdot \frac{w\delta c_e S}{\delta + wS} \left[ \left( \frac{V_e(t)}{V(t)} \right)^n - 1 \right] \quad \text{and} \quad V_e(t) = \alpha \cdot P(t)^\beta \quad (5.3.1)$$

where  $V$  is the volume of the inlet,  $t$  is time,  $w$  is the vertical exchange coefficient,  $\delta$  is the horizontal exchange coefficient,  $c_E$  is the equilibrium concentration for the system as a whole,  $S$  is the surface area of the channel,  $c_b$  is the concentration of the bed,  $n$  is the concentration transport exponent,  $V_e$  is the equilibrium volume of the channel and  $P$  is the tidal prism.  $\alpha$  and  $\beta$  are constants that characterise the equilibrium condition (typically based on a relationship such as that proposed by O'Brien and Jarret, see Townend (2005)). In addition,  $\tau$  is the morphological time scale given by:

$$\tau = \frac{1}{c_E \cdot n} \left( \frac{V_e}{w_s \cdot S} + \frac{V_e}{\delta} \right) \quad (5.3.2)$$

This can be computed for the current model settings using *Utilities>Morphological Timescale*.

The input parameters are updated each time step, using the evolving channel form, except for the equilibrium parameters. These are based on the initial channel volume and prism ( $\alpha=V/P$ ,  $\beta=1$ ) unless specific values are defined (*Setup>System Parameters>Sediment Parameters*).

## 5.4 Shoreline Change

The inclusion of change to the open coast shoreline is simply to provide a context and an indication of relative change. For this reason a simple approach is adopted based on the well known Bruun relationship. This is essentially a cross-shore mass balance where erosion at the top of the beach profile is balanced by deposition offshore, so that the profile maintains its form relative to the sea level (Bruun, 1962). In the transgression model the increment of retreat is obtained from;

$$\Delta s = \xi \cdot slr \quad \text{where} \quad \xi = L/h \quad (5.4.1)$$

where  $h$  is a characteristic closure depth for the adjacent beaches,  $L$  is the distance offshore to the closure depth and  $slr$  is the change in sea level over a given time interval.

## 5.5 Constraints

Various constraints can be imposed on the model. These include modifications to the model generated channel form as specified in *Setup>System Parameters>Morphological Modifications* and the various constraints that can be set in *Setup>System Parameters>Transgression Parameters*.

The Morphological Modifications option is used to introduce fixed cuboid forms into the model (e.g. to represent a reclamation or a dredged pocket). These modifications that are spatially fixed and are only included in the Transgression Model if they have been added to the Channel Form that is used to define the initial channel using the *Utilities>Add Modifications* option.

In addition, a number of options can be switched on/off using the settings defined in *Setup>System Parameters>Transgression Parameters*. The options include:

1. Including the flood plain in the mass balance. When set to true (1), only the channel is used to compute the transgression distance and no sediment is added to the flood plain. However the channel margins can erode and it is assumed that the area immediately adjacent to the channel does accrete (applied over a distance of 4 y-axis grid cells).
2. A high water constraint is equivalent to adding an embankment or sea defence. The flood plain is excluded from the model and the channel margins cannot erode. In the modelled grid an embankment is added, standing 2 m above the initial high water mark.
3. Geological constraints are defined over lengths of channel by a start and end distance from the initial mouth. These are only included when the Geological Constraints option is set to true (1). Like the Morphological Modifications, these are spatially fixed. However, the channel form can accrete over constrained reaches but it cannot erode.

## 5.6 Model Configuration

Defining a set of parameters that work together for all the different elements of the model can be challenging. Several utilities are provided which are designed to aid the process. The tool is designed for large scale systems that are typically 10s of kilometres in length. The following summarises a typical workflow to set up a model.

### 5.6.1 System dimensions and properties

The channel dimensions can be developed using charts and maps, or a suitable DTM and GIS tools<sup>10</sup>. Alternatively, a bathymetry can be loaded into ChannelForm and *Setup>Add Properties* used to get the gross properties of the channel. The results can be viewed on the *Form-Props* tab.

This may require various manipulations to get sensible results, including:

- Subsampling the model domain to focus on the area of the channel
- Careful definition of the mouth location (this can be adjusted using *Setup>Edit Inlet Definition*.)
- Straightening of the channel using a conformal mapping (using *Setup>Grid Tools>To Curvilinear*, see Section 3.4.5).

For the valley, a suitable DTM can be used. Again, this may need to be conformably mapped, to be approximately straight, if it is to be used in the transgression with an idealised channel form model. Maps and geological records (e.g., sections of the Pleistocene valley base) can also be used to dimension an idealised valley (see Section 3.4.1). The Utilities for River Dimensions, Valley Thalweg and Area of Flood Plain can be used to help develop a suitable idealised valley form (see Section 3.5).

### 5.6.2 Hydraulic parameters

If using the CSTmodel to compute along-channel elevations, it is often helpful to carry out some runs in the CSTmodel App. This allows the parameter settings for the model to be checked, over the range of parameters anticipated, to ensure that the model converges to a solution. Estuary length, rates of convergence (width and csa), Manning friction coefficient and the *Distance Increment Run Parameter* can all influence the model behaviour for given channel dimensions.

### 5.6.3 Minimum data requirements for models

For the exponential and power form models the minimum data input required includes:

- The form properties: *Setup>Form Parameters>Exp Form Parameters*, Or *Setup>Form Parameters>Power Form Parameters*.
- Water levels at boundary (tidal amplitude and mean sea level): *Setup>System Parameters>Hydraulic Parameters > Tidal Forcing*.
- The sediment parameters to define the marsh (if included) and the river regime: *Setup>System Parameters>Sediment Parameters*. [If neither are required, use *Setup* to create a null entry – accept default values and set others to zero].
- The model grid: *Setup>Run Parameters>Grid Parameters*

And, to add a water surface variation along the channel requires: *Setup>System Parameters>Hydraulic Parameters > Hydraulic Model*.

---

<sup>10</sup> DTM – digital terrain model; GIS – Geographical Information System



### 5.6.4 Kinematic and Quasi-dynamics models

A kinematic version of the model can be run by setting a value of NaN for the sediment flux (as defined in *Setup>System Parameters>Transgression Parameters*: (see Section 3.4.2) or when the transgression model is run (see Section 3.6). This uses a single time step based on the step duration x number of time steps (see Section 3.4.3 for Run Parameters settings).

The alternative is the quasi-dynamic mode, which updates the morphology at each time-step in the simulation. Depending on the input definition this updates all morphological components included and the water surfaces for updated channel morphology (running the CSTmodel if this was used in the input channel model). In addition, the transgression distance is modified by any sediment exchange with the external environment. This can be a constant flux or computed based on the channel form at each time-step (see Section 5.1 for further explanation).

## 5.7 Model Convergence

The model can be sensitive to the time step and grid resolution. If either are too small this can result in relatively long run times. Conversely if the time step is too long or the grid too coarse, the model can be unstable, or the results erratic. In experiments for the models reported in Townend et al (2020), the grid resolution had only a small impact, whereas the time step was found to affect the results, especially if the sea level rise used was non-linear.

When the hydraulic model is used to define the along-channel water level surfaces the interval used to define along-channel resolution influences the convergence of the CSTmodel (DistInt is set in *Setup>System Parameters>Hydraulic Parameters > Hydraulic Model*). Reducing the value increases computation time and can make the model convergence over sensitive. Some trials may be needed depending on the size of the system being modelled in ChannelForm and the grid resolution being used. The default value of 5000 was found to work well for systems of the order of 100+ km in length.

When creating channel and valley form models, the input parameters are saved as part of the Case meta-data. These data are then used when running the transgression model (rather than the current input parameter settings). For most form model parameters, the channel and/or valley need to be re-run if the input parameters are changed before they can be used in the transgression model. However, some parameters can be reset during the initialisation of the model (see Section 3.6, *Run>Transgression model*.), including the definition of sea level rise (SLRrate), any cycles in the tidal range (CycleAmp), and the coefficients used in the sediment flux calculations (EqDensity, RiverDensity, TransportCoeff, EqScaleCoeff and EqShapeCoeff), or the option to define a constant sediment flux.

A number of input parameters are used interactively when the transgression model is run. This does not change the Input Parameters settings, as displayed on the tabs. However, the values included in the model Case metadata are the values at the end of the run and can be viewed by using *Project>Cases>View settings*. These values can be compared with the initial conditions by viewing the settings of the input Channel Case that was used. The values that can change include: xTidalLimit, zMouthInvert, HWmouthWidth, LWmouthWidth, HWwidthElength, LWmouthElength and the parameter values that the user can change when the transgression model is run, as noted above.

## 5.8 Study Workflow

In addition to the idealised simulations presented in Townend et al (Townend et al., 2020), ChannelForm can also be used to examine specific locations. The following workflow was developed to examine the Oka estuary in Northern Spain (Cearreta and Monge-Ganuzas, 2013; Garnier et al., 2022). The data available included a detailed channel bathymetry, digital terrain model of the surrounding valley, water levels, and conceptual model of the Holocene evolution of the estuary based



on detailed interpretation of several cores. The latter included an along-valley section of the Pleistocene valley floor.

### 5.8.1 Quick view

To examine how well the channel and valley forms fit together, they can be quickly loaded and combined. This allows additional modifications or adjustments for subsequent construction of models to be identified. For example, the vertical range of the channel will be typically of O[10m] whereas the surround valley can extend over 100s of metres. This can obscure the visibility of the channel when plotted and so an upper surface may need to be added to limit the vertical range (*Setup>Grid Tools>Add Surface*). There may also be a need to re-grid or sub-sample the grid domain.

- (i) Load the DTM for the valley into ChannelForm
- (ii) Load the DTM for the channel into ChannelForm
- (iii) In steps (i) and (ii) the grids can be re-oriented, if need be, so that they are aligned.
- (iv) Combine the two grids using *Setup>Grid Tools>Combine Grids*, using the Min option when prompted

### 5.8.2 Valley model

- (i) Load the valley DTM into ChannelForm. The valley needs to be aligned (approximately) to the x-axis, which may require the grid to be flipped and/or rotated as it is being loaded (graphical interface supports this). The distance from the grid origin to the mouth of the estuary also needs to be defined.
- (ii) If the valley is not straight, it may be necessary to apply a conformal mapping. This can be done by first digitising a line to define the valley centreline using *Setup>Grid Tools>Digitise Line*. (*No need to include elevations*) and then applying a mapping using *Setup>Grid Tools>To Curvilinear*. Set the Maximum number of iterations (default of 500 is fine in this case) and the Tolerance (the default is based on the average x interval of the centreline coordinates divided by 1000 and is ~5m in this case. A value of 5-10m usually works well). Note: the transformation can take some time to run.
- (iii) Extend shoreline seaward using *Utilities>Add Shoreline>Extrapolate Shore*.
- (iv) Digitise a line, to define the pre-Holocene valley base, along the approximate centre-line of the conformal valley using *Setup>Grid Tools>Digitise Line* and include elevations.
- (v) Add the pre-Holocene base to the conformal valley using *Utilities>Add Thalweg to Valley*. This uses the distance from the origin to the estuary/channel mouth to construct the mask. [Tip: to check the valley base against the current bathymetry along the same centreline run the function to generate the plot and then Quit (the Accept/Quit prompt prevents interrogation of the plot). The plot can then be rotated to check the positioning of the channel relative to the bathymetry]. The user is prompted for the upper cut-off, offshore mask width and valley base width. The upper cut-off is used to mask out the channel in the valley grid. The offshore mask width defines the width either side of the channel centreline to be adjusted to seaward of the existing estuary mouth and the valley base width replicates the thalweg depths for the defined width about the thalweg (i.e., +/- width/2).<sup>11</sup>

---

<sup>11</sup> The Holocene conformal valley can be mapped back to cartesian coordinates using *Setup>Grid Tools>From Curvilinear*. This does a reasonable job of the main valley and channel but loses most of the surrounding topography. To clean this up, use *Setup>Grid Tools>Add Surface* with a cut-off level chosen to remove most of the valley above the channel domain and a surface level of NaN, then combine with the source valley grid using *Setup>Grid Tools>Combine Grids*.

- (vi) To create a valley that can be used with the Form model, the Holocene conformal valley grid needs to be subsampled to the model domain and the axes set to match the model grid. This can be done using *Setup>Grid Tools>Sub-Grid*. Then use *Setup>Grid Tools>Translate Grid* to adjust the coordinates to match the model grid, e.g., by offsetting the grid origin and inverting the axes, if necessary. This may not produce an exact match to the model grid, so re-grid the valley model, using *Setup>Grid Tools>Re-Grid* with a form model grid as the template (Existing), or the defined settings (New).
- (vii) To improve the plot projections, reduce the vertical range, if required, using *Setup>Grid Tools>Add Surface*.
- (viii) If the resulting valley is still not very straight or is unduly complex, consider creating an idealised valley that captures the overall shape and dimensions of the actual valley. Define the idealised valley using *Setup>Form Parameters>Valley Parameters* and then create the valley using *Run>Valley Model*.

(ix)

### 5.8.3 Create conformal channel to extract channel properties

- (i) If there are areas of no data around the estuary channel DTM, add a horizontal surface using *Setup>Grid Tools>Add Surface* at a suitable elevation.
- (ii) Digitise a line to define the channel centreline using *Setup>Grid Tools>Digitise Line* (no need to include elevations).
- (iii) Generate a conformal channel using *Setup>Grid Tools>To Curvilinear*.
- (iv) Subsample the resulting grid to focus on the conformal channel using *Setup>Grid Tools>Sub-Grid*. Make sure the location of the mouth is correctly defined in this grid using *Setup>Edit Inlet Definition*.
- (v) Get the gross and along-channel properties for the resultant channel using *Setup>Add Properties*. Define water levels when prompted. The results can then be viewed on the *Form-Props* tab.

### 5.8.4 Create Form model

- (i) Use the conformal valley and channel grids and properties to extract the input parameters needed for the Form model. This is likely to include the following:  
*Setup>Form Parameters>Exp Form Parameters*  
*Setup>Run Parameters>Shore Parameters*  
*Setup>System Parameters>Hydraulic Parameters>Tidal Forcing*  
*Setup>System Parameters>Hydraulic Parameters>Hydraulic Model*  
*Setup>System Parameters>Sediment Parameters*  
*Setup>Run Parameters>Grid Parameters*  
*Setup>Run Parameters>Run Time Parameters*.
- (ii) Run the chosen Form model (Exponential Form, Power Form or CKFA Form).

### 5.8.5 Holocene Transgression

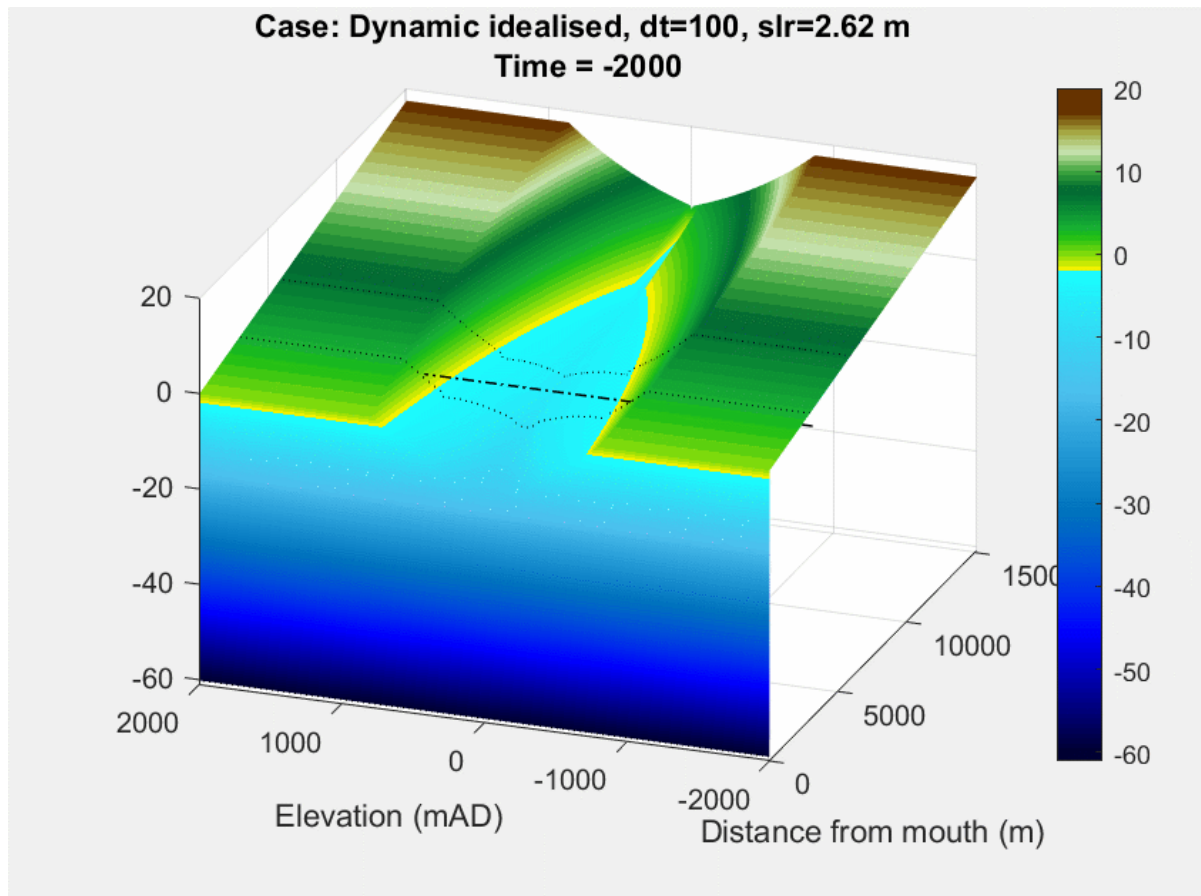
Note: Getting the model domain for the valley with a shore and Holocene basement to be compatible with the idealised channel model can take several iterations. If the run time with the chosen grids is excessively long this can be reduced by re-gridding to a coarser grid resolution.

- (i) Adjust the Valley model Distance to the mouth to define the required starting position for the Channel model. For example, setting the value to zero ensures that the channel model starts at the seaward boundary of the Valley model, whereas a value greater than zero will initialise the channel model at the specified distance from the boundary.

- (ii) Define the required transgression parameters using *Setup>Run Parameters>Transgression Parameters* (see Section 3.4.2).
- (iii) Run the transgression model in kinematic mode (no iteration), to check the setup, using *Run>Transgression Model*; ensuring that the Sediment Flux is set to a value of 'NaN'.
- (iv) Modify the run time parameters so that there are only a limited number of time steps of the required time-step (say between 1 and 10 depending on the computing power available).
- (v) If this all looks sensible, then the full runs can be made using *Run>Transgression Model*.

There are several tools provided to explore the results on the *Analysis* menu. These include the bespoke animation shown in Figure 14, which is obtained using *Analysis>Transgression Plots>Animation*.

Figure 14 – Animation of idealised Valley and Channel based on the dimensions of the Oka estuary.



## 6 Supporting Information

### 6.1 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 3.6). 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  $t$  (as long as 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,mobj) ',inp.eqn]); %handle to anonymous function

[varout{:}] = heq(t,x,y,z,mobj);
```

or when using dstables:

```
heq = str2func(['@(dst,mobj) ',inp.eqn]); %handle to anonymous function

[varout{:}] = heq(dst,mobj);
```

This function is then evaluated with the defined variables for  $t$ ,  $x$ ,  $y$ , and  $z$  and optionally  $mobj$ , where  $mobj$  passes the handle for the main UI to the function. Some functions may alter the length of the input variables ( $x$ ,  $y$ ,  $z$ ,  $t$ ), or return more than one variable. In addition, the variables selected can be sub-sampled when each variable is assigned to the  $X$ ,  $Y$ , or  $Z$  buttons. The dimensions of the vector or array with these adjustments applied need to be dimensionally correct for the function being called. This may influence how the output can be saved (see Section 6.1.2).

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 6.1.2 and 3.6).

*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.*

Equations can use functions such as  $\text{diff}(x)$  - 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 data set. This can be done by adding a NaN at the beginning or the end:

e.g.:  $[\text{NaN};\text{diff}(x)]$

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.

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,  $z$ .*

*NB3: When using Matlab compound expressions, such as the above sub-sampling expression, the expression must be enclosed in square brackets to distinguish it from a function call.*

Adding the comment `%time` or `%rows`, allows the the row dimension to be added to the new dataset. For example if `x` and `y` data sets are timeseries, then a Matlab™ expresion, or function call, can be used to create a new time series as follows:

```
x^2+y %time
```

### 6.1.1 Calling an external function

The Derive Output UI 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 ChannelForm are detailed in Section 6.1.3.

The input variables for the function must match the syntax used for the call from the Derive Output UI, as explained above. In addition, functions can return a single value, one or more vectors or arrays, or a dstable (see Section 6.1.2). If the variables have a dimension (e.g., *time*) then this should be the first variable, with other variables following. If there is a need to handle additional dimensions then use the option to return a dstable.

If there is no output to be passed back, the function should return a variable containing the string 'no output' to suppress the message box, which is used for single value outputs (numerical or text).

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 and assigned to the X, Y, Z buttons but they are defined in the call using *dst*, for example:

```
[time,varout] = myfunction(dst, 'usertext', mobj);  
  
dst = myfunction(dst, 'usertext', mobj);
```

where 'usertext' and *mobj* are call strings and a handle to the model, respectively.

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 as variables, containing one or more data sets.

### 6.1.2 Input and output format for external functions

There are several possible use cases:

#### 6.1.2.1 Null return

When using a function that generates a table, plots a figure, or some other stand alone operation, where the function does not return data to the main UI, the function should have a single output variable. The output variable can be assigned a text string, or 'no output', if no user message is required, e.g.:

```
function res = phaseplot(x,y,t,labels)  
...  
res = {'Plot completed'}; %or res = {'no output'}; for silent mode  
...  
end
```

### 6.1.2.2 Single value output

For a function that may in some instances return a single value this should be the first variable being returned and can be numeric or text, e.g.:

```
function [qtime,qdrift] = littoraldriftstats(qs,tdt,varargin)
...
%Case 1 - return time and drift
qdrift = array1;
qtime = array2;
%Case 2 - return summary value
qtime = mean(array2); %return single value
%Case 3 - return summary text
qtime = sprintf('Mean drift = %.1f',mean(array2)); %return test string
...
end
```

### 6.1.2.3 Using variables

If only one variable is returned (length>1), or the first variable is empty and is followed by one or more variables, the user is prompted add the variables to:

- i) Input Cases – one of the datasets used in the function call;
- ii) New Case – use output to define a new dataset;
- iii) Existing Case – add the output to an existing dataset (data sets for the selected existing case and the data being added must have the same number of rows.

In each case the user is prompted to define the properties for each of the variables.

**Note** that variable names and descriptions must be unique within any one dataset.

```
function y = moving(x,m,fun)
%a single variable is returned with no rows
y is a vector or array
...
end
```

or

```
function [x,y,z] = afunction(x,m,fun)
%multiple variables returned but the first variable is empty
x = [];
y and z are a vectors or arrays
...
end
```

When the first variable defines the rows of a table and subsequent variables the table entries, all variables must be the same length for the first dimension. This is treated as a new Case and the user is prompted to define the properties for each of the variables.

```
function [trange,range,hwl,lwl] = tidalrange(wl,t,issave,isplot)
%first variable is row dimension followed by additional variables
trange,range,hwl,lwl are vectors or arrays
...
end
```



#### 6.1.2.4 Using dstables

When the output has multiple variables of a defined type it can be more convenient to define the dsproperties within the function and return the data in a dstable. This avoids the need for the user to manually input the meta-data properties. In addition, if the function generates multiple dstables, these can be returned as a struct, where the struct fieldnames define the Dataset name.

```
function dst = tidalrange(wl,t,issave,isplot)
    %dst is a dstable with variables, dimensions and dsproprties assigned
    %as required, or a struct of dstables with the struct fieldnames defining
    %each Dataset.
    dst = ...

    ...

end
```

Similarly if the input is also using dstables, the syntax is as follows:

```
function dst_out = myfunction3(dst_in,'usertext',mobj)
    %dst_in is one or more input dstables, 'usertext' is some additional
    %instruction to the function and mobj is a handle to the model
    %allowing access to other datasets. dst_out is either a dstable, or a
    %struct of dstables with the struct fieldnames defining each Dataset.
    dst = ...

    ...

end
```

### 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 muitoolbox. 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.

### 6.1.3 Pre-defined functions

The following examples are provided within ChannelForm, 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. **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:

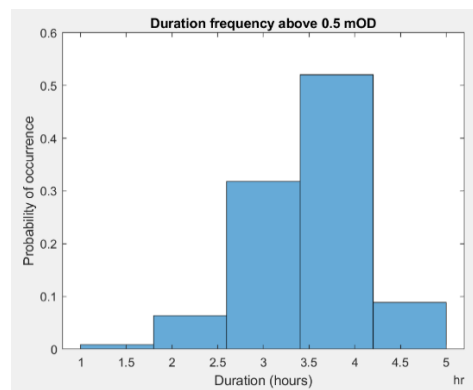
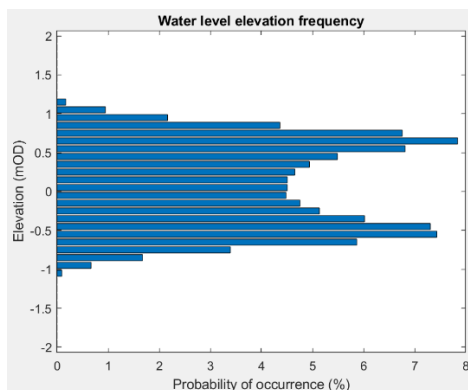
`recursive_plot(x, 'varname', nint)`, where `x` is the variable, '`varname`' is a text string in single quotes and `nint` is an integer value that defines the size of the offset.

3. **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).

4. **Plot a vertical histogram of the frequency of occurrence and a conventional histogram of the duration frequency** using the function:

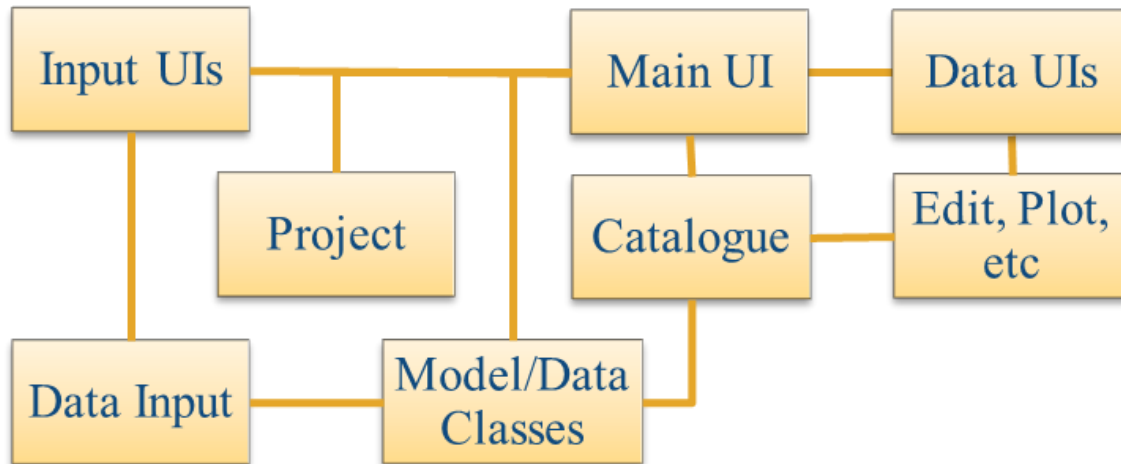
`vertical_histogram(X, t)` where `X` is the variable and `t` is time. Plot options include Water level elevation frequency, Water level spectrum, Elevations above a threshold, Duration of threshold exceedance, Elevation frequency above threshold. Designed to analyse water levels but could easily be adapted for other variables.



## 7 Program Structure

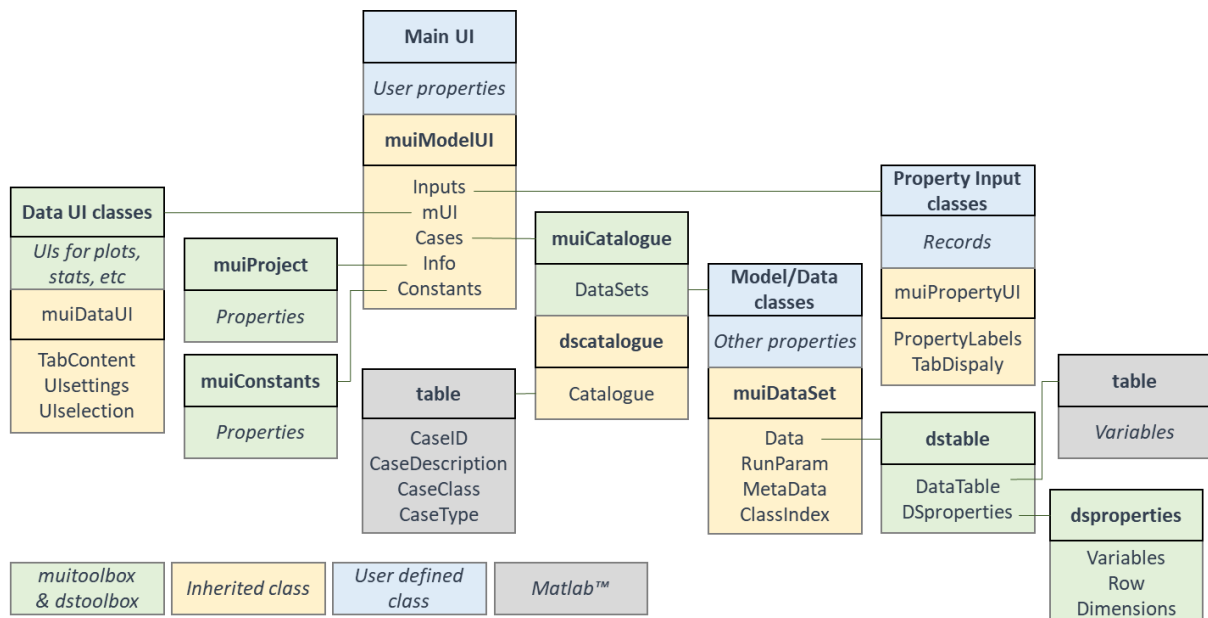
The overall structure of the code is illustrated schematically in Figure 15. This is implemented through several classes that handle the graphical user interface and program workflows (Main UI) and several classes that handle the data manipulation and plotting (Input UIs and Data UIs).

Figure 15 – High level schematic of program structure



The interfaces and default functionality are implemented in the ChannelForm App using the following mui toolbox classes depicted in Figure 16, which shows a more detailed schematic of the program structure. See the mui toolbox and dstoolbox documentation for more details.

Figure 16 – schematic of program structure showing how the main classes from mui toolbox and dstoolbox are used



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

### Model classes

**ChannelForm** - main UI for the ChannelForm interface, which implements the *muiModelUI* abstract class to define main menus.

- CF\_ExpData** - handles data input for exponential form parameters.
- CF\_PowerData** - handles data input for power form parameters.
- CF\_ValleyData** - handles data input for valley form parameters.
- CF\_HydroData** - handles data input of the hydraulic parameters used by CSTmodel App.
- CF\_SediData** - handles data input for sediment flux parameters.
- CF\_ShoreData** – handles data input for shoreline form parameters.
- CF\_ModsData** - handles data input for imposed modifications to the form.
- CF\_TransData** - handles data input for channel transgression parameter settings.
- CF\_FormModel** - methods needed to create and manipulate exponential, power, or CKFA form models.
- CF\_ValleyModel** - methods needed to create and manipulate an idealised valley form.
- CF\_TransModel** - transgression model to compute the movement of a channel within a valley in response to sea level rise using a simple kinetic model.
- RunProperties** - define the model run time parameters.
- WaterLevels** - setup and access to water level definitions, tidal parameters and changes over time.

### Grid classes

- GD\_GridProps** - class inherits *muiPropertyUI* abstract class, providing an interface to define the extent and intervals of a cartesian grid.
- GDInterface** - an abstract class to support classes that need additional functionality to handle grids. The class inherits *muiDataSet* abstract class and together they provide an extensive set of methods to handle datasets of various types (eg from models or imported files).
- GD\_ImportData** - class inherits *GDInterface* abstract class (see above) to load xyz data from a file.

### Model functions

Functions that implement various form models:

- cf\_changeplot* - plot volume difference over the model run and the variation of sediment exchange with transgression distance.
- cf\_cst2grid* - interpolate the CST water levels onto the model grid.
- cf\_exp\_models* - construct idealised channel form using exponential functions in y to determine width and CKFA cross-section to determine z at each x interval.
- cf\_model\_tabs* - generate tabPlot and tabProperties for any GDInterface model.
- cf\_offset\_wls* - translate water levels landwards when mouth is some distance from x=0 and pad the water level vectors with values at the mouth.
- cf\_pow\_model* - function to compute 3D form of a creek or tidal channel using power laws to define width and hydraulic depth variations.
- cf\_property\_plots* - plots displayed on Property tab in ChannelForm model.
- cf\_sectionplot* - plot a set of along-channel cross-sections and the channel centre-line (thalweg) for the initial and final grid.
- cf\_set\_hydroprops* - set water levels for form model using either the surface defined by the

*cst\_model* or high water at the mouth and a reducing tidal amplitude.

*cf\_summarygridplot* - plot initial grid, new grid and difference.

*cf\_summarytransplot* - composite set of plots to show results from Transgression table.

*cf\_valley\_model* - construct idealised channel form using 3D CKFA model.

*cf\_writetable* - write GrossProps and Transgression output tables from Transgression model to a file.

*ckfa\_3D\_form* - construct 3D form of an idealised channel based on the CKFA model.

*ckfa\_dimensions* - utility to get the CKFA channel form properties: hydraulic depth, hm, width convergence length, Lw, critical tidal velocity amplitude, Ucr.

*ckfa\_form\_model* - construct idealised channel form using 3D CKFA model

*ckfa\_form\_solver* - find the hydraulic depth, e-folding length for width and the peak tidal amplitude for defined tidal range and sediment properties.

*ckfa\_wave\_form* - obtain depth and width of wave formed profile at high and low water

*ckfa\_wave\_profile* - depth and width of wave formed profile  $d=dw*(1-y/yw)^{2/3}$ .

*get\_river\_profile* - get river profile using regime theoretical profile (Cao & Knight, 1996).

*get\_river\_regime* - get river regime properties using regime theoretical profile (Cao & Knight, 1996).

*get\_sed\_flux* - single element ASMITA model to compute amount of sedimentation.

## Grid functions

Functions used to manipulate cartesian grids can be found in the *muiAppGridFcns* folder and includes the following:

*gd\_ax\_dir* - check direction of grid axes and reverse if descending, OR find grid orientation using ishead and direction of x-axis, OR check a grid axis direction by prompting user.

*gd\_basin\_hypsometry* - compute area and volume hypsometry from gridded elevation data.

*gd\_basin\_indices* - get the indices of the grid x-axis that fall within the basin or channel, when the mouth is offset from the grid origin. (NB: assumes basin/channel is aligned with the x-axis). Also returns the index of mouth position on the x-axis.

*gd\_basin\_properties* - use the basin hypsometry from *gd\_basin\_hypsometry* to compute several along-channel/x-axis morphological properties.

*gd\_colormap* - check if Mapping toolbox is installed to use land/sea colormap, or call *cmap\_selection*. the *cmap\_selection* function in the *mutoolbox* if not available.

*gd\_digitisepoints* - creates figure to interactively digitise points on a grid and add elevations if required.

*gd\_dimensions* - get the grid dimensions for a grid struct (as used in GDinterface).

*gd\_grid\_line* - create a grid from scattered data points input as xyz tuples.

*gd\_gross\_properties* - compute the gross properties of a gridded bathymetry.

*gd\_plan\_form* - compute planform variation along the x-axis at specified planar levels.

*gd\_plotgrid* - create pcolor plot of gridded surface.

*gd\_property\_plots* - plots displayed on Property tab or stand-alone figure in Apps that use GDinterface, such as ChannelForm and ModelSkill.

*gd\_section\_properties* - compute the width, cross-sectional area and prism along channel.

*getconvergelength* - least squares fit using `fminsearch` to find the convergence length of a channel from a distance-width (xy) data set.

*gd\_selectpoints* - accept figure to interactively select one or more points on a grid.

*gd\_setpoint* - interactively select a point on a plot and return the point coordinates. Includes an option to enter an additional value at the selected point (e.g. for elevation).

*gd\_startendpoints* - accept figure to interactively select start and end points on a grid.

*gd\_subdomain* - figure to interactively select a subdomain of a grid.

*gd\_property\_plots* - plots displayed on Property tab in ChannelForm model and figure in ModelSkill.

*gd\_xy2sn* - map grid from cartesian to curvilinear coordinates with option to return the elevations on the source cartesian grid, or as a curvilinear grid.

*gd\_sn2xy* - map grid from curvilinear to cartesian coordinates.

*getconvergelength* - least squares fit using `fminsearch` to find the convergence length of a channel from a distance-width xy data set.

*getsubgrid* - extract a subdomain from a grid and return the extracted grid and the source grid indices of the bounding rectangle.

*a\_star* - implements the A\* search algorithm to find the shortest path given constraints (inaccessible cells) and a cost function (e.g. water depths). Author: Alex Ranaldi, 2022,

[https://github.com/alexranaldi/A\\_STAR](https://github.com/alexranaldi/A_STAR).

*InterX* - intersection of two curves. MATLAB Central File Exchange, Author: NS, 2010,

<https://www.mathworks.com/matlabcentral/fileexchange/22441-curve-intersections>.

*xy2sn* and *sn2xy* - Cartesian to Curvilinear coordinate forward and backward transformation.

MATLAB Central File Exchange, Author: Bart Vermeulen, 2022,

<https://www.mathworks.com/matlabcentral/fileexchange/55039-cartesian-to-curvilinear-coordinate-forward-and-backward-transformation>.<sup>12</sup>

Further details can be found by using the `>>help <function name>` command in the Command Window.

### Generic functions

Other functions used in ChannelForm, include:

*river\_regime* - function to compute the width and hydraulic depth of a river section given the discharge and energy slope.

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

*settling\_velocity* - calculate the settling velocity using Soulsby equation.

*ucrit* - compute the critical flow velocity for a given critical shear stress and wave conditions in the wave-current case.

---

<sup>12</sup> The functions by Dugge are equivalent with the order of centreline and co-ordinates reversed in the calls from *gd\_xy2sn* and *gd\_sn2xy*. See Juernjakob Dugge, 2015, [jdugge/xy2sn](https://github.com/jdugge/xy2sn), <https://github.com/jdugge/xy2sn>, which requires *arclength* - John D'Errico, 2012, <https://www.mathworks.com/matlabcentral/fileexchange/34871-arclength>. *distance2curve* - John D'Errico, 2013, <http://www.mathworks.de/matlabcentral/fileexchange/34869-distance2curve>. *interparc* - John D'Errico, 2012, <https://www.mathworks.com/matlabcentral/fileexchange/34874-interparc>.

## 8 Bibliography

- Boon III, J.D., Byrne, R.J., 1981. On basin hypsometry and the morphodynamic response of coastal inlet systems. *Marine Geology*, 40, 27-48.
- Bouws, E., Günther, H., Rosenthal, W., Vincent, C.L., 1985. Similarity of the wind wave spectra in finite depth water: Part 1 - spectral form. *Journal of Geophysical Research*, 90(C1), 975-986.
- Bruun, P., 1962. Sea level rise as a cause of shore erosion. *Journal of Waterways and Harbours Division, ASCE*, 88, 117-130.
- Cai, H., 2014. A new analytical framework for tidal propagation in estuaries. PhD, TU Delft, TU Delft, 187 pp.
- Cai, H., Savenije, H.H.G., Jiang, C., Zhao, L., Yang, Q., 2016. Analytical approach for determining the mean water level profile in an estuary with substantial fresh water discharge. *Hydrology and Earth System Sciences*, 20(3), 1177-1195. [10.5194/hess-20-1177-2016]
- Cao, S., Knight, D., 1996. Regime theory of alluvial channels based upon concepts of stream power and probability. *Water, Maritime & Energy*, 118(3), 160-167.
- Cao, S., Knight, D., 1997. Entropy-based design approach of threshold alluvial channels. *Journal of Hydraulic Research*, 35(4), 505-524.
- Cearreta, A., Monge-Ganuzas, M., 2013. Evolución paleoambiental del estuario del Oka (Reserva de la Biosfera de Urdaibai, Vizcaya): respuesta al ascenso del nivel marino durante el Holoceno. *Jornadas de Geomorfología Litoral*, 163-166.
- Coles, S., 2001. An Introduction to Statistical Modeling of Extreme Values. Springer Series in Statistics. Springer-Verlag, London.
- Dal Monte, L., Di Silvio, G., 2004. Sediment concentration in tidal lagoons. A contribution to long-term morphological modelling. *Journal of Marine Systems*, 51(1-4), 243-255.
- Dean, R.G., 1977. Equilibrium beach profiles: US Atlantic and Gulf coasts. Ocean Engineering Report No 12, Newark, Delaware.
- Dean, R.G., 1991. Equilibrium beach profiles: characteristics and applications. *Journal of Coastal Research*, 7(1), 53-84.
- Di Silvio, G., 1999. Interaction between marshes, channels and shoals in tidal lagoons, River, coastal and estuarine morphodynamics. IAHR Symposium Proceedings, Sept 1999. University of Genova, Department of Environmental Engineering, 1, pp. 695-704.
- Donelan, M.A., Hamilton, J., Hui, W.H., 1985. Directional Spectra of wind-generated waves. *Phil.Trans.R.Soc.Lond.A.*, 315(1534), 509-562.
- Dronkers, J., 2016. Dynamics of coastal systems. Advanced Series in Ocean Engineering, 41. World Scientific, Singapore.
- Evangelisti, G., 1955. On tidal waves in a canal with variable cross-section. Proc 6th General Meeting. International Association of Hydraulic Research, pp. A10-11-A10-12.
- Friedrichs, C.T., 1995. Stability shear stress and equilibrium cross-sectional geometry of sheltered tidal channels. *Journal of Coastal Research*, 11(4), 1062-1074.

- Friedrichs, C.T., Aubrey, D.G., 1994. Tidal propagation in strongly convergent channels. *Journal of Geophysical Research*, 99(C2), 3321-3336.
- Friedrichs, C.T., Aubrey, D.G., 1996. Uniform bottom shear stress and equilibrium hypsometry of intertidal flats. In: C. Pattiaratchi (Ed.), *Mixing in estuaries and coastal seas*. Coastal and Estuarine Studies. American Geophysical Union, Washington, pp. 405-429.
- Gao, S., 2007. Modeling the growth limit of the Changjiang Delta. *Geomorphology*, 85(3-4), 225-236. [10.1016/j.geomorph.2006.03.021]
- 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>]
- Hughes, S.A., 1984. The TMA shallow-water spectrum description and applications. CERC-84-7, Washington DC.
- Hughes, S.A., 2002. Equilibrium cross sectional area at tidal inlets. *Journal of Coastal Research*, 18(1), 160-174.
- Hunt, J.N., 1964. Tidal oscillations in estuaries. *Geophys.J.R.Ast.Soc.*, 8, 440-455.
- Jay, D.A., 1991. Tidal long-wave propagation in channels with strong topography. *Journal of Geophysical Research*, 96(20), 585-598.
- Lamb, H., 1932. *Hydrodynamics*. Cambridge University Press.
- Lee, S.-C., Mehta, A.J., 1994. Equilibrium hypsometry of fine-grained shore profiles. INTERCOH '94, 4th Nearshore & Estuarine Cohesive Sediment, Transport Conf. HR Wallingford, pp. 1-10.
- Lee, S.-C., Mehta, A.J., 1997. Problems in characterizing dynamics of mud shore profiles. *Journal of Hydraulic Engineering, ASCE*, 123(4), 351-361.
- Leopold, L.B., Langbein, W.B., 1962. The concept of entropy in landscape evolution. Professional Paper 500-A, United States Government Printing Office, Washington.
- Partheniades, E., 1965. Erosion and deposition of cohesive soils. *Proceedings of the American Society of Civil Engineers*, 91(HY1), 105-139.
- Pillsbury, G.B., 1956. *Tidal Hydraulics*. Corps of Engineers, U S Army, Vicksburg, Mississippi.
- Prandle, D., Rahman, M., 1980. Tidal response in estuaries. *Journal of Physical Oceanography*, 10(10), 1552-1573.
- Rees, J.G., 2006. Sea-level, topographical and sediment supply controls on Holocene sediment composition in the Humber Estuary. *Phil.Trans.R.Soc.Lond.A.*, 364, 993-1008. [<https://doi.org/10.1098/rsta.2006.1750>]
- Roberts, W., Le Hir, P., Whitehouse, R.J.S., 2000. Investigation using simple mathematical models of the effect of tidal currents and waves on the profile shape of intertidal mudflats. *Continental Shelf Research*, 20(10-11), 1079-1097.

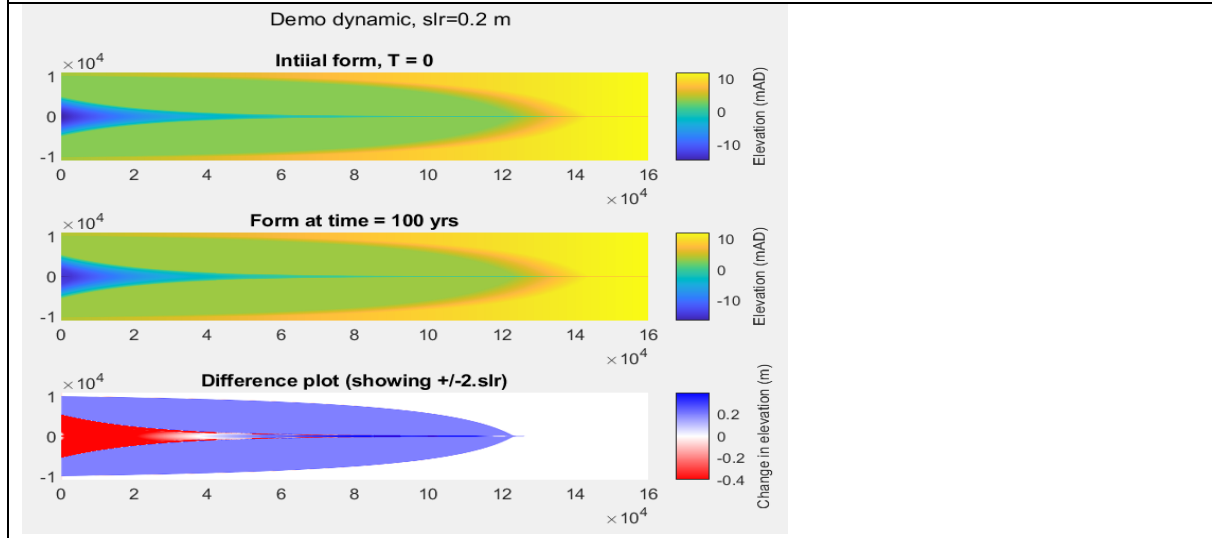


- Savenije, H.H.G., 2012. Salinity and tides in alluvial estuaries. Elsevier, Amsterdam, Netherlands.
- Soulsby, R., Clarke, S., 2005. Bed shear stresses under combined waves and currents on smooth and rough beds. TR137, Wallingford.
- Stive, M.J.F., Wang, Z.B., 2003. Morphodynamic modelling of tidal basins and coastal inlets. In: C. Lakkhan (Ed.), Advances in coastal modelling. Elsevier Sciences, Amsterdam, pp. 367-392.
- 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., 2008. Hypsometry of estuaries, creeks and breached sea wall sites. Proc.Instrn Civ.Engrs., Maritime Engineering, 161(MA1), 23-32.
- Townend, I.H., 2010. An exploration of equilibrium in Venice Lagoon using an idealised form model. Continental Shelf Research, 30(8), 984-999. [<https://doi.org/10.1016/j.csr.2009.10.012>]
- Townend, I.H., 2012. The estimation of estuary dimensions using a simplified form model and the exogenous controls. Earth Surface Processes and Landforms, 37, 1573-1583. [<http://dx.doi.org/10.1002/esp.3256>]
- Townend, I.H., Pethick, J., 2002. Estuarine flooding and managed retreat. Phil.Trans.R.Soc.Lond.A, 360(1796), 1477-1495. [<https://doi.org/10.1098/rsta.2002.1011>]
- 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., 2016. 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., Zhou, Z., Guo, L., Coco, G., 2020. A morphological investigation of marine transgression in estuaries. Earth Surf. Process. Landf., 46, 626–641. [<https://doi.org/10.1002/esp.5050>]
- van Veen, J., van der Spek, A.J.H.M., Stive, M.J.F., Zitman, T.J., 2005. Ebb and flood channel systems in the Netherlands tidal waters. Journal of Coastal Research, 21(6), 1107-1120.
- Wang, Z.B., Jeuken, M.C.J.L., Gerritsen, H., de Vriend, H.J., Kornman, B.A., 2002. Morphology and asymmetry of the vertical tide in the Westerschelde estuary. Continental Shelf Research, 22, 2599-2609.
- Whitehouse, R.J.S., Soulsby, R.L., Roberts, W., Mitchener, H., 2000. Dynamics of estuarine muds. Thomas Telford, London.
- Winterwerp, J.C., 2007. On the sedimentation rate of cohesive sediment. In: P.-Y. Maa, L.P. Sanford, D.H. Schoellhamer (Eds.), Estuarine and Coastal Fine Sediment Dynamics: INTERCOH 2003. Proceedings in Marine Science 8. Elsevier, Amsterdam, pp. 209-226.
- Woolnough, S.J., Allen, J.R.L., Wood, W.L., 1995. An exploratory numerical model of sediment deposition over tidal salt marshes. Estuarine, Coastal and Shelf Science, 41, 515-543.

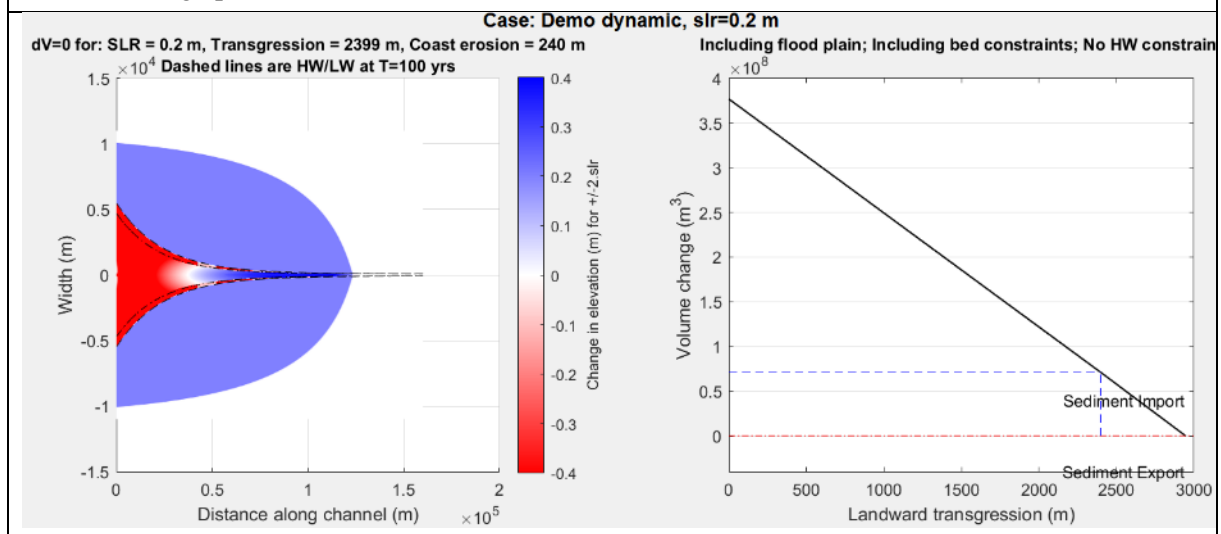


## Appendix 1 – Sample Output for demo.mat file

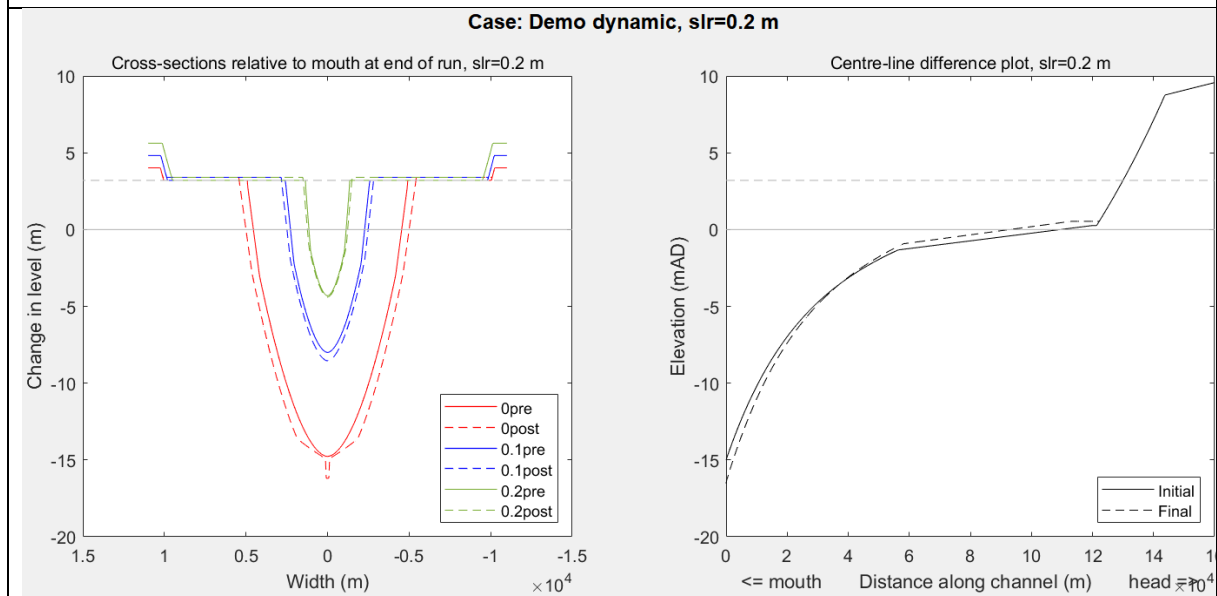
(a) Grid plot



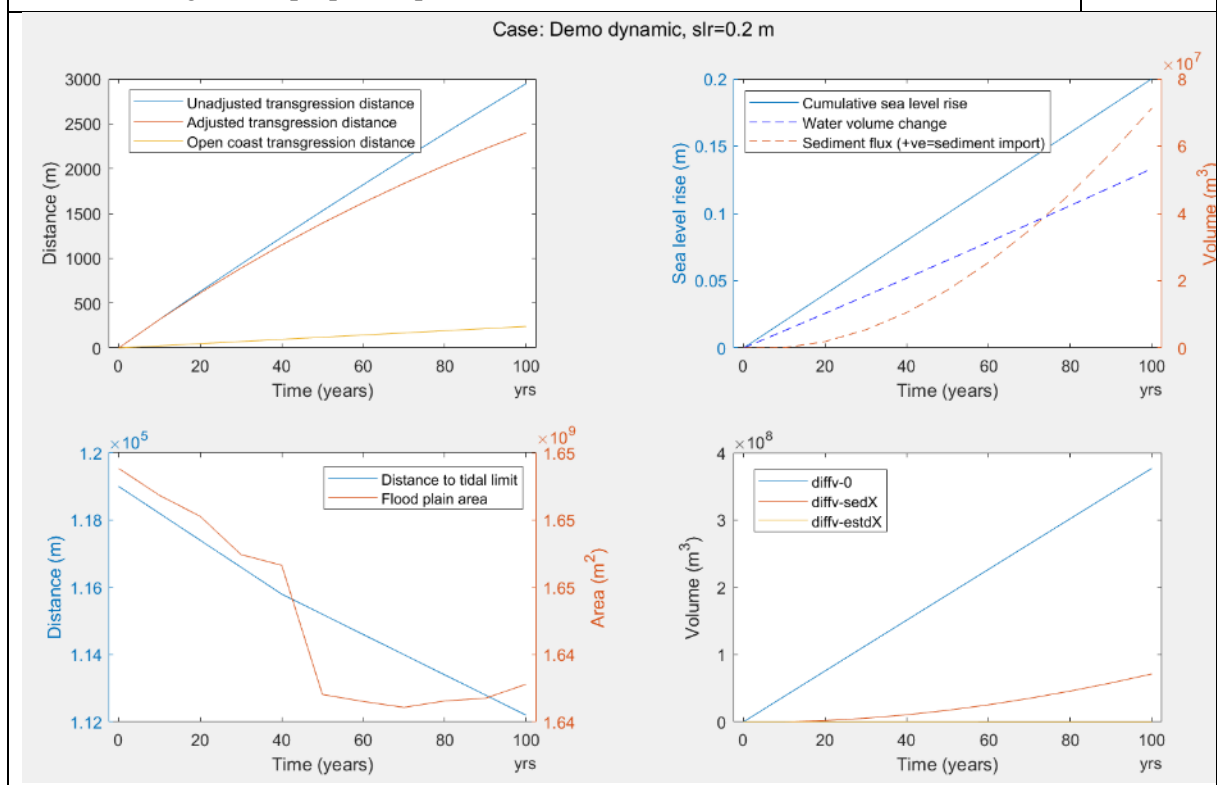
(b) Change plot



## (c) Sections plot



## (d) Transgression properties plot

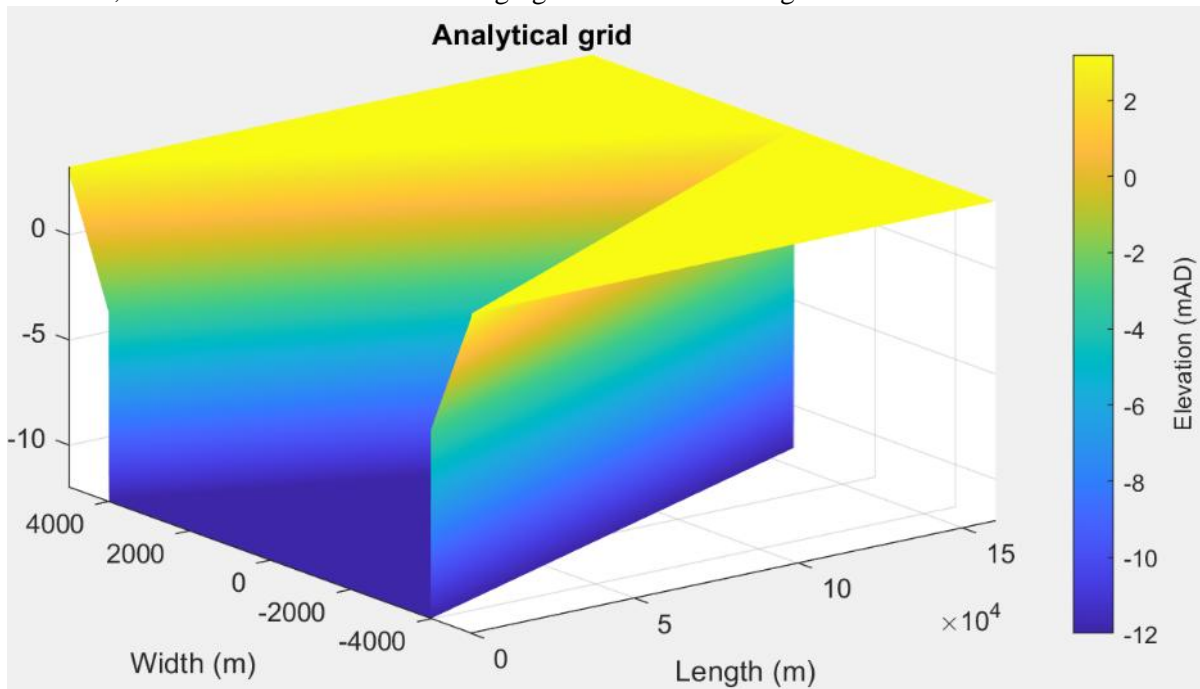


## Appendix 2 – Grid model property error estimates

The functions used to derive hypsometry depend on the grid resolution and some along channel properties derived from cross-sections such as width and area also depend on the alignment of the channel relative to the x-axis. To test the former a simple channel was defined with the following properties (all in metres):

Length	HW Width	LW Width	Tidal amplitude	Mean tide level	Depth to mtl
160,000	10,000	8,000	3.0	0 or 0.065	12.0

These are used to generate a rectangular channel section of constant depth and a linearly sloping intertidal, with the channel widths converging to zero over the length of the channel.



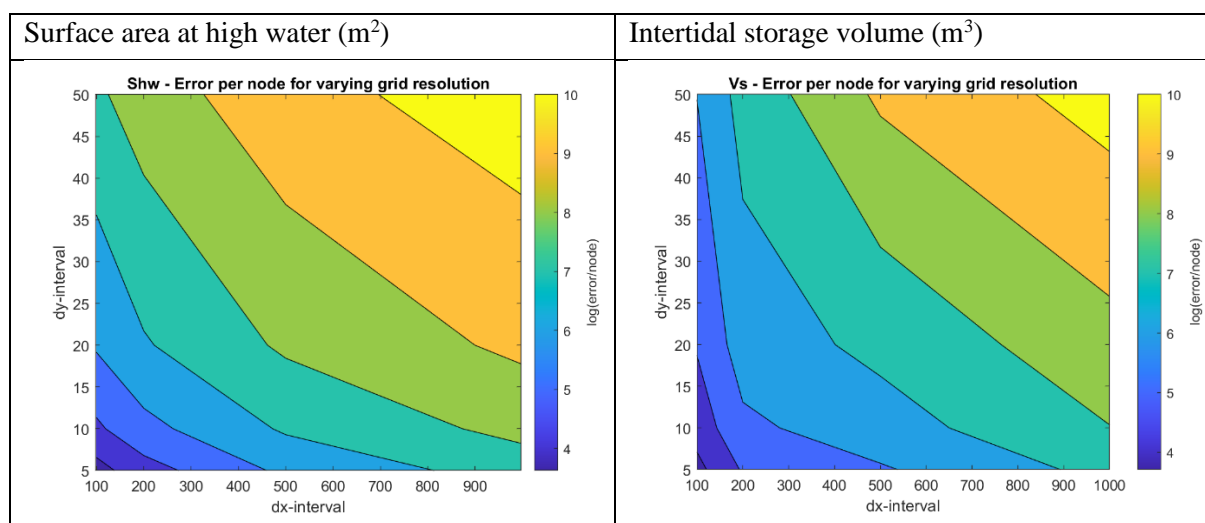
Grids were created with varying grid resolutions based on combinations of:

delX = 100, 200, 500, 1000 m.

delY = 5, 10, 20, 50 m.

The grids were loaded into ChannelForm and the channel properties computed. These runs were for a mean tide level = 0.

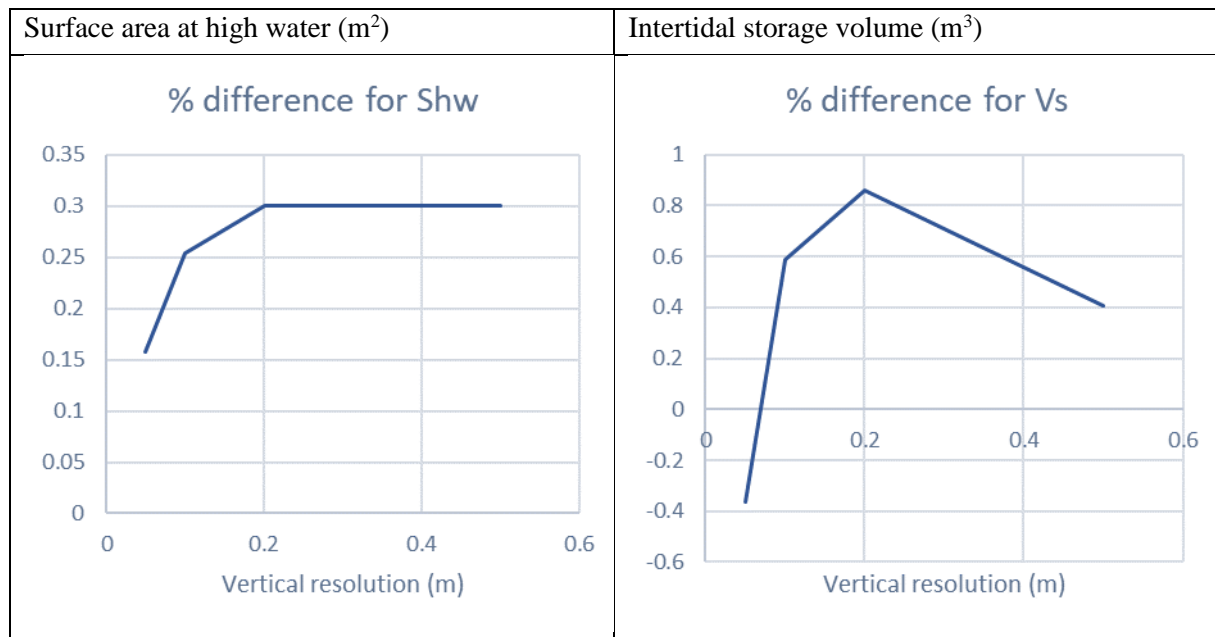
The plots below show the log of the error (difference from the analytical result) per node in the grid. The percentage error varies from 0.15-1.5% for Shw and from 0.27-2.2% for Vs. Whilst the percentage errors are generally small because the absolute values are large ( $O[10^6-10^{10}]$ ) the magnitude of the error can be large. For Surface area, greater resolution in x and y both improve the estimate, whereas for Storage volume below an x-resolution of ~400 the y-resolution begins to have less and less effect.



Summary of range of percentage error for each of the gross property variables (see Section 3.11.5).

Variable	Range of % error for sampled grid resolutions
Shw - High Water Surface Area (m <sup>2</sup> )	0.2-1.5
Slw - Low Water Surface Area (m <sup>2</sup> )	0.4-1.4
Vhw - High Water Volume (m <sup>3</sup> )	0.2-1.5
Vlw - Low Water Volume (m <sup>3</sup> )	0.1-1.3
PrA - Tidal Prism using cross-sectional areas (m <sup>3</sup> )	0.2-1.5
PrV - Tidal Prism using hypsometry volumes (m <sup>3</sup> )	0.4-1.9
Gamma - Dronkers' Gamma (-)	0.8-1.7
Vs - Storage Volume (m <sup>3</sup> )	0.3-2.2
Vc - Channel Volume (m <sup>3</sup> )	0.2-1.4
Am - Mean Tide Cross-sectional Area at mouth (m <sup>2</sup> )	0.0-0.5
hyd - Hydraulic depth of channel (m)	1.7-1.9
aoh - Amplitude to Depth ratio (-)	1.7-2.0
PrvAm - Prism to Cross-sectional Area ratio at mouth (m)	0.4-2.4

The default resolution in the vertical is 0.1 m. By varying this within the ChannelForm App the influence of vertical resolution was also examined. However with mtl=0, the vertical resolution does not make much difference because the intertidal is a plain slope and the vertical grid intervals align with the start and end of the slope (+3 and -3). This case therefore used mtl = 0.065m. Again the percentage differences are small in both cases (<1%).





## Appendix 3 – Channel form solver equations

**Drag coefficient:**

$$c_{dr} = \left[ \frac{0.4}{\ln(h/z_o) - 1} \right]^2; \quad c_{ds} = 1.615 \cdot 10^{-4} \cdot \exp[6 \cdot \text{Re}^{-0.08}];$$

$$\text{Re} = U \cdot h / \nu; \quad u_{cr} < u_{cs} \rightarrow C_d = c_{dr}, \quad \text{else} \quad C_d = c_{ds}$$

**L<sub>w</sub>– width convergence length:**

$$\text{if } \varepsilon_H^2 + \varepsilon_{L_w}^2 - 1 > 0 \rightarrow$$

$$L_w = \frac{1}{k} \tan^{-1} \left[ \begin{array}{l} \left( \frac{\varepsilon_{L_w} \left( 1 + \sqrt{\varepsilon_H^2 \cdot \left( \varepsilon_{L_w}^2 + \varepsilon_H^2 - 1 \right)} \right)}{\varepsilon_{L_w}^2 + \varepsilon_H^2} \right)^{-1} - \frac{\varepsilon_{L_w} + \sqrt{\varepsilon_H^2 \cdot \left( \varepsilon_{L_w}^2 + \varepsilon_H^2 - 1 \right)}}{\varepsilon_{L_w}^2 + \varepsilon_H^2} \\ \left( \frac{\varepsilon_{L_w} \left( -1 + \sqrt{\varepsilon_H^2 \cdot \left( \varepsilon_{L_w}^2 + \varepsilon_H^2 - 1 \right)} \right)}{\varepsilon_{L_w}^2 + \varepsilon_H^2} \right)^{-1} - \frac{-\varepsilon_{L_w} + \sqrt{\varepsilon_H^2 \cdot \left( \varepsilon_{L_w}^2 + \varepsilon_H^2 - 1 \right)}}{\varepsilon_{L_w}^2 + \varepsilon_H^2} \end{array} \right] \Bigg\}^{\text{max}}$$

$$\text{else } L_w = \frac{1}{k} \tan^{-1} \left( \frac{2\varepsilon_H}{1 + \varepsilon_H^2}, \frac{1 - \varepsilon_H^2}{1 + \varepsilon_H^2} \right);$$

$$k = \frac{\omega}{\sqrt{gh_m}}; \quad \varepsilon_H = \frac{a\pi\beta}{4h_m}; \quad \varepsilon_{L_w} = \left( 1 - \exp\left(\frac{-L_e}{L_w}\right) \right) \quad \text{and}$$

$$\beta = \frac{W_s}{W_m} \approx \left( \frac{d-a}{d} \right)^{r-1} = \kappa \left( \frac{rh_m - a}{rh_m} \right)^{r-1} \quad \kappa \approx 1$$

**L<sub>A</sub> – area convergence length:**

$$L_w = L_A \left( \frac{1 - \varepsilon_H \cdot \sin(k \cdot L_A)}{\cos(k \cdot L_A)} \right) \quad \text{and} \quad 1 - \varepsilon_H \cdot \sin(k \cdot L_A) \approx 1$$

$$L_A = L_w \cos(\varepsilon_A); \quad \varepsilon_A = k \cdot L_A \quad \text{solve iteratively, given } L_w$$

**Tidal velocity amplitude:**  $U = \frac{a \cdot \omega \cdot L_A}{h_m}$

**Bed Shear stress:**  $\tau_a = \rho \cdot C_d \cdot U^2$

**Erosion-deposition balance:**  $\frac{m_e}{\pi} \left\{ (\tau_a - 2\tau_{cr}) \cdot \left( \frac{\pi}{2} - \sin^{-1} \left( \sqrt{\frac{\tau_{cr}}{\tau_a}} \right) \right) + \sqrt{\tau_{cr} (\tau_a - \tau_{cr})} \right\} - w_s \cdot \rho_c = 0$

When solving for depth, the convergence length is adjusted to account for the influence of waves, as follows:

Fetch is given by  $F = \sqrt{2} (\alpha W_m)$  where  $\alpha$  is a coefficient, e.g.,  $\alpha=0.6$  is the width at  $L_w/2$ .

Wave height and period are computed from the wind speed and fetch over a depth of  $h_m$ .

The area convergence is adjusted to take account of waves at the mouth. The change in width is computed due to the formation of a wave profile at mean tide and low water, to give the change in width and CSA at mean tide.

Uses the shear stress due to waves:  $\tau_w = \rho \cdot C_{dw} \cdot u_w^2$ , where  $u_w = \frac{1}{2} \sqrt{\frac{g}{d_e}} H$  and balances erosion and

deposition to find  $d_e$ :  $\frac{m_e}{\pi} \left\{ (\tau_w - 2\tau_{cr}) \cdot \left( \frac{\pi}{2} - \sin^{-1} \left( \sqrt{\frac{\tau_{cr}}{\tau_w}} \right) \right) + \sqrt{\tau_{cr} (\tau_w - \tau_{cr})} \right\} - w_s \cdot \rho_c = 0$ .

The change in width is then given by:  $\frac{dW}{2} = \frac{3\pi}{4C_{dw}} \frac{d_e^2}{H}$  and the adjusted convergence length is

obtained from:  $L'_w = -L_e / \log \left( \frac{W_r}{W_m + dW} \right)$ ;  $L'_A = -L_e / \log \left( \frac{A_r}{A_m + dA} \right)$ .

The tidal current amplitude is updated with the new value of  $L_A$  and the shear stress at the bed is obtained using the function tau\_bed.m for the bed shear stress under combined wave-current action (Soulsby and Clarke, 2005). This allows the erosion-deposition balance to be obtained using the combined effect of waves and tides:

$$\frac{m_e}{\pi} \left\{ (\tau_{wc} - 2\tau_{cr}) \cdot \left( \frac{\pi}{2} - \sin^{-1} \left( \sqrt{\frac{\tau_{cr}}{\tau_{wc}}} \right) \right) + \sqrt{\tau_{cr} (\tau_{wc} - \tau_{cr})} \right\} - w_s \cdot \rho_c = 0$$

Tidal prism is then given by:

$$P = 2a \cdot \cos(\varphi) \cdot W_m L_w \varepsilon_{L_w}; \quad \varphi = \varepsilon_A = k \cdot L_A; \quad \varepsilon_{L_w} = \left( 1 - \exp \left( \frac{-L_e}{L_w} \right) \right); \quad k = \frac{\omega C_r L_A}{g h_m}; \quad C_r = \frac{8}{3\pi} \frac{C_d U}{h_s}$$

or

$$P = \int_{\frac{\varepsilon_A}{\omega}}^{\frac{\varepsilon_A + \pi}{\omega}} U \sin(\omega t - \varepsilon_A) \cdot A_m \left( 1 + \frac{a \cdot \cos(\omega t)}{h_s} \right) dt = \frac{U A_m}{2\omega} (4 - \pi \varepsilon_h \cdot \sin(\varepsilon_A))$$