

TUFLOW FV User Manual

Build 2020.01

Sediment Transport and Particle Tracking Modules

[Sediment Transport
Commands](#)

[Particle Tracking
Commands](#)

www.tuflow.com

[TUFLOW FV Wiki](#)

[TUFLOW Tutorial Model](#)

support@tuflow.com

[How to Use This Manual](#)

[Chapters](#)

[Table of Contents](#)

[List of Figures](#)

[List of Tables](#)

[Appendices](#)

How to Use This Manual

This manual is designed for both hardcopy and digital usage. It is provided in both its native Microsoft Word 2013 version and as a pdf document. The pdf version is faster to navigate around and, for the first time, all links are preserved.

Section, Table and Figure references are [styled like this](#) and linked. In Word, to go to the link hold down the control (Ctrl) key and click on the Section, Table or Figure number in the text to move to the relevant page. In the pdf document, the links should be directly accessible.

Similarly, and most importantly, script or control file commands are hyperlinked and are easily accessed through the lists at the end of the manual. To quickly go to the end of the manual press Ctrl End. There are also command hyperlinks within the text. Command text can be copied and pasted into the text files to ensure correct spelling.

Web and email links are [styled like this](#). Much content now resides on the TUFLOW and TUFLOW FV Wiki with web links provided in this document to those Wiki pages. Other useful keys are Alt Left / Right arrow to link backwards / forwards to the last clicked locations in the document. Ctrl Home returns to the front page, which also contains useful links.

A secondary window can be opened in Word by selecting View->New Window or in Adobe Acrobat by selecting Window->New Window, allowing you to view different sections of the document in different windows. For example, the TUFLOW command lists could be viewed in one window and the section describing the functionality in another window.

Constructive suggestions are always welcome (please email support@tuflow.com).

About This Manual

This document is the User Manual for the TUFLOW FV Sediment Transport and Particle Tracking Modules, release 2020.01 It should be used in combination with the TUFLOW FV User Manual and Release Notes: <https://www.tuflow.com/FV%20Documentation.aspx>.

Chapters

1	Sediment Transport Module	1-1
2	Particle Tracking Module	2-1
3	References	3-1
Appendix A	fvshed File Commands	A-1
Appendix B	fvptm File Commands	B-1

Table of Contents

1	Sediment Transport Module	1-1
1.1	Sediment Transport Module Description	1-3
1.1.1	Introduction	1-3
1.2	Available Sediment Transport Models	1-6
1.2.1	Global Models	1-6
1.2.2	Sediment Fraction Models	1-6
1.2.3	External Models	1-7
1.3	Scientific Documentation	1-8
1.3.1	Concentration Profile	1-8
1.3.2	Bed Roughness	1-8
1.3.2.1	<i>Specified k_s</i>	1-8
1.3.2.2	<i>Proportional to d_{50}</i>	1-8
1.3.2.3	<i>van Rijn (2004)</i>	1-9
1.3.2.4	<i>Bed roughness coupling</i>	1-9
1.3.3	Bed Shear Stress	1-10
1.3.3.1	<i>Default</i>	1-10
1.3.3.2	<i>Bijker</i>	1-11
1.3.3.3	<i>Van Rijn (2004)</i>	1-11
1.3.4	Settling Model	1-12
1.3.4.1	<i>None</i>	1-12
1.3.4.2	<i>Constant w_s</i>	1-12
1.3.4.3	<i>Flocculation</i>	1-12
1.3.4.4	<i>Flocculation + hindered settling</i>	1-13
1.3.4.5	<i>van Rijn (1984)</i>	1-13
1.3.4.6	<i>van Rijn (2004)</i>	1-14
1.3.5	Deposition Model	1-15
1.3.5.1	<i>None</i>	1-15
1.3.5.2	<i>Unhindered</i>	1-15
1.3.5.3	<i>Krone</i>	1-15
1.3.6	Erosion Model	1-17
1.3.6.1	<i>None</i>	1-17
1.3.6.2	<i>Mehta</i>	1-17
1.3.6.3	<i>van Rijn (1984)</i>	1-17
1.3.6.4	<i>Soulsby-vanRijn</i>	1-18
1.3.6.5	<i>Bijker</i>	1-19
1.3.6.6	<i>van Rijn (2004)</i>	1-19
1.3.7	Bed Load Model	1-20

1.3.7.1	<i>None</i>	1-20
1.3.7.2	<i>Meyer-Peter and Müller</i>	1-20
1.3.7.3	<i>MPM-Shimizu</i>	1-21
1.3.7.4	<i>Soulsby - Van Rijn</i>	1-22
1.3.7.5	<i>Bijker</i>	1-23
1.3.7.6	<i>Wilcock-Crowe</i>	1-24
1.3.7.7	<i>van Rijn (2004)</i>	1-24
1.3.8	Critical Stress Model	1-25
1.3.8.1	<i>None</i>	1-25
1.3.8.2	<i>Constant</i>	1-25
1.3.8.3	<i>Soulsby</i>	1-25
1.3.8.4	<i>Soulsby-Egiazaroff</i>	1-25
1.3.9	Consolidation Model	1-27
1.3.10	External Model	1-28
1.3.10.1	<i>TR2004 Bed Forms and Bed Roughness</i>	1-29
1.3.10.2	<i>TR2004 Bed Shear Stress</i>	1-31
1.3.10.3	<i>TR2004 Critical Shear Stress</i>	1-32
1.3.10.4	<i>TR2004 Settling Velocity</i>	1-32
1.3.10.5	<i>TR2004 Erosion Model</i>	1-32
1.3.10.6	<i>TR2004 Bedload Model</i>	1-34
1.4	TUFLOW FV Control File (.FVC) for STM	1-35
1.4.1	Simulation Configuration	1-35
1.4.2	Materials	1-35
1.4.3	Boundary Conditions	1-35
1.4.4	Initial Conditions	1-35
1.4.5	Outputs	1-36
1.5	STM Control File	1-37
1.5.1	Introduction	1-37
1.6	Simulation Configuration	1-39
1.6.1.1	<i>Time commands</i>	1-39
1.6.1.2	<i>Morphological flags</i>	1-39
1.6.1.3	<i>Depth limit commands</i>	1-39
1.6.1.4	<i>Bed Armouring</i>	1-40
1.6.2	Bed restart files	1-41
1.6.2.1	<i>Bed Warmup</i>	1-41
1.7	Global Model Specifications	1-42
1.7.1.1	<i>Bed roughness model</i>	1-42
1.7.1.2	<i>Bed shear model</i>	1-42
1.7.1.3	<i>Bed slumping model</i>	1-43

1.7.1.4	<i>None</i>	1-43
1.7.1.5	<i>Simple (angle of repose)</i>	1-44
1.7.1.6	<i>“External” models</i>	1-44
1.8	Sediment Fraction (Group) Blocks	1-45
1.8.1	Sediment properties	1-45
1.8.2	Settling model	1-46
1.8.3	Erosion model	1-47
1.8.4	Deposition model	1-48
1.8.5	Bed load model	1-48
1.8.6	Critical stress model	1-49
1.8.7	Consolidation model	1-50
1.9	Material Blocks	1-51
1.9.1.1	<i>Reference height</i>	1-53
1.9.1.2	<i>Bed roughness</i>	1-53
1.9.1.3	<i>Sediment flux scaling</i>	1-53
1.9.2	Sediment fraction properties	1-53
1.9.3	Bed layer properties	1-53
1.9.4	Initial Sediment Mass Conditions	1-54
1.10	STM Outputs	1-55
1.10.1	Timeseries Output	1-55
1.10.2	Profile Output	1-55
1.10.3	Flux Output	1-55
1.10.4	Mass Output	1-55
1.10.5	Map Output Parameters	1-56
2	Particle Tracking Module	2-1
2.1	Introduction	2-2
2.2	Computational Framework	2-3
2.2.1	Overview	2-3
2.2.2	Lagrangian Scheme	2-4
2.2.3	Eulerian Scheme	2-5
2.2.4	Forcing Scheme	2-5
2.2.4.1	<i>Hydrodynamic Forcing HD Engine</i>	2-5
2.2.4.2	<i>Hydrodynamic Forcing Offline Mode</i>	2-6
2.2.4.3	<i>Wind Forcing</i>	2-6
2.2.4.4	<i>Wave Forcing</i>	2-6
2.3	Particle Behaviour Options	2-7
2.3.1	Particle Sediment Transport	2-9
2.3.2	Particle motility	2-10

2.4	TUFLOW FV Control File (.FVC) for PTM	2-11
2.4.1	Simulation Configuration	2-11
2.4.2	Offline Mode	2-11
2.4.3	Materials	2-11
2.4.4	Outputs	2-11
2.5	PTM Control File	2-12
2.5.1	Introduction	2-12
2.5.2	Simulation Configuration	2-14
	2.5.2.1 <i>NScalar</i>	2-14
	2.5.2.2 <i>Memory Allocation</i>	2-14
	2.5.2.3 <i>Timestep commands</i>	2-14
	2.5.2.4 <i>Sediment Transport commands (Global)</i>	2-14
	2.5.2.5 <i>Depth limit commands (Global)</i>	2-15
	2.5.2.6 <i>Restart File</i>	2-15
2.5.3	Particle Groups	2-15
	2.5.3.1 <i>Particle Motility</i>	2-18
2.5.4	Material Blocks	2-19
2.5.5	Particle Seeding	2-20
	2.5.5.1 <i>Particle Release</i>	2-20
	2.5.5.2 <i>Point source</i>	2-21
	2.5.5.3 <i>Polygon source</i>	2-21
	2.5.5.4 <i>Polyline source</i>	2-21
	2.5.5.5 <i>XYZ source</i>	2-21
	2.5.5.6 <i>Moving point source</i>	2-22
	2.5.5.7 <i>Vertical distribution</i>	2-22
	2.5.5.8 <i>Mass flux timeseries</i>	2-22
2.5.6	Open Boundaries	2-23
2.6	Particle Tracking Module Output	2-24
2.6.1	Lagrangian	2-24
2.6.2	Eulerian	2-24
2.6.3	Visualisation libraries	2-25
3	References	3-1
	SEDIMENT TRANSPORT COMMANDS	A-2
	PARTICLE TRACKING COMMANDS	B-2

List of Figures

Figure 1	TUFLOW Sediment Transport Module Conceptual Model	1-3
Figure 2	STM Update dt Sequence	1-5
Figure 3	TUFLOW FV Sediment Control File Overview	1-38
Figure 4	An Illustration of Bed Armouring Process	1-40
Figure 5	Bed Slumping – Angle of Repose	1-43
Figure 6	PTM Update dt Sequence	2-3
Figure 7	TUFLOW FV Particle Tracking Control File Overview	2-13

List of Tables

Table 1	TUFLOW STM globally applied profile and bed models	1-6
Table 2	TUFLOW STM Sediment Fraction Models	1-7
Table 3	TUFLOW STM “External” models	1-7
Table 4	Input parameters for VanRijn04 external model	1-28
Table 5	Bed roughness models and associated parameter set requirements	1-42
Table 6	Bed shear stress models and associated parameter set requirements	1-43
Table 7	Bed slumping models and associated parameter set requirements	1-44
Table 8	External models and associated parameter set requirements	1-44
Table 9	Settling models and associated parameter set requirements	1-46
Table 10	Erosion models and associated parameter set requirements	1-47
Table 11	Deposition models and associated parameter set requirements	1-48
Table 12	Bed load models and associated parameter set requirements	1-48
Table 13	Critical stress models and associated parameter set requirements	1-49
Table 14	Consolidation models and associated parameter set requirements	1-50
Table 15	Sediment Transport Module output parameters	1-56
Table 16	Lagrangian particle behaviour options.	2-7
Table 17	Sediment Transport models available to the PTM	2-9
Table 18	Description of particle motility options	2-10
Table 19	PTM Group available models and associated parameter set requirements	2-16
Table 20	Motility models and associated parameter set requirements	2-18
Table 21	Particle Tracking Output Parameters	2-26

1 Sediment Transport Module

Chapter Contents

1.1	Sediment Transport Module Description	1-3
1.1.1	Introduction	1-3
1.2	Available Sediment Transport Models	1-6
1.2.1	Global Models	1-6
1.2.2	Sediment Fraction Models	1-6
1.2.3	External Models	1-7
1.3	Scientific Documentation	1-8
1.3.1	Concentration Profile	1-8
1.3.2	Bed Roughness	1-8
1.3.3	Bed Shear Stress	1-10
1.3.4	Settling Model	1-12
1.3.5	Deposition Model	1-15
1.3.6	Erosion Model	1-17
1.3.7	Bed Load Model	1-20
1.3.8	Critical Stress Model	1-25
1.3.9	Consolidation Model	1-27
1.3.10	External Model	1-28
1.4	TUFLOW FV Control File (.FVC) for STM	1-35
1.4.1	Simulation Configuration	1-35
1.4.2	Materials	1-35
1.4.3	Boundary Conditions	1-35
1.4.4	Initial Conditions	1-35
1.4.5	Outputs	1-36
1.5	STM Control File	1-37
1.5.1	Introduction	1-37
1.6	Simulation Configuration	1-39
1.6.2	Bed restart files	1-41
1.7	Global Model Specifications	1-42
1.8	Sediment Fraction (Group) Blocks	1-45
1.8.1	Sediment properties	1-45
1.8.2	Settling model	1-46
1.8.3	Erosion model	1-47
1.8.4	Deposition model	1-48
1.8.5	Bed load model	1-48
1.8.6	Critical stress model	1-49
1.8.7	Consolidation model	1-50

1.9	Material Blocks	1-51
1.9.2	Sediment fraction properties	1-53
1.9.3	Bed layer properties	1-53
1.9.4	Initial Sediment Mass Conditions	1-54
1.10	STM Outputs	1-55
1.10.1	Timeseries Output	1-55
1.10.2	Profile Output	1-55
1.10.3	Flux Output	1-55
1.10.4	Mass Output	1-55
1.10.5	Map Output Parameters	1-56

1.1 Sediment Transport Module Description

1.1.1 Introduction

The TUFLOW Sediment Transport Module (STM) is a flexible and sophisticated bed load and suspended load sediment transport model that enables the 2D or 3D simulation of:

- Sediment transport in creeks, rivers, reservoirs, estuaries, coastal and ocean environments,
- Sediment transport due to currents and/or wave driven processes,
- Morphological evolution with hydrodynamic feedback,
- Sediment exchange between the water column and the bed (deposition and erosion),
- Advection and dispersion of suspended sediment,
- Bed load transport, bed slumping, bed consolidation and sediment sorting/armouring processes.

One-or-more sediment fractions can be simulated as they are distributed within the bed and transported as bed or suspended load. This fraction-based implementation allows a high level of control over sediment characteristics. For example, within a single model run, sediment fraction groups can be assigned as cohesive or non-cohesive and there is flexibility to select from a range of common sediment transport models/equations independently for each fraction. Figure 1 provides a conceptual model of the discrete sediment transport processes modelled by the STM and each component is mentioned in *Italics* in the following sections.

The STM is a Eulerian solver and fundamentally tracks sediment mass across discrete control volumes defined by the hydrodynamic model mesh or grid. If a Lagrangian scheme is preferred the [Particle Tracking Module](#) provides sediment transport behaviour within a Lagrangian frame of reference.

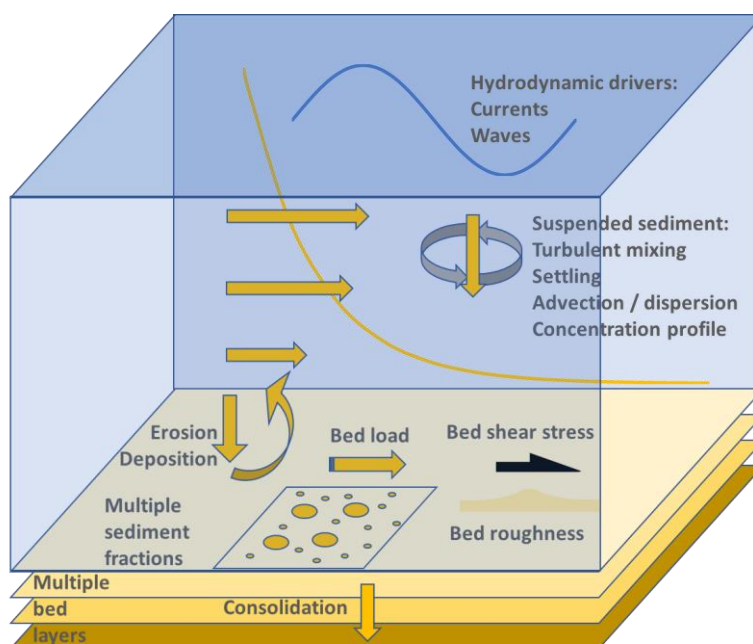


Figure 1 TUFLOW Sediment Transport Module Conceptual Model

The Hydrodynamic (HD) Engine controls the overall simulation and communicates the *Hydrodynamic Drivers* such as *Currents and Waves* to the STM. The HD Engine, Advection Dispersion (AD) Module and STM work together to calculate the horizontal and vertical *Turbulent Mixing, Settling Advection Dispersion* and *Concentration Profiles* of suspended sediment within the water column. *Multiple Suspended Sediment Fractions* are individually tracked as scalar constituents by the AD Module.

Provided with hydrodynamic and suspended sediment conditions, the STM is responsible for resolving the wave-current boundary layer including potential bed forms (*Bed Roughness*) and resultant *Bed Shear Stress*. The STM calculates *Bed Load Transport* potential and sediment *Erosion-Deposition* exchanges at the bed and tracks the resultant change in sediment mass in the bed.

The STM conceptualises the bed as consisting of *Multiple Vertically Stratified Bed Layers*. Each layer comprises a mixture of the defined *Multiple Sediment Fractions*. The mass (kg/m^2) of each sediment fraction within an individual cell's bed layer is tracked by the STM. The total quantity of sediment in each bed layer (kg/m^2) and the associated dry density (kg/m^3) are used to calculate the layer thickness (m).

The uppermost layer that contains a finite sediment mass is the 'active layer' and it is subject to sediment erosion and deposition. If the active layer erodes entirely, sediment is then progressively eroded from the underlying discretised layers. Sediment always deposits to the active layer and can be transferred to lower (usually higher density) layers via *Consolidation*. Sediment sorting and armouring of the active layer is also possible.

Each sediment transport process is calculated on a timestep specified via the STM [Update dt](#) (refer 1.6.1.1) which is set independently to the HD timestep. The STM Update dt will typically be much larger than the HD timestep, for example, the HD timestep may be sub-second to seconds and the STM Update dt may occur every five to fifteen minutes. The purpose of this approach is to speed up simulations which are run over the larger timescales which morphological changes typically occur. The adopted Update dt is specific the sediment transport problem being investigated, and the user is required to use some judgement in its selection.

The net water column exchange, bed load and bed slumping are integrated over an STM Update dt using an explicit Eulerian scheme. At computational cells experiencing net erosion, the mass update proceeds from the active layer down through underlying layers until the calculated mass increment has been accounted for or the sediment source is exhausted. In computational cells experiencing net deposition, the resulting mass increase will be accumulated in the top bed layer.

Sediment consolidation fluxes are subsequently updated by looping through the layers transferring mass to the next lower layer as required. Following the sediment mass updates, the layer thicknesses are re-calculated in each bed cell based on the specified layer dry density values. The total change in bed layer thickness is calculated for each computational cell and is passed to the HD model.

Where [Morphological coupling](#) is specified, the updated bed level is applied to the HD model. For more information on morphological setup and parameters please refer to Section 1.6.1.2.

Figure 2 summarises the compute sequence that occurs at each STM Update dt. The blue boxes indicate steps where there is communication between the STM and HD/AD modules. The brown boxes are computed by the STM directly.

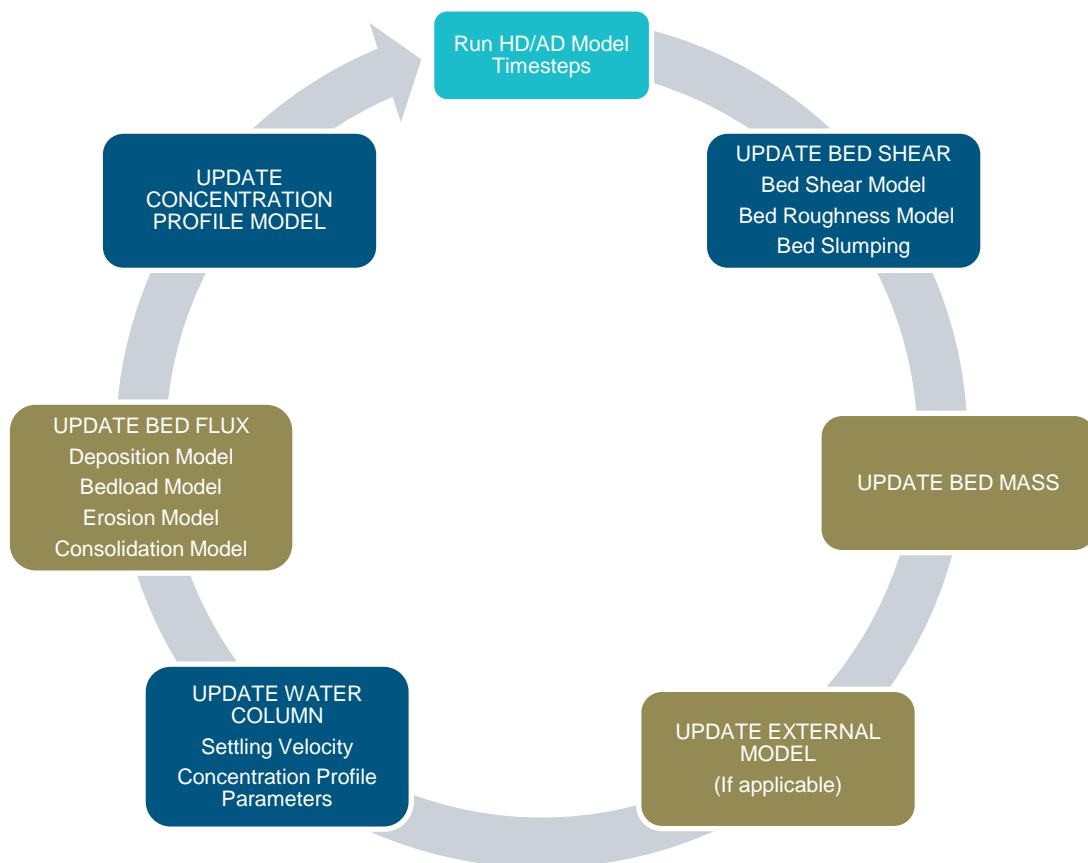


Figure 2 STM Update dt Sequence

1.2 Available Sediment Transport Models

The STM is built on a library of differing model/equation options as provided in Table 1, Table 2 and Table 3. If you have a specific model or equation set not currently available, that you would like to include in the STM, please contact support@tufLOW.com.

The available sediment transport models can be separated into three core categories as follows:

- (1) Global models that are assigned to all sediment fractions. Concentration profile, bed roughness, bed shear stress and bed slumping models are required to be globally specified (Section 1.2.1)
- (2) Sediment fraction models that are assigned to individual sediment fractions independently of other sediment fractions being computed. Settling, erosion, deposition, bed load, critical stress and consolidation models can all be specified on a fraction by fraction basis (Section 1.2.2).
- (3) External models that are assigned globally but can be applied to one-or-more sediment fractions. Currently the van Rijn TRANSPOR model is the only external model supported by the STM (Section 1.2.3).

1.2.1 Global Models

The STM models presented in Table 1 are specified globally and apply to all sediment fractions. The concentration profile models are required to be applied to the entire model domain. In contrast, whilst still applying to all sediment fractions, bed roughness, bed shear and bed slumping models can be varied spatially via material blocks.

More information on each of the globally specified models is provided in Section 1.7.

Table 1 TUFLOW STM globally applied profile and bed models

Concentration profile	Bed roughness	Bed shear stress	Bed slumping
<ul style="list-style-type: none"> • Default (3D higher-order reconstruction) • First-order (higher-order reconstruction only at bed interface) • Legacy (no reconstruction) 	<ul style="list-style-type: none"> • Specified Nikuradse Roughness (ks) • Proportional to median sediment diameter (d50) • van Rijn (2004) 	<ul style="list-style-type: none"> • Default • van Rijn (2004) • Bijker (1984) 	<ul style="list-style-type: none"> • None • Simple (specified slope)

1.2.2 Sediment Fraction Models

The STM provides the flexibility to define fraction independent models/equations. Table 2 provides an overview of the available sediment fraction models for settling, erosion, deposition, bed load, critical shear stress and consolidation. More information on the setup and parameters of each is provided in Section 1.8. Notably, some models are more applicable to cohesive (CS) sediment, while others will be more applicable to non-cohesive (NCS) fractions or a combination of the two. Where applicable the labels NCS or CS are provided next to each sediment fraction model within Table 2. Where a model is applicable to both NCS and CS no label is included.

Table 2 TUFLOW STM Sediment Fraction Models

Settling	Erosion	Deposition	Bed load ¹	Critical Stress	Consolidation
<ul style="list-style-type: none"> • None • Constant ws • Flocculation (CS) • Flocculation + hindered settling (CS) • van Rijn (1984) (NCS) • van Rijn External (2004) (NCS) 	<ul style="list-style-type: none"> • None • Mehta (CS) • van Rijn (1984) (NCS) • van Rijn External (2004) (NCS) • Soulsby-van Rijn (NCS) • Bijker (NCS) 	<ul style="list-style-type: none"> • None • Unhindered • Krone (CS) 	<ul style="list-style-type: none"> • None • Meyer-Peter-Müller • MPM-Shimizu • van Rijn External (2004) • Soulsby-van Rijn • Bijker • Wilcock-Crowe 	<ul style="list-style-type: none"> • None • Constant • Soulsby • Soulsby-Egiazaroff 	<ul style="list-style-type: none"> • None • Constant

¹ Non-cohesive sediment fractions may be transported as both bed load and suspended load, whereas cohesive sediments tend to only be transported as suspended load.

1.2.3 External Models

The STM currently includes the van Rijn TRANSPOR “external” sediment transport model, which in itself provides a comprehensive boundary layer and sediment transport model.

External model parameters are required to be specified globally via the [External model](#) and [External model parameters](#) commands. Once the external model is specified, one-or-more individual sediment fractions can call the van Rijn external model to specify settling, erosion or bed load. Additionally, van Rijn can also be used to globally assign bed roughness and bed shear models for all fractions. The compatible TRANSPOR models are listed as ‘van Rijn External (2004)’ in Table 1 and Table 3. For more information on setup and parameters please refer to Section 1.7.1.6.

Table 3 TUFLOW STM “External” models

External models
<ul style="list-style-type: none"> • van Rijn (2004) TRANSPOR

1.3 Scientific Documentation

1.3.1 Concentration Profile

The default concentration profile model implemented in the STM provides a ‘higher order’ reconstruction of the suspended sediment concentration profile, ultimately providing the STM with suspended sediment concentration estimates at the bottom face of each cell (for a 2D model this is the bed) and the cell’s corresponding cell centre. Using these estimates, vertical suspended sediment settling factors and mixing factors are calculated using the ratio of cell-averaged to cell-bottom concentrations and concentration-gradients respectively within each cell. These factors are passed to the HD model and applied in calculating the net vertical exchange fluxes.

Where relevant (i.e. for 3D model configurations), vertical turbulent mixing between 3D layers in the water column is calculated by the HD Engine and this information is passed back to the STM. The vertical diffusivity for an individual sediment fraction may have a “beta-factor” applied to represent the settling-velocity dependant increase in diffusivity.

1.3.2 Bed Roughness

The prediction of bed roughness is one of the most fundamental problems in the modelling of sediment transport. Bed shear stress (Section 1.3.3) drives incipient particle motion (and by extension sediment transport) and is strongly dependent on bed roughness. Sediment transport in return influences bed roughness by changing bed material distribution and bed forms forming a feedback loop.

The STM offers three common [Bed roughness model](#) options provided as follows:

1.3.2.1 Specified k_s

The k_s option applies a fixed Nikuradse bed roughness height throughout the simulation at a given cell. The bed roughness values are specified by the [Bed roughness parameters](#) command, k_{sc} is the bed roughness for current and k_{sw} the bed roughness for waves. If waves are not modelled k_{sw} is ignored. [Bed roughness parameters](#) command can be specified either globally in the sediment control file, or individually in the [Material](#) block.

```
Bed roughness model == ks
Bed roughness parameters == 0.01,0.01 ! ksc, ksw
```

1.3.2.2 Proportional to d_{50}

With the absence of ripples and dunes, bed roughness is often assumed to be proportional to d_{50} of the bed material. One of the most widely used relationship, according to [Soulsby \(1997, pp48\)](#), is:

$$k_s = 2.5d_{50}$$

However, different studies suggest the proportion may vary widely, and it is strongly recommended to carry out model calibration/sensitivity analysis to assess a suitable multiplier. For situations where wave

forcing is important a second parameter is applied, which is the ratio between the bed roughness's for wave and current (i.e. k_{sw}/k_{sc}).

```
Bed roughness model == d50
Bed roughness parameters == 2.5, 1.0
```

1.3.2.3 van Rijn (2004)

[van Rijn \(2004\)](#) developed a comprehensive model to predict bed roughness for currents and waves considering mixed sediment and bed forms (e.g. ripples, dunes) in coastal environments. The details of the model are documented in Section 1.3.10.

1.3.2.4 Bed roughness coupling

The bed roughness model specified within the STM can either be independent of the HD Engine bottom drag model or coupled to it. If independent, the STM and HD Engine will use two differing calculations for bed roughness and bed shear stresses, the STM calculations used to drive sediment transport processes, the HD Engine bed roughness used to calculate the hydraulic τ_b source term. If coupled, the STM will update k_s in the HD Engine at each STM Update dt. Use the [Bed roughness coupling](#) command to turn this option on:

```
Bed roughness coupling == 1
```

1.3.3 Bed Shear Stress

Bed shear stress is a measure of friction force acting on a bed of channel/coastal area and an essential input parameter for erosion and bedload models. Please also refer to the documentation on the bed roughness model selection within Section 1.3.2.

1.3.3.1 Default

`Bed shear model == default` considers bed shear stresses induced by both currents and waves (where applicable).

Current induced bed shear stress is calculated as:

$$\tau_{b,c} = \rho f_c U^2$$

where:

ρ is density of fluid (kg/m³)

U is depth averaged current velocity for 2D model and bottom cell velocity for 3D model (m/s)

f_c is friction coefficient due to current:

$$f_c = \left[\frac{\kappa}{\ln(11z'/k_{sc})} \right]^2$$

where:

κ is Von Karman constant 0.41

z' is water depth for 2D model and bottom cell thickness for 3D model (m)

k_{sc} is bed roughnesses for currents (m)

Wave induced bed shear stress depends on whether the flow is ‘smooth turbulent’, or ‘rough turbulent’.

This flow regime is decided by the wave Reynolds number Re_w and the relative roughness r :

$$Re_w = \frac{U_w A_w}{\nu}$$

$$r = \frac{A_w}{k_{sw}}$$

where:

U_w is orbital velocity amplitude (m/s)

$A_w = U_w T_w / 2\pi$ is semi-orbital excursion (m)

T_w is wave period (s)

ν is kinematic viscosity of water (m²/s)

k_{sw} is bed roughness for wave (m)

For rough turbulent flow, the rough bed friction coefficient f_{wr} is calculated as:

$$f_{wr} = \exp(5.21r^{-0.194} - 5.98)$$

While for smooth turbulent flow, the rough bed friction coefficient f_{ws} is calculated as:

$$f_{ws} = 0.035Re_w^{-0.16}$$

Wave induced bed shear stress is calculated as:

$$\tau_{b,w} = \frac{1}{2}\rho f_w U_w^2$$

$$f_w = \max(f_{wr}, f_{ws})$$

Combined bed-shear stress due to both currents and waves is:

$$\tau_{b,cw} = [\tau_{b,cw,m}^2 + 0.5\tau_{b,w}^2]^{1/2}$$

with:

$$\tau_{b,cw,m} = \tau_{b,c} \left[1 + 1.2 \left(\frac{\tau_{b,w}}{\tau_{b,c} + \tau_{b,w}} \right)^{3.2} \right]$$

1.3.3.2 Bijker

Bed shear model == Bijker [Bijker \(1967, 1971\)](#)'s bed shear stress model is similar to the default method, but it based on the depth averaged velocity \bar{U} and the depth h for both 2D and 3D models. The Bijker bed shear stress model would typically be used in combination with the Bijker erosion and bed load models. Current induced bed shear stress is calculated as:

$$\tau_{b,c} = \frac{1}{8}\rho f_c \bar{U}^2$$

$$f_c = \frac{8g}{[18\log_{10}(12h/k_{sc})]^2}$$

Wave induced bed shear stress depends on the wave semi-orbital excursion A_w and the relative roughness r as follows:

$$f_w = \begin{cases} \exp(5.2r^{-0.19} - 6) & (A_w > 0.001m) \\ 0 & (\text{otherwise}) \end{cases}$$

$$\tau_{b,w} = \frac{1}{4}\rho f_w U_w^2$$

The combined bed-shear stress due to both currents and waves in this model is:

$$\tau_{b,cw} = [\tau_{b,c}^2 + 0.5\tau_{b,w}^2]^{1/2}$$

1.3.3.3 Van Rijn (2004)

Please refer to Section 1.3.10 for bed shear specification using the external TRANSPOR2004 model.

1.3.4 Settling Model

Suspended sediment concentration (g/m^3 conveniently also mg/L) can enter the water column either input as HD boundary conditions or by erosion from the bed. The STM reconstructs vertical suspended sediment concentration profiles from the cell-averaged sediment concentration values resolved by the HD/AD modules. Based on the specified [Concentration profile model](#) (refer Section 1.3.1) an analytical concentration profile is calculated for each computational cell as a function of an assumed vertical diffusivity and calculated settling velocity

The settling velocity of each sediment fraction is calculated by the STM based on the specified [Settling model](#). A range of settling models are available and are discussed in the sections that follow. In short, the settling velocity is passed to the HD model where it is used to calculate vertical advective exchanges within the water column and to the bed. An internal limiter is applied to the net vertical exchange fluxes (turbulent mixing and settling) to avoid CFL-related numerical instabilities.

1.3.4.1 None

```
Settling model == none
```

Settling of suspended sediment is not modelled.

1.3.4.2 Constant w_s

A constant settling velocity (w_s) defined by the [Settling parameters](#) command is applied to calculate the vertical sediment flux. In an example of still water column, the equation governing the vertical sediment balance is:

$$w_s C = -K_s \frac{\partial C}{\partial z}$$

where:

C is suspended sediment concentration (g/m^3)

K_s is turbulence diffusivity of suspended sediment (m^2/s)

```
Settling model == Constant
```

```
Settling parameters == <ws>
```

1.3.4.3 Flocculation

In estuarine environments, fine sediments (clays/silts etc.) eroded from the upstream catchment can coalesce to form flocs larger than the contributing sediment particle size when encountering saline water. These flocs can then settle at a higher speed than the individual particles.

The settling velocity of the flocs are influenced by sediment concentration and salinity:

$$w_{s,floc} = w_{s0} (C/C_{floc})^\alpha [1 - S_1 \exp(S_2 Sal)]$$

where:

w_{s0} is settling velocity without flocculation (m/s)

C_{floc} is concentration when flocculation commences (g/m^3)

α is a power coefficient

S_1 and S_2 are salinity dependence coefficients

Sal is salinity (psu)

The flocculation model's input parameters can be defined by the [Settling parameters](#) command:

```
Settling model == Flocculation
```

```
Settling parameters == <ws0>, <cfloc>, <alpha>, <s1>, <s2>
```

1.3.4.4 Flocculation + hindered settling

As sediment concentration increases, flocs may begin to settle at a reduced speed due to interactions with neighbouring flocs. This process known as hindered settling can be represented via the addition of $w_{s,hind}$ as follows:

$$w_{s,floc} = w_{s0} [\min(C, C_{hind})/C_{floc}]^\alpha [1 - S_1 \exp(S_2 Sal)]$$

$$w_{s,hind} = w_{s,floc} [1 - \min(1, C/C_{hind})]^n$$

where:

C_{hind} is concentration where hindered settling commences (g/m^3)

n is a power coefficient

```
Settling model == Flocculation-Hindered
```

```
Settling parameters == <ws0>, <cfloc>, <alpha>, <s1>, <s2>, <chind>, <n>
```

1.3.4.5 van Rijn (1984)

For natural sand (no flocculation nor hindering) the settling velocity formula proposed by [van Rijn \(1984b\)](#) can be selected as the [Settling model](#), instead of specifying any w_s value. This model requires no input parameters and the formula is based on a dimensionless grain size D_* :

$$D_* = \left[\frac{g(s-1)}{\nu^2} \right]^{1/3} d$$

where:

g is gravity acceleration (9.81 m/s^2)

s is ratio of densities of sediment and water

ν is kinematic viscosity of water (m²/s)

d is grain diameter (m)

Settling velocity is plotted against D_* as:

$$w_s = \begin{cases} \frac{\nu D_*^3}{18d} & (d \leq 100\mu\text{m}) \\ \frac{10\nu}{d} [(1 + 0.01D_*^3)^{1/2} - 1] & (100\mu\text{m} < d \leq 1000\mu\text{m}) \\ \frac{1.1\nu D_*^{1.5}}{d} & (1000\mu\text{m} < d) \end{cases}$$

Settling model == VanRijn84

1.3.4.6 van Rijn (2004)

Please refer to Section 1.3.10 for settling model specification using the external TRANSPOR model.

1.3.5 Deposition Model

Deposition from the water column to the bed can be optionally switched off, allowed to settle freely based on the settling velocity calculated by the [Settling model](#), or can be subject to a limiting bed shear stress for deposition whereby no material will deposit above a specified shear stress.

Computationally at each cell, the STM internally calculates a deposition factor f_d for each sediment fraction using the specified [Deposition model](#). If using completely unhindered deposition `Deposition model == ws` then f_d is set to 1.0. When using `Deposition model == Krone`, the model calculates the value f_d which ranges from 0 (no deposition) to 1 (unhindered and equivalent to `Deposition model == ws`). Once calculated, f_d is passed back to the hydrodynamic model, where it is used to scale the deposition mass flux ($\text{g/m}^2/\text{s}$) at each HD timestep, which is subsequently integrated over an STM update timestep to obtain the mass exchange between the STM and HD Engine at each [Update dt](#).

The available deposition model equations are further described below, and their parameters are detailed in Section 1.8.4.

1.3.5.1 None

```
Deposition model == none
```

Suspended sediment deposition from water column to bed layer is not calculated. The deposition factor $f_d = 0$.

1.3.5.2 Unhindered

```
Deposition model == ws
```

The deposition flux from water column to bed layer is calculated based on settling velocity and suspended sediment concentration as:

$$F_d = f_d w_s C_b$$

where:

w_s is the settling velocity specified by the [Settling model](#) (m/s)

C_b is near bed sediment concentration (g/m^3)

Note: The deposition factor $f_d = 1$.

1.3.5.3 Krone

```
Deposition model == Krone
```

```
Deposition parameters == <taucd>
```

Commonly known as Krone deposition equation, the deposition flux is adjusted by a deposition factor of:

$$F_d = f_d w_s C_b$$

$$f_d = \left(1 - \frac{\tau_b}{\tau_{cd}}\right)$$

where:

τ_b is bed shear stress (N/m²)

τ_{cd} is critical shear stress for deposition (N/m²)

1.3.6 Erosion Model

Sediment erosion and resuspension are calculated as a mass flux (g/m²/s) using the specified [Erosion model](#) for each sediment fraction and computational cell. The erosion rates for each sediment fraction are passed to the HD Engine and applied as a water column source term to the lowest cell in the water column. A flux limiter is applied where the erosion rate could result in negative sediment mass during a single STM update timestep. The available erosion models and their parameters are detailed in Section 1.8.3 and their equations are detailed in the following sections.

1.3.6.1 None

`Erosion model == None`

Erosion is not modelled.

1.3.6.2 Mehta

The Mehta model (also commonly known as the Partheniades Formula) is a simple shear stress excess formula used to calculate the erosion flux from bed layer to water column:

$$F_e = Er \left(\frac{\tau_{b,cw}}{\tau_{ce}} - 1 \right)^\alpha$$

where:

Er is the erosion rate constant (g/m²s)

$\tau_{b,cw}$ is combined bed shear stress due to currents and waves (N/m²) (see Section 1.3.3)

τ_{ce} is critical shear stress for erosion (N/m²)

α is a power coefficient

This model is simple, but the input parameters should be calibrated based on experimental or field measurement data.

`Erosion model == Metha`

`Erosion parameters == <Er>, <tauce>, <alpha>`

For multi sediment fraction model, the flux for each fraction ($F_{e,i}$) is adjusted based on the fraction of each sediment class (p_i) in the top layer:

$$F_{e,i} = p_i F_e$$

Note that this adjustment for multi sediment fraction model applies to other erosion models as well.

1.3.6.3 van Rijn (1984)

[Garcia and Parker \(1991\)](#) compared seven empirical formulas for bed erosion rate and validated each against a large experimental data set. [van Rijn \(1984b\)](#)'s formula is one of the models that corresponds best with the experimental data, and is recommended for use with currents alone by [Soulsby \(1997\)](#).

pp140). The erosion flux is expressed as the product of settling velocity w_s and reference volumetric concentration C_a :

$$F_e = w_s C_a$$

$$C_a = Er \frac{d}{z_a D_*^{0.3}} \left(\frac{\tau_{b,cw}}{\tau_{ce}} - 1 \right)^{1.5}$$

where:

Er is a coefficient (-)

d is grain size (m)

$\tau_{b,cw}$ is combined bed shear stress due to currents and waves (N/m²) (see Section 1.3.3)

τ_{ce} is critical shear stress for erosion (N/m²)

D_* is dimensionless grain size introduced in Section 1.3.4.5.

z_a is reference height (m)

Note that the reliability of this model depends on the selection of the [Reference height](#) z_a . [Garcia and Parker \(1991\)](#) assumed $z_a = 0.05h$ in their study, while [van Rijn \(2007b\)](#) later recommended to set z_a as half the bed roughness height, with a minimum value of 0.01m. z_a needs to be specified by the [za](#) command in the [Material](#) block.

Erosion model == VanRijn84

Erosion parameters == <Er>, <tauce>

1.3.6.4 Soulsby-vanRijn

Erosion model == Soulsby_VanRijn

The suspended load part of [Soulsby-van Rijn \(1997, pp183\)](#)'s total load model (Section 1.3.7.4) can be used to derive C_a . After obtaining the suspended load q_s by using term A_{sb} , it can be converted to C_a assuming the following vertical profiles for sediment concentration and velocity:

$$C(z) = C_a \left[\frac{z}{z_a} \frac{(h - z_a)}{(h - z)} \right]^{-w_s/\kappa u_*}$$

$$U(z) = \frac{u_*}{\kappa} \ln \left(\frac{z}{z_0} \right)$$

$$q_s = \int_{z_a}^h C(z) U(z) dz$$

where:

z_a is reference height (m)

w_s is settling velocity (m/s)

κ is Von Karman constant 0.41

u_* is bed stress velocity (m/s)

z_0 is bed roughness length (m)

q_s is the suspended load calculated by Soulsby-Van Rijn (1997)'s total load model ($\text{g/m}^2 \cdot \text{s}$)

1.3.6.5 Bijker

Erosion model == Bijker

Similar to the Soulsby-van Rijn (1997)'s method introduced above, the suspended load part of [Bijker \(1967, 1971\)](#)'s total load model (Section 1.3.7.5) can be used to derive C_a .

1.3.6.6 van Rijn (2004)

Please refer to Section 1.3.10 for erosion specification using the external TRANSPOR model.

1.3.7 Bed Load Model

Bed load transport in the STM is calculated for each 2D computational cell representing the bed surface as a mass flux (kg/m/s) vector [Qbx, Qby] using the specified [Bed load model](#). An upwinded, face-normal bed load flux is subsequently calculated at each cell face and a boundary integral is calculated to determine the sediment mass rate of change due to bed load divergence. The available bed load models and their parameters are detailed in Section 1.8.5 and their equations are detailed in the sections that follow.

Among the models offered by TUFLOW FV, [Meyer-Peter and Müller \(1948\)](#)'s and [Wilcock and Crowe \(2003\)](#)'s models are developed for gravel rivers under the under the impact of currents only. [Van Rijn \(2004\)](#)'s, [Soulsby-van Rijn \(1997\)](#)'s and [Bijker \(1967, 1971\)](#)'s models can be used for considering the impact of both currents and waves.

1.3.7.1 None

Bedload model == None

Bedload is not modelled.

1.3.7.2 Meyer-Peter and Müller

[Meyer-Peter and Müller \(1948\)](#)'s bedload model was originally developed for well-sorted fine gravel, and the formula uses non-dimensionalised bed shear stress (or Shield's stress):

$$\tau_* = \frac{\tau_b}{(\rho_s - \rho)gd}$$

to obtain a nondimensionalised bedload transport rate:

$$q_{b*} = \frac{q_b}{\sqrt{(s-1)gd^3}} = 8(\tau_* - \tau_{*c})^{1.5}$$

where:

τ_b is bed stress stress (N/m²)

ρ_s is density of particle (kg/m³)

ρ is density of water

g is gravity acceleration (m/s²)

d is particle size (m)

q_b is the volumetric bedload transport rate per unit width (m³/m·s)

$s = \rho_s/\rho - 1$

τ_{*c} is a constant determined in experiment and the commonly used values are: 0.06 (Shields, gravel), 0.03 (Parker, mixed size gravel), and 0.047 (Meyer-Peter and Müller, well-sorted fine gravel). The value of non-dimensionalised critical shear stress can be also applied as τ_{*c} .

For multi sediment fraction model, bedload rate for each fraction ($q_{b,i}$) is adjusted based on the fraction of each sediment class (p_i) in the top layer:

$$q_{b,i} = p_i q_b$$

Note that this adjustment for multi sediment fraction model applies to other bedload models as well.

The three parameters in the model (factor 8, τ_{*c} , and exponent 1.5) can be specified by user using the following commands:

```
Bedload model == MPM
```

```
Bedload parameters == <fac>, <taucr>, <alpha>
```

1.3.7.3 MPM-Shimizu

[Shimizu et al \(1995\)](#) applied Hasegawa (1983)'s method to consider the impact of bed slope on the direction of bedload transport in the Meyer-Peter and Müller's bedload model. The bedload components in the direction of the bed shear stress \hat{s} and perpendicular to the direction of the bed shear stress \hat{n} have the following relationship:

$$q_{b*} = \sqrt{q_{b*,\hat{s}}^2 + q_{b*,\hat{n}}^2}$$

with

$$\frac{q_{b*,\hat{n}}}{q_{b*,\hat{s}}} = \frac{\sqrt{\tau_{*c}}}{\mu_s \mu_k \tau_*} \frac{\partial z}{\partial \hat{n}}$$

where:

μ_s and μ_k are the static and kinetic friction coefficient (assumed as 0.6 and 0.48, respectively)

$\partial z / \partial \hat{n}$ is the bed slope component perpendicular to bed shear stress direction.

τ_* and τ_{*c} are nondimensionalised bed shear stress and critical shear stress, respectively.

The obtained bedload rate in \hat{s} and \hat{n} directions are then converted to x, y directions using:

$$q_{b*,x} = q_{b*,\hat{s}} \frac{\tau_{*,x}}{\tau_*} - q_{b*,\hat{n}} \frac{\tau_{*,y}}{\tau_*}$$

$$q_{b*,y} = q_{b*,\hat{s}} \frac{\tau_{*,y}}{\tau_*} + q_{b*,\hat{n}} \frac{\tau_{*,x}}{\tau_*}$$

Required bedload parameter inputs are same as the Meyer-Peter and Müller's model.

```
Bedload model == MPM_Shimizu
```

```
Bedload parameters == <fac>, <taucr>, <alpha>
```

1.3.7.4 Soulsby - Van Rijn

Bedload model == Soulsby_VanRijn

[Soulsby \(1997, pp183\)](#) developed a coastal sediment transport model for total load transport rates. Either the load transport rate or just the bedload part can be applied in STM. The total load formula reads:

$$q_t = A_s \bar{U} \left[\left(\bar{U}^2 + \frac{0.018}{C_D} U_{w,rms}^2 \right)^{1/2} - \bar{U}_{cr} \right]^{2.4} (1 - 1.6 \tan \beta)$$

$$A_{sb} = \frac{0.005 h (d_{50}/h)^{1.2}}{[(s-1)g d_{50}]^{1.2}}$$

$$A_{ss} = \frac{0.012 h D_*^{-0.6}}{[(s-1)g d_{50}]^{1.2}}$$

$$A_s = A_{sb} + A_{ss}$$

where:

\bar{U} is depth-averaged current velocity (m/s)

$U_{w,rms}$ is root-mean-square wave orbital velocity (m/s)

C_D is the drag coefficient due to the current along"

$$C_D = \left[\frac{0.40}{\ln(h/z_0)} \right]^2$$

\bar{U}_{cr} is threshold current velocity (m/s)

β is slope of bed in streamwise direction, positive if flow runs uphill

z_0 is bed roughness length (m)

D_* is dimensionless grain size:

$$D_* = \left[\frac{g(s-1)}{v^2} \right]^{1/3} d_{50}$$

v is kinematic viscosity of water (m²/s)

The threshold velocity \bar{U}_{cr} is obtained from the improved Shields' curve proposed by [Soulsby and Whitehouse \(1997\)](#) (see Section 1.3.8.3). The dimensionless critical bed shear stress τ_{*c} is converted to a depth-averaged threshold velocity using:

$$\bar{U}_{cr} = 7 \left(\frac{h}{d_{50}} \right)^{1/7} [g(s-1)d_{50}\tau_{*c}]^{1/2}$$

The total load and bedload model can be selected by:

Bedload model == Soulsby_VanRijn_Total

Or

Bedload model == Soulsby_VanRijn

where A_s becomes A_{sb} only.

Note: this model is developed for sediment sizes of $0.1\text{mm} < d_{50} < 10\text{mm}$ and the model will exit with an error if the sediment size exceeds this range.

1.3.7.5 Bijker

Bedload model == Bijker

[Bijker \(1967, 1971\)](#) proposed the following formula for the net bedload transport rate averaged over a sinusoidal wave-cycle:

$$q_b = A_B u_* d_{50} \exp \left[\frac{-0.27 g (s - 1) d_{50}}{\mu (u_*^2 + 0.016 U_w^2)} \right]$$

$$A_B = \begin{cases} 2 & (H_w/h < 0.05) \\ 2 + 3 (H_w/h - 0.05) & (0.05 \leq H_w/h < 0.4) \\ 5 & (0.4 \leq H_w/h) \end{cases}$$

$$\mu = \left[\frac{\ln(12h/\Delta_r)}{\ln(12h/d_{90})} \right]^{1.5}$$

where:

A_B is the breaking wave coefficient

H_w is wave height (m)

u_* is bed stress velocity due to current alone (m/s)

U_w is wave orbital velocity amplitude (m/s)

μ is 'ripple factor'

Δ_r is ripple height (m)

The parameters for the breaking wave coefficient can be specified by users using the following command:

Bedload parameters == <Abs>, <Gbs>, <Abd>, <Gbd>

with the default values of 5, 0.4, 2, 0.05, respectively. Bijker's bedload model can be applied in conjunction with the Bijker's bed shear stress model (Section 1.3.3.2) for the consistency of the bedload calculation.

The Bijker's total load can be applied, instead of calculating just the bedload rate. Suspended load is related to bedload using the bed concentration reference value:

$$q_s = 1.83 q_b \left[I_1 \ln \left(\frac{33h}{\Delta_r} \right) + I_2 \right]$$

where I_1 and I_2 are Einstein integrals for the suspended load.

To model the total transport rate, use command:

Bedload model == Bijker_Total

To model the bedload only:

Bedload model == Bijker

1.3.7.6 Wilcock-Crowe

Bedload model == Wilcock_Crowe

[Wilcock and Crowe \(2003\)](#) developed a model for mixed sand/gravel sediments, which considers a hiding function and incorporates a nonlinear effect of sand content on gravel transport rate. Bed load transport rates are expressed a dimensionless parameter for each sediment size fraction D_i (note this dimensionless parameter is different from the q_{s*} used in the models above):

$$W_i^* = \frac{(s-1)gq_s}{u_*^3}$$

W_i^* is plotted as a function of $\phi = \tau_b/\tau_{ri}$ as:

$$W_i^* = \begin{cases} 0.002\phi^{7.5} & (\phi < 1.35) \\ 14\left(1 - \frac{0.894}{\phi^{0.5}}\right)^{4.5} & (\phi \geq 1.35) \end{cases}$$

where:

τ_b is bed stress stress (N/m²)

τ_{ri} is the reference shear stress that is defined as the value of τ_b at which W_i^* is equal to a small reference value $W_r^* = 0.002$

τ_{ri} is calculated from the reference shear stress for the mean grain size (D_m) of the bed surface:

$$\frac{\tau_{ri}}{\tau_{rm}} = \left(\frac{D_i}{D_m}\right)^b$$

$$b = \frac{0.67}{1 + \exp(1.5 - D_i/D_m)}$$

These two equations represent a hiding function for each grain size in a mixed sediment, and τ_{rm} is calculated from the following two equation considering the percentage of sand on the bed surface F_s :

$$\tau_{rm}^* = \frac{\tau_{rm}}{(s-1)\rho g D_m}$$

$$\tau_{rm}^* = 0.021 + 0.015 \exp(-20F_s)$$

1.3.7.7 van Rijn (2004)

Please refer to Section 1.3.10 for bed load specification using the external TRANSPOR model.

1.3.8 Critical Stress Model

Critical shear stress τ_{*c} defines the threshold of motion for sediments and is an important input to erosion and bedload formulas. For many erosion models and bedload models, τ_{*c} is built into the model, while Metha's and van Rijn (1984)'s erosion models and Meyer-Peter and Müller and MPM_Shimizu bedload models allow a user specified value or alternatively they use a critical shear stress model. For further details on adding the [Critical stress model](#) and [Critical stress parameters](#) to the model please refer to Section 1.8.6

1.3.8.1 None

```
Critical stress model == None
```

Critical shear stress is not calculated.

1.3.8.2 Constant

Use a constant value defined by the [Critical stress parameters](#) command with a unit "N/m²".

```
Critical stress model == Constant
Critical stress parameters == <tauc>
```

1.3.8.3 Soulsby

[Soulsby and Whitehouse \(1997\)](#) proposed an algebraic expression that improved Shields' curve. The Critical Shields parameter (or dimensionless bed shear stress) is defined as:

$$\tau_{*c} = \frac{\tau_c}{(\rho_s - \rho)gd}$$

It can be plotted against the dimensionless grain size D_* :

$$D_* = \left[\frac{g(s-1)}{\nu^2} \right]^{1/3} d$$

The expression reads:

$$\tau_{*c} = \frac{0.3}{1 + 1.2D_*} + 0.055[1 - \exp(-0.02D_*)]$$

```
Critical stress model == Soulsby
```

Note: No parameters are required for the Soulsby critical stress model.

1.3.8.4 Soulsby-Egjazaroff

```
Critical shear parameters == Soulsby_Egjazaroff
```

The hiding-exposure phenomenon becomes important for mixed sand/gravel beds. The sediment fractions smaller than the median grain size exhibit relatively 'equal' mobility, while the coarser

fractions are more ‘exposed’ than when they exist in a uniform material size bed. As the result, the critical shear stress for the coarser fractions become considerably smaller than that predicted by the Shields’/Soulsby’s curves, which is derived for surface layer with uniform grain size.

The d_{50} method introduced by [Egiazaroff \(1965\)](#) can be applied with the Soulsby’s curve to consider the hiding factor:

$$\xi_i = \frac{\tau_{*c,i}}{\tau_{*c,d_{50}}} = \left[\frac{\log(19)}{\log(19d_i/d_{50})} \right]^2$$

When this method is selected, the dimensionless critical shear for d_{50} is calculated from Soulsby’s curve first and those values for all fractions are obtained based on the relationship above.

1.3.9 Consolidation Model

Bed consolidation of bed sediment can be modelled via downward sediment flux from one bed layer to the next. This can either be switched off or set as a constant download flux using the [Consolidation model](#) command. For more information on bed consolidation setup and parameters please refer to Section 1.8.7.

1.3.10 External Model

Professor Leo C. van Rijn originally published his sediment transport model in 1984 ([van Rijn 1984a, b, c](#)), which focussed on sediment transport and bed roughness in steady river flow. These studies have been cited extensively and validated over a range of flow and sediment conditions. The method was later improved and extended to coastal flow conditions with combined currents and waves, and became a unified model framework for the sediment transport of fine silts to coarse sand and gravel. The updated model was published by [van Rijn et al \(2004\)](#) and implemented as the TRANSPOR2004 model (or TR2004). A Fortran routine of this model was also made available (www.aquapublications.nl) and can optionally be used as an external routine linked to TUFLOW FV hydrodynamic model using the [External model](#) command.

```
External model == VanRijn04
```

The basic hydrodynamic parameters (depth, velocity, water temperature, salinity and etc) required by TR2004 is calculated by TUFLOW FV hydrodynamic model, and the wave related parameters can be linked with external wave model forcing (for example SWAN). The basic sediment characteristics (d_{10} , d_{50} , d_{90} and etc) need to be specified globally using the [External model parameters](#) command and the meanings of the parameters are described in Table 4:

```
External model parameters == <d10>, <d50>, <d90>, <ur>,
                             <bf_type>, <f_ws>, <f_tauc>,
                             <f_current_efficiency>, <f_wave_efficiency>,
                             <f_wave_assymetry>, <f_wave_streaming>
```

Table 4 Input parameters for VanRijn04 external model

Parameters	Unit	Description
d10	m	Grain diameter exceeded by 90% of total sample
d50	m	Median grain diameter of bed material
d90	m	Grain diameter exceeded by 10% of total sample
ur	m/s	Wave induced return velocity. If set to 9, the default method is used to calculate the return velocity. (see Appendix A of 1)
bf_type	-	Bed form type. 1 = river, 2 = estuary, 3 = sea (Section 1.3.10.2)
f_ws	-	A factor to adjust settling velocity. Set as 1.0 to keep settling velocity unchanged. (Section 1.3.10.4)
f_tauc	-	A factor to adjust critical shear stress. Set as 1.0 to keep critical shear stress unchanged. (Section 1.3.10.3)
f_current_efficiency	-	A factor to adjust current related efficiency factor. Set as 1.0 to keep the default setting. (Section 1.3.10.5)
f_wave_efficiency	-	A factor to adjust wave related efficiency factor. Set as 1.0 to keep the default setting. (Section 1.3.10.5)

Parameters	Unit	Description
f_wave_assymetry	-	A factor to adjust wave velocity asymmetry effects. Set as 1.0 to keep the default setting. (see Section 3.2.5 of van Rijn et al (2004))
f_wave_streaming	-	A factor to adjust wave-induced streaming velocity. Set as 1.0 to keep the default setting. (see Section 3.2.5 of van Rijn et al (2004))

In the case of multiple sediment fractions the TR2004 model is run for the combined sediment case, that is all sediment fractions. In this case the local d_{10} , d_{50} , d_{90} variables are calculated internally and the input parameter values are ignored.

The model includes the predictions of bed roughness, bed shear stress, suspended sediment transport and bedload transport. This manual only scratch the surface of the TR2004 model, while the in-depth description of the model can be found in [van Rijn et al \(2004\)](#) or [van Rijn \(2007a, b, c\)](#).

1.3.10.1 TR2004 Bed Forms and Bed Roughness

Current-related bed roughness

Currents and waves may deform a bed into various types of bed features. TR2004 model categorise the bed features into small ripples, mega-ripples and dunes, and the size/height of these features are influenced by the wave-current regime, which is decided by the mobility parameter ψ :

$$\psi = \frac{U_{wc}}{(s-1)gd_{50}}$$

$$U_{wc}^2 = U_c^2 + U_w^2$$

where:

s is ratio of densities of sediment and water

U_c is depth-averaged current velocity (m/s)

U_w is peak orbital velocity near bed (m/s)

It is assumed that the physical current-related roughness of small-scale ripples is given by:

$$k_{s.c.r} = \begin{cases} 150f_{cs}d_{50} & (\psi \leq 50 \text{ lower wave - current regime, movable ripples}) \\ (182.5 - 0.652\psi)f_{cs}d_{50} & (50 < \psi \leq 250 \text{ transitional regime}) \\ 20f_{cs}d_{50} & (\psi > 250 \text{ upper wave - current regime, sheet flow}) \\ 20d_{silt} & (d_{50} < d_{silt}) \end{cases}$$

$$f_{cs} = \max \left[\left(0.25d_{gravel}/d_{50} \right)^{1.5}, 1 \right]$$

with $d_{silt} = 0.032\text{mm}$, and $d_{gravel} = 2\text{mm}$.

The bed roughness of mega-ripples is expressed as a function of the flow depth h and the mobility parameter ψ :

$$k_{s.c.mr} = \begin{cases} 0.0002f_{fs}\psi h & (\psi \leq 50) \\ (0.011 - 0.00002\psi)f_{fs}h & (50 < \psi \leq 550) \\ 0.02 & (\psi > 550 \text{ and } d_{50} \geq 1.5d_{sand}) \\ 200d_{50} & (\psi > 550 \text{ and } d_{50} < 1.5d_{sand}) \\ 0 & (d_{50} < d_{silt}) \end{cases}$$

$$f_{fs} = \max[(d_{50}/1.5d_{sand}), 1]$$

with $d_{sand} = 0.062\text{mm}$.

Similar as for the roughness of mega-ripples, the effective roughness of dunes is proposed to be:

$$k_{s.c.d} = \begin{cases} 0.0004f_{fs}\psi h & (\psi \leq 100) \\ (0.048 - 0.00008\psi)f_{fs}h & (100 < \psi \leq 600) \\ 0 & (\psi > 600) \\ 0 & (d_{50} < d_{silt}) \end{cases}$$

Note that $k_{s.c.d}$ is only calculated if the [External model parameters](#) 'bf_type' is set to 1 (for rivers).

Finally, the total physical current-related roughness ($k_{s,c}$) is assumed to be:

$$k_{s,c} = [k_{s,c,r}^2 + k_{s,c,mr}^2 + k_{s,c,d}^2]^{0.5}$$

Also, 'apparent' bed roughness k_a is needed in the model as it is the dominant roughness factor due to wave-current interaction processes:

$$\frac{k_a}{k_{s,c}} = \exp\left(\frac{\gamma U_w}{U_c}\right), \text{ with } \left(\frac{k_a}{k_{s,c}}\right)_{max} = 10$$

$$\gamma = 0.8 + \varphi - 0.3\varphi^2$$

where:

U_c is depth averaged current velocity (m/s)

U_w is strength of the peak orbital velocity (m/s)

φ is angle between wave direction and current direction (in radians between 0 and π)

Wave-related bed roughness

Regarding the physical wave-related bed roughness, it is assumed that only ripples with a length scale of the order of the wave orbital diameter near the bed are relevant, while mega-ripples and dunes with much larger length scales do not contribute to the roughness, thus:

$$k_{s,w} = k_{s,w,r} = k_{s,c,r}$$

If the [Bed roughness coupling](#) is set to 1, the apparent bed roughness k_a and the wave-related bed roughness $k_{s,w}$ values are passed back to the hydrodynamic and wave models, respectively.

1.3.10.2 TR2004 Bed Shear Stress

The current-related friction coefficient (based on the dimensionless Darcy-Weisbach approach) can be computed as:

$$f_c = \frac{8g}{[18\log_{10}(12h/k_{s,c})]^2}$$

And the current induced bed shear stress is calculated as:

$$\tau_{b,c} = \frac{1}{8}\rho f_c U_c^2$$

The wave-related friction coefficient is computed as:

$$f_w = \exp\left(5.2 \frac{A_w}{k_{s,w}}^{-0.19} - 6\right)$$

where:

$A_w = U_w T_w / 2\pi$ is semi-orbital excursion (m)

T_w is wave period (s)

And the time-averaged bed shear stress induced by wave induced is calculated as:

$$\tau_{b,w} = \frac{1}{4}\rho f_w U_w^2$$

Finally, the combined bed-shear stress due to both currents and waves in this model is:

$$\tau_{b,cw} = \alpha_{cw}\tau_{b,c} + \tau_{b,w}$$

$$\alpha_{cw} = \left[\frac{\ln(30\delta_m/k_a)}{\ln(30\delta_m/k_{s,c})}\right]^2 \left[\frac{\ln(30h/k_{s,c}) - 1}{\ln(30h/k_a) - 1}\right]^2, \text{ with } \alpha_{cw,max} = 1$$

where

α_{cw} is wave-current interaction coefficient

$\delta_m = 2\delta_w$ ($\delta_{m,min} = 0.05$, $\delta_{m,max} = 0.2$) is thickness of effective fluid mixing layer (m)

$2\delta_w$ is thickness of wave-boundary layer (m):

$$\delta_w = 0.36A_\delta \left(\frac{A_\delta}{k_{s,w}}\right)^{-0.25}$$

A_δ is near-bed peak orbital excursion (m)

1.3.10.3 TR2004 Critical Shear Stress

The dimensionless bed shear stress τ_{*c} is calculated as a function of the dimensionless grain size D_* according to the improved Shields' curve proposed by van Rijn:

$$\tau_{*c} = \begin{cases} 0.115D_*^{-0.5} & (1 < D_* \leq 4) \\ 0.14D_*^{-0.64} & (4 < D_* \leq 10) \\ 0.04D_*^{-0.1} & (10 < D_* \leq 20) \\ 0.013D_*^{0.29} & (20 < D_* \leq 150) \\ 0.055 & (150 < D_*) \end{cases}$$

with:

$$\tau_{*c} = \frac{\tau_c}{(\rho_s - \rho)gd}, \quad D_* = \left[\frac{g(s-1)}{v^2} \right]^{1/3} d$$

where:

τ_c is critical shear stress (N/m²)

An adjusted critical shear stress $\tau_{c,1}$ based on the fraction of mud is also used in the erosion/bedload models:

$$\tau_{c,1} = (1 - p_{mud})^3 \tau_c$$

where:

p_{mud} is fraction of mud (0 to 0.3)

In addition, the 'f_tauc' parameter from the [External model parameters](#) command can be used to adjust the calculated critical shear stress.

1.3.10.4 TR2004 Settling Velocity

The settling velocity model is basically same with the [van Rijn \(1984b\)](#) model (Section 1.3.4.5). The 'f_ws' parameter from the [External model parameters](#) command can be used to adjust the calculated settling velocity.

1.3.10.5 TR2004 Erosion Model

The erosion flux is expressed as the product of settling velocity w_s and reference volumetric concentration C_a :

$$F_e = w_s C_a$$

For single sediment fraction model, C_a is calculated as:

$$C_a = 0.015(1 - p_{mud}) \frac{d_{50}}{z_a} D_*^{-0.3} T_{cw}^{1.5}$$

$$C_{a,max} = 0.05$$

where:

$z_a = \max(0.5k_{s,c,r}, 0.5k_{s,w,r}, 0.01)$ is reference height (m). Note that this z_a is calculated independently from the z_a parameter specified in the [Material](#) block.

T_{cw} is dimensionless bed-shear stress parameter under the combined effect of currents and waves:

$$T_{cw} = \frac{\tau'_{b,cw} - \tau_{c,1}}{\tau_c}$$

where:

$\tau'_{b,cw}$ is effective bed-shear stress acting on a given bed material size (or grain-related bed shear stress) (N/m²). Under the impacts of both currents and waves, $\tau'_{b,cw}$ is calculated as:

$$\tau'_{b,cw} = \alpha_{cw}\mu_c\tau_{b,c} + \mu_w\tau_{b,w}$$

where:

α_{cw} is wave-current interaction coefficient (see Section 1.3.10.2)

μ_c and μ_w are current related and wave related efficiency factors, respectively:

$$\mu_c = f'_c / f_c$$

$$f'_c = \frac{8g}{[18\log_{10}(12h/k_{90})]^2}$$

$$\mu_w = 0.7/D_*$$

$$\mu_{w,min} = 0.14 \text{ for } D_* \geq 5$$

$$\mu_{w,max} = 0.35 \text{ for } D_* \leq 2$$

Furthermore, the [External model parameters](#) 'f_current_efficiency' and 'f_wave_efficiency' can be used to adjust μ_c and μ_w .

For multi sediment fraction model, $C_{a,i}$ for each fraction is calculated as:

$$C_{a,i} = 0.015p_i \frac{d_i}{z_a} D_{*,i}^{-0.3} T_{cw,i}^{1.5}$$

$$C_{a,max,i} = 0.05$$

$$T_{cw,i} = \lambda_i \frac{\tau'_{b,cw} - \tau_{c,1}(d_i/d_{50})\xi_i}{\tau_c(d_i/d_{50})}$$

where:

p_i is fraction of sediment class d_i

$\lambda_i = (d_i / d_{50})^{0.5}$ is correction factor of excess bed-shear stress related to grain roughness effects

$\xi_i = [\log(19)/\log(19d_i / d_{50})]^2$ is hiding factor from [Egiazaroff \(1965\)](#)

1.3.10.6 TR2004 Bedload Model

The bed load transport rate model for single sediment fraction model is calculated as:

$$q_b = 0.5(1 - p_{mud})f_{slope1}\rho_s d_{50} D_*^{-0.3} (\tau'_{b,cw}/\rho)^{0.5} T_{cw}$$

where:

$f_{slope1} = 1/(1+\beta_{slope}/0.6)$ is an adjustment factor for bed-slope effects.

β_{slope} is bed-slope in the direction of flow

For multiple sediment fractions:

$$q_{b,i} = 0.5p_i f_{slope1} \rho_s d_i D_{*,i}^{-0.3} (\tau'_{b,cw,i}/\rho)^{0.5} T_{cw,i}$$

1.4 TUFLOW FV Control File (.FVC) for STM

The HD Engine in combination with the AD Module are responsible for providing the drivers to the STM. This section summarises the mandatory and optional commands in the TUFLOW FV Control File (.fvc) required to simulate sediment transport. For full details on the HD and AD setup, please refer to the [TUFLOW FV User Manual](#).

1.4.1 Simulation Configuration

To enable sediment there are two mandatory commands that are required in the .fvc file as follows:

The include sediment command:

```
Include Sediment == 1,0 ! (Enabled, Density coupling)
```

And the sediment control file command whose content is described in detail within Section 1.5:

```
Sediment control file == ..\SED_001.fvsted ! .fvsted
```

Please refer to the Simulation Configuration Chapter of the [TUFLOW FV User Manual](#) for further information.

1.4.2 Materials

An integer material ID is assigned to each model cell within the .fvc. This same material ID is used by the STM to apply spatially varying sediment characteristics. For more information on assigning the location and ID of materials please refer to the [TUFLOW FV User Manual](#) Model Geometry Chapter.

1.4.3 Boundary Conditions

Suspended sediment concentration (in mg/L) can optionally be specified as scalar inputs to a number of different compatible boundary condition types (for the full range of options please refer to the [TUFLOW FV User Manual](#) Boundary Condition Chapter). A zero gradient bed load boundary condition can be optionally specified using the bed load transport bc block command. For example:

```
bc == Q, 1, ..\bc_dbase\Upstream_Q_Temp_Sal_Sed_WQ_002.csv
bc header == time_hr, flow_m3s-1, sal_ppt, temp_degC, FineSed_mgL-1, Sand_mgL-1
bed load transport == 1
end bc
```

1.4.4 Initial Conditions

The HD Engine handles initial suspended sediment conditions and bed elevation where the latter can be variable when morphological coupling is enabled. Initial conditions relating to the bed form are computed and saved by the STM and their setup and usage is detailed in Sections 1.9.4 (initial bed mass distribution) and Section 1.10 (bed restart files).

Initial suspended sediment concentrations (in mg/L) can optionally be specified in the .fvc using the initial sediment concentration, initial scalar profile, initial condition 2D, initial condition 3D commands. Alternatively, hydraulic restart files from a previous run can be saved via the .fvc command write restart dt and read using the restart file command

For further information on hydraulic initial conditions and restart files please refer to the Boundary and Initial Conditions Chapter of the [TUFLOW FV User Manual](#).

1.4.5 Outputs

The HD Engine is responsible for outputting all results from the STM using output blocks in the .fvc. Further discussion on STM outputs is provided Section 1.10.

1.5 STM Control File

1.5.1 Introduction

The sediment control file contains the commands required to define sediment characteristics and processes. It is called by the HD Engine driver file via the .fvc command sediment control file (Section 1.4). The sediment control file can be broken down into four key sections command types as follows:

- Simulation Configuration
- Global Model Specifications
- Sediment Fraction Blocks
- Material Blocks
 - Layer sub-blocks

Within each of these broad command types, there are a wide range of different options as showcased in Figure 3

Core to the implementation of the STM is the concept of *sediment fractions* and the use of *sediment fraction blocks*. A run can have one or more sediment fractions, up to a total of ten although it should be noted that a typical assessment may use two or three fractions or on occasion five fractions. An increased number of sediment fractions will impact upon the speed of the simulation and size of the model output files. The number of fractions required will be based on the particle size distribution for the site and sediment characteristics. It is advised to initially start with less sediment fractions and build up complexity as needed.

Commands are applied in a cascading manner. For example, simulation and global model commands are applied to all sediment fractions being simulated. The sediment fraction blocks that follow provide independent, flexible control over how each sediment type is simulated. *Material blocks* allow for the assignment of spatial variability in sediment parameters. Additionally, *layer sub-blocks* that reside within material blocks can be used to vary sediment quantities and properties both spatially and as a function of bed layer.

Each of the command types and their options are detailed within Sections 1.6 to 1.9 that follow. These sections link heavily with Appendix B Appendix A which details required syntax and example syntax use.

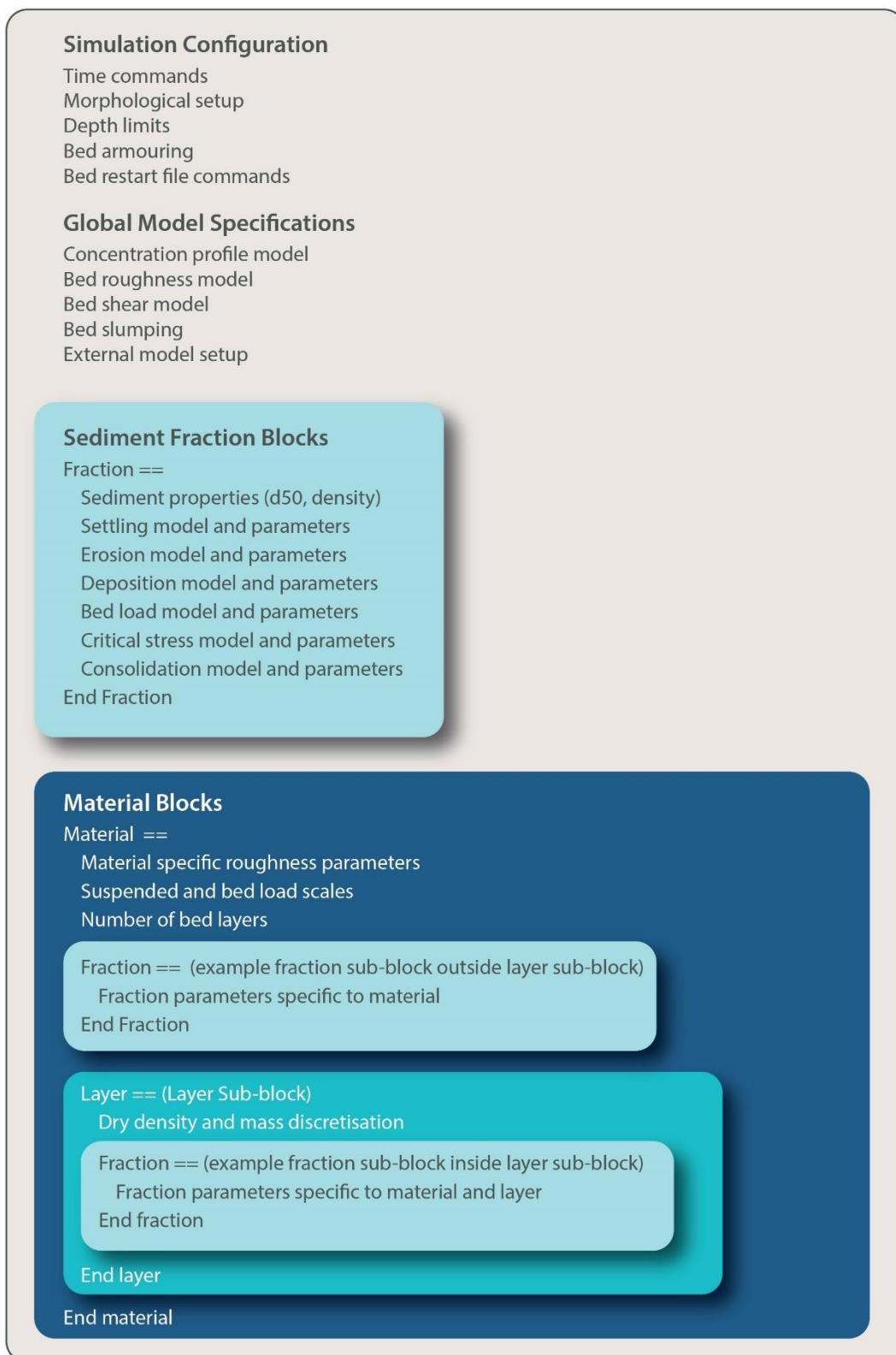


Figure 3 TUFLOW FV Sediment Control File Overview

1.6 Simulation Configuration

1.6.1.1 Time commands

The STM simulation timestep ([Update dt](#)) is required to be set (in seconds). This defines the update interval to share information between the STM and HD/AD model (refer Figure 2). An optional [Start time](#) can be specified if the STM simulation should commence after the HD simulation has warmed up for a period.

```
Update dt == 600. ! s
Start Time == 1.0 ! hours
```

1.6.1.2 Morphological flags

The STM can optionally provide morphological feedback to the HD model using the [Morphological coupling](#) flag.

```
Morphological Coupling == 1
```

Set to 1 to enable morphological coupling, 0 to disable.

A [Morfac](#) (>1.0) can be specified in order to accelerate the evolution of bed mass composition and morphology. This might be undertaken in order to warm up the bed for a subsequent simulation or may be undertaken in order to represent a longer period of evolution than the actual simulation time.

```
Morfac == 12.0
```

[Morfac](#) will multiply the mass transferred in all exchanges during sediment calculations. i.e. between the bed and water column (via pickup and settling) and will also increase any bedload movement. If using a value of 12, 12x more mass transfer will occur than if set to 1 (the default). Using a morfac is commonly used as a way to artificially extend the time period being modelled. For example, a year of currents and/or wave data could be simulated with a morfac of 5 to pseudo represent the sediment transport behaviour that would occur over 5 years of the same flow conditions.

Note that the [Morfac](#) is only applied within the STM bed update and is not applied to the sediment exchange fluxes within the HD model. Therefore, a simulation using a [Morfac](#) ($\neq 1.0$) will not necessarily conserve sediment mass globally.

While use of a [Morfac](#) (>1.0) during warmup simulation/s can accelerate this process it is not always guaranteed to evolve towards the same 'equilibrium' state as a simulation with [Morfac](#)=1.0. In particular, the model may respond to short-term erosion events in an exaggerated and unrealistic manner.

1.6.1.3 Depth limit commands

A set of global depth limits may be applied in order to control particle erosion and deposition behaviour in shallow cells.

[Erosion depth limits](#): Erosion rate is limited to 0 if water depth is less than 0.05m. Between, 0.05 and 0.1m, the erosion rate is linearly scaled down. This is used to avoid unreasonable erosion in cells with

small depths. This is particularly applicable at the wet-dry interface. Please note this will affect both bed load transport and bed pickup into the water column. For example:

```
Erosion depth limits == 0.05, 0.10
```

Deposition depth limits: Analogous to erosion depth limits but for deposition. For example:

```
Deposition depth limits == 0.01, 0.05
```

Wave depth limits: For limiting wave height in shallow water. For example:

```
Wave depth limits == 0.05, 0.15
```

1.6.1.4 Bed Armouring

Natural bed/riverbed materials typically consist of sediment mixtures comprising different grain sizes and sediment types. It is easier to mobilise finer sands and silts, which may leave coarser materials behind and form an armouring top layer that can protect underlying finer materials from being further eroded. A bed **Armour layer thickness** specification can be used to control the amount of material in the top active layer where multiple bed layers are specified. If the updated active layer thickness is less than the specified minimum, then mass is exchanged from the underlying layers in order to address this shortfall. Otherwise, if the updated active layer thickness is more than the specified maximum value then mass is exchanged to the next layer.

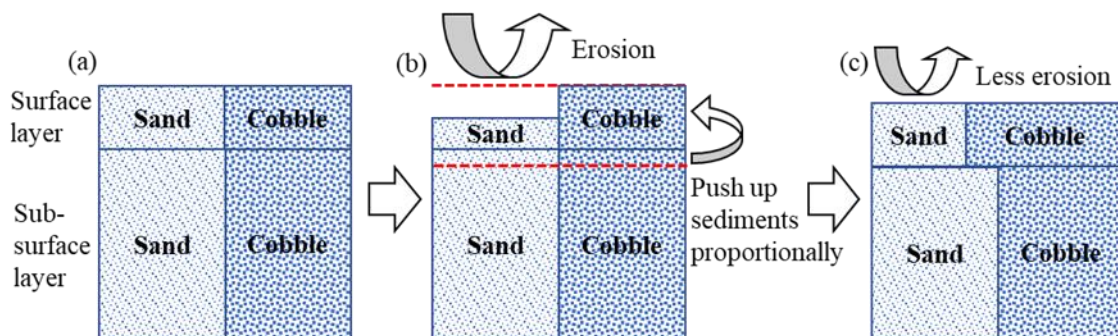


Figure 4 An Illustration of Bed Armouring Process

The **Armour layer thickness** command has two parameters. Sediments from the underlying layers are pushed up if the first layer thickness become smaller than 'min_thickness', while the sediment in the first layer is pushed down to the underlying layer if the first layer thickness exceeds 'max_thickness'. The default **Armour layer thickness** limits are [0., 9999.] (in metres), where the minimum armour layer thickness limit of 0 means the top layer can be completely eroded.

```
Armour layer thickness == min_thickness, max_thickness !m
```

Note that apart from the resulting adjustments to the surface layer sediment composition, the layer properties (e.g. dry density and erosion parameters) are not adjusted as sediment migrates up from underlying layers. Therefore, the armour layer minimum thickness may not be an appropriate schematisation of a progressively eroding bed with increasingly stiff underlying layers. In this case a zero minimum thickness limit would be appropriate.

1.6.2 Bed restart files

[Bed restart files](#) may be output as part of an STM simulation and can be used to initialise subsequent simulations. The restart file saves the mass distribution of sediment fractions within the bed and are often used as initial model conditions following a bed warmup simulation (Section 1.6.2.1). A STM bed restart file is always output upon successful completion of a simulation and may optionally be written at a specified frequency during the simulation using the [write restart dt](#) command. The restart file can also be optionally overwritten at the restart dt using the [restart overwrite](#) command.

Using a restart file as an initial condition is as simple as providing a path to an existing restart file:-

```
bed restart file == log\PreviousRun_bed.rst
```

Note the bed restart file should not be confused with the HD restart file specified within the .fvc. The HD model restart file will save suspended sediment information and also the bed elevation, the latter being important if morphological coupling is enabled.

1.6.2.1 Bed Warmup

Creating a bed warmup file is an important model setup workflow commonly used to initialise a sediment model. A typical workflow is to provide a ‘best-estimate’ of an initial bed profile via the use of [Material](#) blocks, [Layer](#) sub-blocks and the [Initial mass](#) command. The bed warm up model is run with representative flows for the study to allow the bed to reach as close to an ‘equilibrium’ condition as possible. This process allows the model to re-distribute sediment spatially throughout the domain based on the applied hydraulic condition. i.e. some areas of the model will have deposition and thicker layers of sediment, whilst others will erode. This aims to ensure that sediment is in the ‘right’ place. i.e. we don’t start a model with fine sands in a location where they are immediately eroded.

The [Morfac](#) command is often used to speed up the model warm up process however some care needs to be exercised with this approach. For example, the model can move towards the desired ‘equilibrium’ state, that mimics the natural setting, but it can also lead to excessive sediment movement. If setting up the bed using *Morfac*, it is common to conduct a further sediment stabilisation simulation by running the model for an extended period with normal morphological coupling set (using a restart file from the *morfac* run).

1.7 Global Model Specifications

Global model specifications and associated parameter sets apply to all sediment fractions across all cells in the domain. A number of bed roughness, bed shear and bed slumping parameters can also be varied spatially using [Material](#) block specifications (refer 1.7.1.1, 1.7.1.2 and 1.7.1.3).

1.7.1.1 Bed roughness model

A Globally specified [Bed roughness model](#) is a mandatory global input and there are three options available as detailed in Table 5. More information on each bed roughness model is provided within the ‘Scientific Documentation’ part of this manual, Section 1.3.2.

```
Bed roughness model == ks                ! Nikuradse Roughness Height
Bed roughness parameters == 0.01,0.01    ! ksc, ksw (m)
```

Only one globally applied [Bed roughness model](#) can be selected for a given simulation. [Bed roughness parameters](#) are specified globally but can also be varied spatially within [Material](#) blocks.

If [Bed roughness coupling](#) is set to 1, the k_{sa} value is passed back to the HD model where it is used to calculate bed friction terms. This coupling should only be applied if the parent HD model is also using the k_s bed roughness model (for more information the HD Engine bed roughness specification refer to the [TUFLOW FV User Manual](#)).

Table 5 Bed roughness models and associated parameter set requirements

Model Names	Model Description	Model Parameter Requirements
k_s (Nikuradse Roughness Height)	The user directly specifies a k_{sc} and k_{sw} roughness height (m)	Npar=2 <ul style="list-style-type: none"> k_{sc} (currents) k_{sw} (waves)
d50 (Factored by modelled median grain size)	$k_{sc} = fac1 * d50$ ($k_{sa} = k_{sc}$) $k_{sw} = fac2 * d50$	Npar=2 <ul style="list-style-type: none"> fac1 fac2
vanRijn04	k_{sc} , k_{sw} and k_{sa} are calculated by van Rijn’s TRANSPOR model	None (Please note the external model and external model parameters are required to be globally specified)

1.7.1.2 Bed shear model

The [Bed shear model](#) calculates the effective bed shear stress used in other STM models (e.g. erosion model). In general, the bed shear stress model considers both steady current and oscillatory wave related shear stress components as well as combined wave-current shear stress statistics such as:

- τ_{auc} – current only bed shear stress component
- τ_{auw} – oscillatory wave bed shear stress amplitude

- $\tau_{\text{aucw_av}}$ – time-averaged bed shear stress due to wave-current interaction
- $\tau_{\text{aucw_rms}}$ – root-mean-square bed shear stress due to wave-current interaction
- $\tau_{\text{aucw_max}}$ – maximum shear stress amplitude due to wave-current interaction

Table 6 Bed shear stress models and associated parameter set requirements

Model Names	Model Description	Model Parameter Requirements
Default	Follows approach outlined in Soulsby (1997). Suitable for combined wave/currents.	None
Bijker	Bijker (1967, 1971) method. Suitable for combined wave/currents.	None
vanRijn04	van Rijn’s TRANSPOR model. Suitable for combined wave/currents.	None, requires specification of external model and external model parameters .

1.7.1.3 Bed slumping model

Slope failure due to excessive bed slope gradients can optionally be enabled. The maximum stable bed slope is calculated based on the specified [Bed slumping model](#). Where the maximum slope is exceeded between adjacent computational cells a slumping flux is calculated in order to return the slope to its limiting value. The slumping flux is applied with the integrated bed load flux in updating the bed mass. For more information on bed slumping setup and parameters please refer to Section 1.7.1.3.

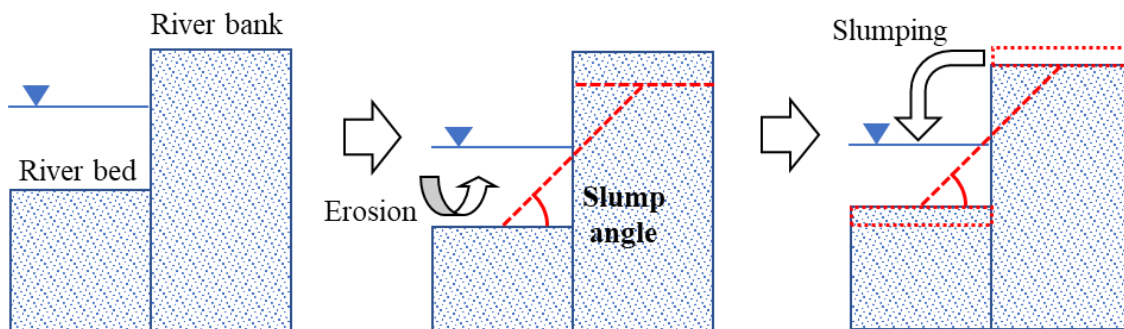


Figure 5 Bed Slumping – Angle of Repose

1.7.1.4 *None*

`Bed slumping model == None`

Slope failure is not modelled.

1.7.1.5 Simple (angle of repose)

A simple slope failure model described above is used. The stable slope angles are defined by the [Bed slumping parameters](#) command, with the first value being underwater slumping angle and the second one being the slumping angle between dry/wet cells.

```
Bed slumping model == Simple
```

```
Bed slumping parameters == <subwater_slope>, <subaerial_slope>
```

The [Bed slumping model](#) calculates the stable bed slope/s to be used in the STM bed update routine. Different sub-aqueous (underwater) and sub-aerial (above water) slopes can be calculated.

Table 7 Bed slumping models and associated parameter set requirements

Model Names	Model Description	Model Parameter Requirements
None	No upper limiting bed slope.	None
Simple	Simple angle of repose.	Npar=2 <ul style="list-style-type: none"> Limiting slope (wet) Limiting slope (dry)

1.7.1.6 “External” models

van Rijn’s TRANSPOR model is available as an [External model](#) option. The associated parameter requirements are described in Table 8. Parameters are assigned using the [External model parameters](#) command. Further detail on the technical aspects of TRANSPOR are provided in Section 1.3.10.

Table 8 External models and associated parameter set requirements

Model Names	Model Description	Model Parameter Requirements
None	-	None
vanRijn04	Van Rijn’s (2004) TRANSPOR model is used to calculate bed roughness, bed shear stress, erosion and bed load transport for all sediment fractions. The modelled values can be applied individually by specifying vanRijn04 for each model. e.g. bed shear model == vanRijn04 erosion model = vanRijn04	Npar=11 <ul style="list-style-type: none"> d10 (only used if Nsed==1) d50 (only used if Nsed==1) d90 (only used if Nsed==1) f_wave_current ur f_w f_tauc f_current_efficiency f_wave_efficiency f_wave_assymetry f_wave_streaming

1.8 Sediment Fraction (Group) Blocks

The TUFLOW STM is configured to allow for the simulation of multiple sediment fractions, representing different grain sizes and properties.

A sediment [Fraction](#) block is required to define properties and model specifications for each sediment fraction. A unique text identifier may be used to identify the sediment fraction, or alternatively an integer id can be used.

```
Fraction == silt
...
End Fraction
```

Alternatively, the syntax `Group == silt` can be used interchangeably with the above.

The total number of sediment fractions (Nsed) is determined by the number of fraction blocks.

The follow sub-sections provide a description of the various sediment fraction properties, models and parameters that can be input within a sediment fraction block.

1.8.1 Sediment properties

Mandatory: The fundamental sediment properties Median grain size ([d50](#)) in metres and [Particle density](#) in kg/m³ must be set in each sediment fraction block.

```
d50 == 0.0002           ! [m]
Particle density == 2650. ! [kg/m3]
```

1.8.2 Settling model

The [Settling model](#) calculates the sediment fraction settling velocity (m/s) at each HD model computational cell. The available settling models and associated parameter requirements are detailed in Table 9. Model parameters are assigned using the [Settling parameters](#) command. Further detail on the technical aspects of each settling model are provided in Section 1.3.4.

Table 9 Settling models and associated parameter set requirements

Model Names	Model Description	Model Parameter Requirements
None (Default)	ws=0.	None
Constant	Constant settling velocity, $w_s = w_{s0}$.	Npar=1 <ul style="list-style-type: none"> ws0 (m/s)
Flocculation	Settling velocity dependent on flocculation and varies with both TSS and salinity.	Npar=6 <ul style="list-style-type: none"> ws0 (Clear water settling velocity) cfloc (Concentration where flocculation commences) alpha (Settling velocity power coefficient) s1 (Salinity dependence coefficient 1) s2 (Salinity dependence coefficient 2) cgel (Concentration where hindered settling commences)
Flocculation-hindered	As above, with additional effect of hindered settling.	Npar=7 <ul style="list-style-type: none"> ws0 cfloc alpha s1 s2 cgel n (Power coefficient in the hindered settling equation)
vanRijn84	van Rijn's 1984 formulation	None
vanRijn04	Calculated by TRANSPOR (van Rijn, 2004)	None, although requires specification of external model and external model parameters..

1.8.3 Erosion model

The [Erosion model](#) calculates the sediment fraction potential erosion flux (pickup into the water column in $\text{g/m}^2/\text{s}$) at each model cell. The potential erosion flux is subsequently scaled by the individual sediment fraction in order to calculate the actual erosion flux.

Where an erosion model requires a specified critical shear stress for erosion (*taucr*) parameter, this may be directly specified or may optionally be calculated by the [Critical stress model](#). In the latter case, the *tauce* parameter should be entered as a negative factor. The absolute value of this factor is used to scale the *tauce* calculated by the critical stress model.

The available erosion models and associated parameter requirements are detailed in Table 10. Model parameters are assigned using the [Erosion parameters](#) command. Further detail on the technical aspects of each erosion model are provided in Section 1.3.6

Table 10 Erosion models and associated parameter set requirements

Model Names	Model Description	Model Parameter Requirements
None (Default)	$Q_e = 0$.	None
Mehta (also commonly known as Partheniades)	$E = E_r(taub/taucr - 1)^{\alpha}$ Typically used for cohesive sediments but this pickup formulation can also be applied to fine sand.	Npar=3 <ul style="list-style-type: none"> • E_r ($\text{g/m}^2/\text{s}$) • $Taucr$ (N/m^2) • Alpha (Exponent)
vanRijn84	van Rijn (1984) suspended sediment pickup formula. Applicable to sand fractions under combined currents-waves.	Npar=2 <ul style="list-style-type: none"> • E_r • $taucr$
Soulsby_vanRijn	Soulsby-van Rijn suspended load formula. Applicable to sand fractions under combined currents-waves.	None
Bijker	Bijker suspended load pickup flux due to combined waves-currents.	Npar=4 <ul style="list-style-type: none"> • Abs (Default = 5.) • Gbs (Default = 0.4) • Abd (Default = 2.) • Gbd (Default = 0.05)
vanRijn04	Suspended load pickup flux calculated by TRANSPOR.	None, although requires specification of external model and external model parameters .

1.8.4 Deposition model

The [Deposition model](#) is used to calculate suspended sediment deposition fluxes to the bed. Where required model parameters are assigned using the [Deposition parameters](#) command. Further detail on the technical aspects of each deposition model are provided in Section 1.3.5.

Table 11 Deposition models and associated parameter set requirements

Model Names	Model Description	Model Parameter Requirements
None (Default)	No deposition	None
ws0	Typically applied to non-cohesive sediments.	None
Krone	Typically applied to cohesive sediments.	Npar=1 • taucd

1.8.5 Bed load model

The [Bed load model](#) calculates each sediment fraction's potential bedload flux (g/m/s) at each model cell. For simulations including multiple sediment fractions the potential bedload flux is then scaled by the individual sediment proportion in the active bed layer in order to calculate the actual bedload flux (for that fraction). The available bed load models and associated parameter requirements are detailed in Table 12. Where required model parameters are assigned using the [Bed load parameters](#) command.

Where the bed load model depends on a specified *tauce* parameter, this may be directly specified or may be calculated by the [Critical stress model](#). In the latter case, the *tauce* parameter should be entered as a negative factor. The absolute value of this factor is used to scale the *tauce* calculated by the critical stress model. Further detail on the technical aspects of each bed load model are provided in Section 1.3.7.

Table 12 Bed load models and associated parameter set requirements

Model Names	Model Description	Model Parameter Requirements
None (Default)	$Q_b = 0$.	None
MPM	Meyer-Peter-Müller formula. Generally applicable to non-cohesive sediment fractions in steady flows.	Npar=3 • fac • taucr • alpha
MPM_Shimizu	Meyer-Peter-Müller formula with Shimizu bed slope adjustment. Generally applicable to non-cohesive sediment fractions in steady flows.	Npar=3 • fac • taucr • alpha

Model Names	Model Description	Model Parameter Requirements
Soulsby_vanRijn	Non-cohesive wave-current transport, bed load component only.	None
Soulsby_vanRijn_Total	Non-cohesive wave-current transport, total load.	None
Bijker	Bijker formula for non-cohesive sediment transport under waves and currents., bed load component only	Npar=4 <ul style="list-style-type: none"> Abs (Default = 5.) Gbs (Default = 0.4) Abd (Default = 2.) Gbd (Default = 0.05)
Bijker_Total	Bijker formula for non-cohesive sediment transport under waves and currents., total load.	Npar=4 <ul style="list-style-type: none"> Abs (Default = 5.) Gbs (Default = 0.4) Abd (Default = 2.) Gbd (Default = 0.05)
Wilcock_Crowe	Bedload transport of non-cohesive sediment fractions in steady flows.	None
vanRijn04	Bed load flux calculated by TRANSPOR	None, although requires specification of external model and external model parameters..
vanRijn04_Total	Total load flux calculated by TRANSPOR	None, although requires specification of external model and external model parameters..

1.8.6 Critical stress model

The [Critical stress model](#) calculates the critical stress for the threshold of sediment mobilisation, which may then optionally be applied to the [erosion model](#) or [bed load model](#) calculations (1.8.3 and 1.8.5 respectively). The available critical stress models and associated parameter requirements are detailed in Table 13. Where required model parameters are assigned using the [Critical stress parameters](#) command. Further detail on the technical aspects of each critical stress model are provided in Section 1.3.8.

Table 13 Critical stress models and associated parameter set requirements

Model Names	Model Description	Model Parameter Requirements
None (Default)		None
Constant	Critical stress is directly specified.	Npar=1 <ul style="list-style-type: none"> Taucr (N/m²)
Soulsby	Soulsby and Whitehouse formula.	None

Model Names	Model Description	Model Parameter Requirements
Soulsby_Egiazaroff	Soulsby and Whitehouse formula with Egiazaroff sediment fraction shielding.	None

1.8.7 Consolidation model

The [Consolidation model](#) calculates the potential consolidation flux ($\text{g}/\text{m}^2/\text{s}$) at all model bed layers. The consolidation flux will be zero at the bottom most bed layer. For simulations including multiple sediment fractions the potential consolidation flux is subsequently scaled by the individual sediment proportion in the bed layer in order to calculate the actual consolidation flux (for that fraction). Consolidation can either be enabled using the “constant” model or disabled as summarised in Table 14. If using the constant model a consolidate rate in $\text{g}/\text{m}^2/\text{s}$ is specified using the [Consolidation parameters](#) command.

Table 14 Consolidation models and associated parameter set requirements

Model Names	Model Description	Model Parameter Requirements
None (Default)	No consolidation.	None
Constant	Potential consolidation flux is directly specified.	Npar=1 • Qcon ($\text{g}/\text{m}^2/\text{s}$)

1.9 Material Blocks

Material blocks allow for spatial definition of parameter sets within the model domain. The material ids are supplied to the STM by the hydrodynamic model. Unless specified within the material block it is assumed that the [Material](#) group is equal to 1.

A material id of 0 can be used to specify default parameters to be applied globally. These global parameter sets are inherited by material blocks but may then be partially or completely overwritten by further material block specifications. For example, if the domain is characterised by a particle sediment characteristic and bed layer distribution it is recommended that these features be configured and set within the default material 0. Then additional material blocks can be used to overwrite these default values were needed.

By default, the STM will assume a single bed layer, however multiple layers can be specified using the [Nlayer](#) flag. Different Nlayer values can be assigned to different material blocks. Subsequent [Layer](#) sub-blocks should be used to specify layer specific properties. Please note that even if using on one layer, `NLayer == 1` is required to be specified.

Within the material block, sub-blocks may be used to specify Fraction, Layer and Layer-Fraction parameters (see example below). These material level specifications will overwrite any default Fraction or Material parameter sets.

Example default material block:

```
!
-----
Material == 0 ! Default material ID

!Bed layer independant commands
Bed roughness parameters == 0.02, 0.01
Suspended Load Scale == 1.5
Bed Load Scale == 0.5

!Material 0, layer independant commands specific to the "fines" sediment fraction
Fraction == fines
    Deposition parameters == 0.15
End Fraction

!
-----
!Bed layer commands
Nlayer == 2

!
-----
!Material 0, Layer 1 specific commands
Layer == 1
    Dry density == 1590.0, 1590.0
    Initial mass == 159., 159. ! Nsed=2, 50/50 split between fines/sand
End Layer

!-----
!Material 0, Layer 2 specific commands
Layer == 2
    Dry density == 1890.0, 1890.0
    Initial mass == 756., 756. ! Nsed=2, 50/50 split between fines/sand

!Material 0, Layer 2, "fines" specific commands
Fraction == fines
    Erosion parameters == 0.002, 0.15, 1.5
    Bedload parameters == 8.0, 0.15, 1.5
End Fraction

!Material 0, Layer 2, "sand" specific commands
Fraction == sand
    Erosion parameters == 0.001, 0.3, 1.5
    Bedload parameters == 4.0, 0.3, 1.5
End Fraction
End Layer
End Material

!
-----
```

1.9.1.1 Reference height

The suspended sediment reference height, [za](#), may be changed from the default value (0.01 m). Please refer to Section 1.3.10 regarding specification of the sediment reference height.

```
za == 0.05      ! (m)
```

1.9.1.2 Bed roughness

[Bed roughness parameters](#) can be optionally specified within a material block, otherwise the global parameter set will be applied. *Npar* is dependent on the specified bed roughness model.

```
Bed roughness parameters == 0.02, 0.01
```

1.9.1.3 Sediment flux scaling

Optional scaling factors can be applied to the calculated erosion fluxes ([Suspended load scale](#)) and bed load fluxes ([Bed load scale](#)). For example, the scaling factors might be used as a calibration factor or they may be used to switch off a flux component within a material block.

```
Suspended load scale == 0.7 ! Scale the erosion flux
```

```
Bed load scale == 0.0      ! Switch off bed load
```

1.9.2 Sediment fraction properties

[Fraction](#) sub-blocks (within a material block) can be used to specify spatially varying sediment fraction parameters.

```
Material == 1
...
  Fraction == silt                ! Silt fraction sub-block
    Deposition parameters == 0.15 ! Material specific deposition parameters
  End Fraction
...
End Material
```

1.9.3 Bed layer properties

[Layer](#) sub-blocks (within a material block) can be used to specify layer properties. It is good practice to use the default material block (material id = 0) to initially specify properties and then to use additional material blocks to overwrite these defaults only as required.

Note that the [dry density](#) of each layer (including the top layer) must be specified for each sediment fraction within a layer sub-block unless it has been inherited from a default material specification.

Within a layer sub-block there may be nested fraction sub-blocks for modifying fraction specific parameter sets.

An example of layer sub-blocks is provided below.

```
Material == 3, 4, 6
```

```
...
```

```

Nlayer == 2
Layer == 1
    Dry density == 1590.0, 1590.0           !Fines (kg/m3), Sand (kg/m3)
End Layer
Layer == 2
    Dry density == 1890.0, 1890.0           !Fines (kg/m3), Sand (kg/m3)
    Fraction == fines
        Erosion parameters == 0.002, 0.15, 1.5
        Bedload parameters == 8.0, 0.15, 1.5
    End Fraction
    Fraction == sand
        Erosion parameters == 0.001, 0.3, 1.5
        Bedload parameters == 4.0, 0.3, 1.5
    End Fraction
End Layer
...
End Material

```

1.9.4 Initial Sediment Mass Conditions

Material blocks can also be used to initialise the mass composition of each sediment fraction within the discretised bed. An [initial mass](#) flag is used followed by sediment mass quantities in kg/m² for each sediment fraction (Nsed in total).

This type of initialisation would typically be undertaken prior to undertaking one or more bed warmup simulations (refer 1.6.2.1). Where a [Bed restart file](#) (refer 1.6.2) is used then any initial mass specifications will be ignored.

```

Material == 10 ! List of material IDs (up to 10 allowed per block)
...
Layer == 1
    Initial mass == 159., 159. ! Nsed=2, 50/50 split between fines/sand
End Layer
Layer == 2
    Initial mass == 756., 756. ! Nsed=2, 50/50 split between fines/sand
End Layer
...
End Material

```

1.10 STM Outputs

The TUFLOW STM outputs are requested via the controlling HD model. For detail about the output file formats please refer to the Outputs Chapter of the [TUFLOW FV User Manual](#).

1.10.1 Timeseries Output

Suspended sediment concentrations can be output to as timeseries using the .fvc points output block type:

```
Output == points
    Output points file == ..\model\geo\output_points.csv ! x,y, label
    Output parameters == SED_1, SED_2 ! Suspended sediment mg/L
    Output interval == 900. ! s
End output
```

1.10.2 Profile Output

Suspended sediment concentration can be output to as timeseries using the .fvc profile output block type (refer HD manual):

```
Output == profile
    Output points file == ..\model\geo\profile_points.csv ! x,y, label
    Output parameters == V, SAL, SED_1, SED_2 ! Output types
    Output interval == 900. ! s
End output
```

1.10.3 Flux Output

Suspended and bed load can be output at nodestring locations using the flux output block _FLUX.csv (refer HD manual)

```
Output == flux
    Output interval == 900. ! s
End output
```

1.10.4 Mass Output

The global model mass of each sediment fraction in both the seabed and the water column _MASS.csv (refer HD Manual)

```
Output == mass
    Output interval == 900. ! s
End output
```


1.10.5 Map Output Parameters

A list of the available STM output parameters is provided in Table 15. While most STM output parameters are compatible with standard map output formats, some are only compatible with NetCDF formats. For example, multi-dimensional outputs column of Table 15 (refer column ‘Supported Formats’). NetCDF outputs can be plotted via the TUFLOW FV [MATLAB](#) or [Python Toolboxes](#).

```
! This block must reside in the .fvc not the sediment control file
Output == netCDF
    Output Parameters == h,d,v,sal, temp, taub,sed_1,sed_2, ksc, thick
    Output Interval == 300.0           ! (s)
End Output
```

Table 15 Sediment Transport Module output parameters

Output Parameter	Data Type	Description	Supported Formats
Hydraulics			
KSC	Scalar	Current related effective bed roughness (m).	All
KSW	Scalar	Wave related effective bed roughness (m).	All
KSA	Scalar	Apparent bed roughness (m)	All
TAUC	Scalar	Current related effective bed shear stress component (N/m ²)	All
TAUW	Scalar	Wave related effective bed shear stress component (N/m ²)	All
TAUCW	Scalar	Combined effective current/wave bed shear stress (N/m ²)	All
Morphological			
DZB	Scalar	Change in elevation between current output time and model start time (Current map output time ZB – Start ZB m)	All
ZB	Scalar	Bed elevation at current output time (m)	All
THICK	Scalar	Total bed thickness (m)	All
D50_LAYER_ D10_LAYER_ D90_LAYER_#	Scalar	Median (d50) sediment grain size for layer # (m). Also d10 and d90 available outputs.	All
Bed Mass			

Output Parameter	Data Type	Description	Supported Formats
BED_MASS	Array of Scalars Nsedfractions x ncells	Bed mass of each sediment fraction over all bed layers (kg/m ²)	NetCDF only.
BED_MASS_TOTAL	Scalar	Total bed mass (kg/m ²)	All
BED_MASS_LAYER_#	Array of Scalars Nsedfractions x ncells	Bed mass of each sediment fraction over layer # (kg/m ²)	NetCDF only
BED_MASS_LAYER_#_TOTAL	Scalar	Total bed mass of all fractions in layer # (kg/m ²)	All
BED_MASS_LAYER_#_SED_#	Scalar	Bed mass of fraction # in layer # (kg/m ²)	All
Suspended sediment			
SED_#	Scalar	Suspended concentration of sediment fraction # (mg/L)	All
TSS	Scalar	Total suspended solids concentration of all fractions (mg/L)	All
SETRATE	Array of Scalars Nsedfractions x ncells	Settling velocity (m/s)	NetCDF only
DEPOSITION	Array of Scalars Nsedfractions x ncells	Deposition rate of suspended sediment (g/m ² /s)	NetCDF only
DEPOSITION_TOTAL	Scalar	Total deposition rate of suspended sediment (g/m ² /s)	All
PICKUP	Array of Scalars Nsedfractions x ncells	Pickup rate of suspended sediment (g/m ² /s)	NetCDF only
PICKUP_TOTAL	Scalar	Total pick up rate of suspended sediment (g/m ² /s).	All
NETSEDRATE	Array of Scalars	Sum of the deposition and pick up rates (g/m ² /s)	NetCDF only

Output Parameter	Data Type	Description	Supported Formats
	Nsedfractions x ncells		
NETSEDRATE_TOTAL	Scalar	Sum of the total deposition and pick up rates (g/m ² /s)	All
SUSPLOAD_TOTAL	Vector	Total Suspended load (g/m/s)	All
SUSPLOAD_SED_#	Vector	Suspended load of sediment fraction # (g/m/s)	All
Bed Load			
BEDLOAD_TOTAL	Vector	Total Bed load (g/m/s)	All
BEDLOAD_SED_#	Vector	Bed load of sediment fraction # (g/m/s)	All
Combined suspended and bed load			
SEDLOAD_TOTAL	Vector	Total Sediment load (g/m/s)	All
SEDLOAD_SED_#	Vector	Total sediment load sediment fraction # (g/m/s)	All

2 Particle Tracking Module

Chapter Contents

2	Particle Tracking Module	2-1
2.1	Introduction	2-2
2.2	Computational Framework	2-3
2.2.1	Overview	2-3
2.2.2	Lagrangian Scheme	2-4
2.2.3	Eulerian Scheme	2-5
2.2.4	Forcing Scheme	2-5
2.3	Particle Behaviour Options	2-7
2.3.1	Particle Sediment Transport	2-9
2.3.2	Particle motility	2-10
2.4	TUFLOW FV Control File (.FVC) for PTM	2-11
2.4.1	Simulation Configuration	2-11
2.4.2	Offline Mode	2-11
2.4.3	Materials	2-11
2.4.4	Outputs	2-11
2.5	PTM Control File	2-12
2.5.1	Introduction	2-12
2.5.2	Simulation Configuration	2-14
2.5.3	Particle Groups	2-15
2.5.4	Material Blocks	2-19
2.5.5	Particle Seeding	2-20
2.5.6	Open Boundaries	2-23
2.6	Particle Tracking Module Output	2-24
2.6.1	Lagrangian	2-24
2.6.2	Eulerian	2-24
2.6.3	Visualisation libraries	2-25

2.1 Introduction

The TUFLOW Particle Tracking Module (PTM) enables the 2D or 3D simulation of discrete Lagrangian particles as they are transported by a flow field and/or other forcing terms (e.g. wind drift). Particle behaviours such as settling, buoyancy, decay, sedimentation and resuspension can all be simulated.

This tracking of discrete particles can be used to output particle age and fate, which are often useful metrics for environmental applications that are not easily modelled using the Eulerian scheme (a scheme used in TUFLOW FV's other modules such as Advection Dispersion, Sediment Transport and Water Quality). Animal migration, search and rescue, ship navigation and salvage, pest species transport and environmental contaminants are a few of many useful examples of particle tracking.

The PTM is invoked through the HD Engine, which controls the overall simulation, supplies the forcing fields to the PTM and handles certain PTM outputs. This chapter describes the PTM. For information about the relevant TUFLOW HD Engine and its commands please refer to the [TUFLOW FV User Manual](#).

2.2 Computational Framework

2.2.1 Overview

The fundamental assumption that TUFLOW FV uses for its HD Engine, Advection Dispersion (AD), Sediment Transport and Water Quality (WQ) Modules is the use of a Eulerian or mesh-based frame of reference (A Eulerian Scheme). This system discretises the model domain into individual cells that are used to solve the NLSWE equations. At each timestep the solution of these equations provide a single discrete value of a given result at each cell.

In contrast, the PTM uses a Lagrangian Scheme that tracks individual particles through time, their trajectory not subject to being computed on a numerical mesh. Further details on the Lagrangian Scheme is provided in Section 2.2.2.

The PTM and HD Engine (and where applicable AD or WQ modules) work together to enable particle tracking simulation. The PTM requests communication with the HD Engine on a user specified interval provided by the STM command [Eulerian timestep](#) (refer Figure 6). At each [Eulerian timestep](#) a two-way data exchange occurs between the HD Engine and STM whereby updates to particle forcings are supplied to the STM (refer Section 2.2.4) and particle concentration outputs are sent back to the HD Engine.

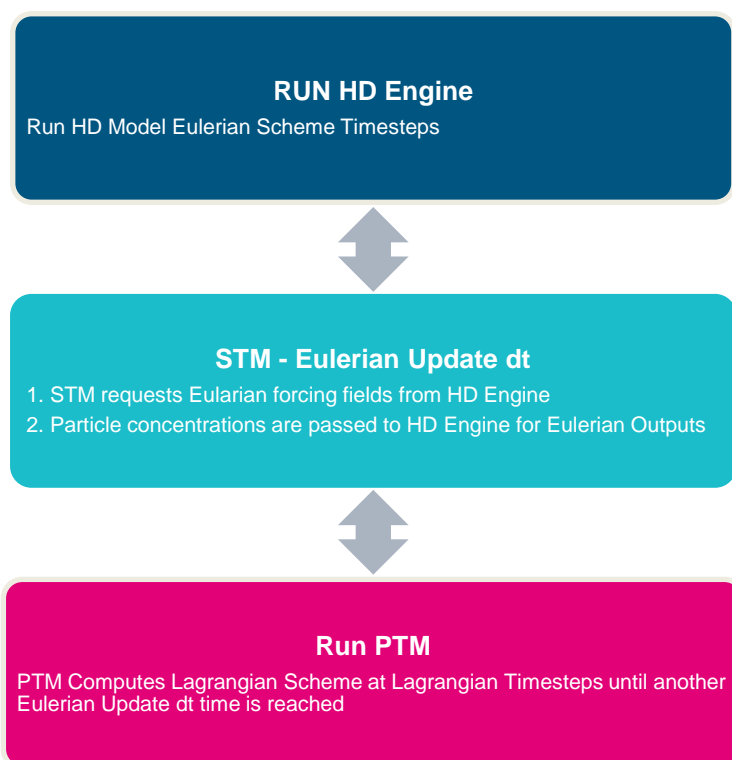


Figure 6 PTM Update dt Sequence

2.2.2 Lagrangian Scheme

The PTM is a Lagrangian model for solving the 3D transport equations for discrete particles. One or more Particle Groups can be specified to control how particles respond to hydrodynamic and other environmental conditions.

The Lagrangian scheme is updated at a specified time interval ([Lagrangian timestep](#)) using a first-order Euler integration. The Lagrangian scheme interacts closely with the TUFLOW FV Eulerian scheme, primarily by interrogating environmental conditions from the HD Engine supplied fields such as velocity, water levels, winds, waves etc on a numerical mesh defined by the HD Engine.

The particle Lagrangian velocity is calculated as the addition of the following vector components:

1. Water column drag (equivalent to the local water velocity; u_0, v_0, w_0)
2. Horizontal random walk component (u', v')
3. Vertical random walk component (w')
4. Particle settling, or buoyancy (w_s)
5. Wind drift (u_{wind}, v_{wind})
6. Wave drift (u_{wave}, v_{wave})
7. Particle motility (u_{mot}, v_{mot} &/or w_{mot})

The horizontal random walk velocity component is scaled by the horizontal diffusivity (ν), according to the following relationship.

$$[u', v'] = R \frac{\sqrt{2\nu * dt}}{dt}$$

where ν is the horizontal diffusivity (m^2/s), dt is the Lagrangian timestep (s) and R is a random number from a standard Gaussian distribution.

The vertical random walk component is scaled by the vertical diffusivity and includes higher-order terms accounting for the vertical concentration gradients.

$$[w'] = \frac{d\nu}{dz} + R \frac{\sqrt{2\nu' * dt}}{dt}$$

where $d\nu/dz$ is the vertical diffusivity gradient at the current location z_0 , and ν' is the vertical diffusivity evaluated at z' . Where z' :

$$[z'] = z_0 + 0.5 \frac{d\nu}{dz} * dt \quad (\text{North et al. 2006}).$$

Both wind drift (u_{wind} , v_{wind}) and wave drift (u_{wave} , v_{wave}) can be applied where wind and wave fields are supplied (refer Section 2.2.4). The calculated drift can be specified to decay linearly with depth below the water surface.

Particle motility, the ability for a particle to move independently of external forces and through it's own means, (u_{mot} , v_{mot} &/or w_{mot}) can be assigned with a range of options available (refer Section 2.3.2).

Fundamentally, individual PTM particles possess “Scalar Mass” which may be specified across one, or more than one scalar constituents. An individual PTM “particle” will typically represent more than one physical particles in the real world. For instance, a PTM particle representing sand-size sediment fractions may have a specified “Scalar Mass” of 1.0 grams, which would in effect be representing approximately 60 million individual sand particles (e.g. “grains”). Specification of the “Scalar Mass” is an important consideration in configuring PTM as it will control the number of discrete PTM particles in the simulation, which will typically involve a compromise between detailed representation and computational efficiency/storage constraints.

2.2.3 Eulerian Scheme

The TUFLOW FV Eulerian Scheme is responsible for the following functions within the PTM:

- Supplying the hydrodynamic model forcing to the Lagrangian particles;
- Calculating additional environmental conditions such as bed shear stress;
- Calculating sediment transport fluxes and parameters:
 - erosion flux, deposition potential, bed load flux
- Calculating cell-averaged concentrations of particles, including bed mass per unit area.

The Eulerian scheme is updated at a specified time interval ([Eulerian timestep](#)). This should be specified based on consideration of temporal gradients in the forcing and integrated particle concentration fields. Typically, the Eulerian scheme timestep would be significantly larger than the [Lagrangian timestep](#).

2.2.4 Forcing Scheme

The forcing fields are supplied to the PTM by the controlling hydrodynamic simulation (e.g. TUFLOW FV) either using the HD Engine calculated hydrodynamics or using an offline mode driving by HD Engine boundary conditions. As the forcing fields are supplied by the HD Engine, they are specified using the same discrete representation, which may be a 2D/3D structured grid or a 2D/3D unstructured mesh.

2.2.4.1 Hydrodynamic Forcing HD Engine

The primary forcing for the PTM is a 2D or 3D hydrodynamic field supplied by the parent HD simulation. For 2D models the vertical profile of velocity can be optionally reconstructed from the

depth-averaged velocity (via the velocity profile model, refer to Table 19) and used to advect particles. The same reconstruction can also be applied to the lowest layer in a 3D hydrodynamic model. Water salinity and temperature and meteorological inputs such as wind and light can optionally be supplied to the PTM by the HD Engine which will allow for further future integration between the PTM with the Water Quality (WQ) Module.

2.2.4.2 Hydrodynamic Forcing Offline Mode

The PTM can optionally be run in ‘offline mode’ which can speed up particle transport simulations by several orders of magnitude. The relevant command in the HD Control file is `Disable Hydros == 1` (refer to [TUFLOW FV User Manual](#) Particle Tracking Chapter or Section 2.4 for relevant .fvc commands). In an offline mode run, the HD Engine hydrodynamic calculations are switched off and water levels, depth and velocities are provided to the PTM via open and gridded boundary conditions supplied to the HD Engine. For example, boundary conditions from 3D global ocean circulation models such as HYCOM can be used to drive the simulation (refer [TUFLOW FV User Manual](#) Boundary Conditions Chapter for further information).

2.2.4.3 Wind Forcing

The PTM may also be forced by the same wind field as the controlling hydrodynamic simulation. Please refer to the hydrodynamic model wind boundary condition section of [TUFLOW FV Manual](#) for more information on configuration. Particle [Wind drift model](#) options are provided in Section 2.5.3.

2.2.4.4 Wave Forcing

The PTM may also be forced by a wave field supplied by the controlling hydrodynamic simulation. Please refer to the hydrodynamic model wave boundary condition section of the [TUFLOW FV Manual](#) for more information on configuration. Particle [Wave drift model](#) options are provided in Section 2.5.3.

2.3 Particle Behaviour Options

Particle group behaviours are primarily specified in a “Group Block” and are therefore uniform across individual group particles. However, selected behaviours e.g. bed roughness, erosion parameters etc. may be linked to parameters that allow for spatial variation (refer Section 2.5.4). An example of this is particle erosion parameters, which may be specified within “Material Blocks”.

The PTM introduces particles to the model domain using one or more [Seed particles](#) blocks. Various options are available for specifying the timing and location of seeded particles (refer Section 2.5.5). Note that particle seeding is specified in terms of either mass or mass flux, and in conjunction with the group [Initial scalar mass](#) specification, will determine the total number of discrete particles entering the simulation.

The particle age since being “seeded” is tracked as the age since it changed state, e.g. between active and in water column, active and dry (on land), active and on the bed.

The behaviours that can be specified at a Particle Group level are provided in Table 16.

Table 16 Lagrangian particle behaviour options.

Particle forcing options	Description	Available options
Horizontal advection	Equal to the reconstructed flow velocity at the particle location. Users can select whether vertical profile reconstruction of the horizontal velocity occurs.	<ul style="list-style-type: none"> No vertical velocity profile reconstruction. Logarithmic profile reconstruction.
Vertical advection	As above for 3D hydrodynamic models only.	N/A
Settling / Buoyancy	Vertical advection speed due to either negative (downwards) or positive (upwards) buoyancy.	<ul style="list-style-type: none"> None Various options (see Table 9)
Horizontal dispersion	Gaussian random walk component that can be applied as a function	<ul style="list-style-type: none"> None Constant diffusivity
Vertical dispersion	Random walk component including higher-order terms to improve accuracy in the case of strong vertical concentration gradients.	<ul style="list-style-type: none"> None Constant diffusivity Parabolic diffusivity HD model diffusivity
Particle motility	Particle advection velocity (potentially 3D) in response to user inputs (fixed direction/magnitude, speed/target, crawl) and/or local environmental conditions interrogated from hydrodynamic model (speed/currents, speed/waves).	<ul style="list-style-type: none"> None Various motility models (refer 2.5.3.1))
Wind drift	Particle drift as a function of wind velocity (where supplied by hydrodynamic model). Will only apply to surface particles with wind drift limits command used to modify the depth to which wind drift can occur..	<ul style="list-style-type: none"> None Proportional to wind velocity plus direction offset

Particle forcing options	Description	Available options
Wave drift	Particle drift as a function of wave field (where supplied by hydrodynamic model).	<ul style="list-style-type: none"> None Stokes drift (depth-averaged) Lagrangian stokes drift (depth-varying)
Erosion (Resuspension)	Pickup of particles located on the bed, into the water column. Pickup flux ($\text{kg}/\text{m}^2/\text{s}$) is calculated using a specified particle erosion model (refer to the Sediment Transport Module section). A stochastic process is used to select individual particles for pickup.	<ul style="list-style-type: none"> None Various options (See sediment transport module Table 10 and Section 2.5.3)
Deposition	Settling to the bed where deposition threshold criteria (if any) are satisfied. The fraction of particles allowed to settle is calculated using specified particle deposition model (refer STM documentation). A stochastic process is used to select and prevent individual particles from depositing.	<ul style="list-style-type: none"> None Various options (See sediment transport module Table 11 and Section 2.5.3)
Bed load transport	Currently under development and testing for particle tracking. Please contact support@tufLOW.com for further information.	

2.3.1 Particle Sediment Transport

The PTM allows for particles to optionally exchange with the bed using routines shared from the TUFLOW Sediment Transport Module (STM). Note this functionality does not require access to the Sediment Transport Module License. The available functionality is summarised in Table 17.

The Sediment Transport routines are fundamentally Eulerian, that is these are calculated primarily on the hydrodynamic model mesh/grid. For advective terms (e.g. settling velocity) the Eulerian values are applied as an additional velocity to the particle advection scheme. Eulerian flux quantities (e.g. erosion) are represented by using a stochastic approach that will randomly select particles for transport until the total mass of particles selected is equivalent to the calculated Eulerian flux.

Table 17 Sediment Transport models available to the PTM

Sediment Transport Models	Calculated quantity	Available options
Settling	Settling velocity (m/s). Positive downwards.	<ul style="list-style-type: none"> None Constant Flocculation Flocculation + Hindered van Rijn 1984 formulae
Bed roughness	Effective bed roughness, k_s (m)	<ul style="list-style-type: none"> Specified k_s Proportional to d_{50}
Bed shear stress	Effective bed shear stress (N/m^2)	<ul style="list-style-type: none"> Default (Soulsby) Bijker
Erosion/Resuspension	Pickup flux ($kg/m^2/s$).	<ul style="list-style-type: none"> None Mehta van Rijn 1984 Soulsby, van Rijn Bijker
Deposition	Fraction of particles allowed to deposit (-).	<ul style="list-style-type: none"> None Unhindered Krone
Bed load transport	Please note that particle bed load transport remains subject to development and testing is not yet enabled. It will be incorporated into future builds of TUFLOW FV.	
Critical shear stress	Bed shear stress at threshold of motion (N/m^2)	<ul style="list-style-type: none"> Constant Soulsby Soulsby_Egiazaroff
External model	The external model is not yet enabled for particle tracking. For more information please contact support@tuflow.com	NA

2.3.2 Particle motility

The PTM supports several options for simulating particle motility, i.e. advective behaviour such as biota swimming behaviour in response to environmental conditions. The motility velocity component is added to other particle velocity components driven by hydrodynamics, wind drag, wave drag, etc. Multiple sequential motility behaviours can be simulated based on specified start times relative to age since particle seeding. A crawling motility can be specified and will be applied when particles are located in dry cells. For further details on particle motility use refer to Section 2.5.3.1.

Table 18 Description of particle motility options

PTM Motility Options	Description
Velocity (horizontal)	Particle swims with specified velocity (u, v specification)
Speed, Waves	Particle swims at specified speed in direction oriented relative to waves
Speed, Currents	Particle swims at specified speed in direction oriented relative to currents
Speed, Target	Particle swims towards a target (X, Y in model coordinate system) at a specified speed.
Crawl, Velocity	Particle crawls at specified velocity when located in dry cell (u, v specification)

2.4 TUFLOW FV Control File (.FVC) for PTM

The HD Engine is responsible for providing the drivers to the PTM. This section summarises the mandatory and optional commands in the TUFLOW FV Control File (.fvc) related to particle tracking. For full details on the HD Engine setup, please refer to the [TUFLOW FV User Manual](#).

2.4.1 Simulation Configuration

To enable particle tracking the following mandatory command is required in the .fvc file as follows:

```
Particle Tracking Control File == particle_control_file.fvptm
```

2.4.2 Offline Mode

If forcing the PTM with hydrodynamics in offline mode (Section 2.4.2) the following command is required to be included in the .fvc. The command can be placed at any location within the .fvc with the exception of bc, material, output, tracer or structure blocks. It is recommended that is placed somewhere near the top of the .fvc with the simulation configuration commands to alert other users of the model that the hydrodynamics are not being actively used.

```
Disable Hydros == 1 ! WL, velocity and depths provided by BCs and not the HD solver
```

2.4.3 Materials

An integer material ID is assigned to each model cell within the .fvc. This same material ID is used by the PTM to apply spatially varying sediment characteristics. For more information on assigning the location and ID of materials please refer to the [TUFLOW FV User Manual](#) Model Geometry Chapter.

2.4.4 Outputs

The HD Engine is responsible for outputting Eulerian results from the PTM using output blocks in the .fvc. Further discussion on PTM Eulerian output is provided Section 2.6.2.

2.5 PTM Control File

2.5.1 Introduction

The particle control file contains the commands required to define particle characteristics and processes. It is called by the HD Engine driver file via the .fvc command particle control file (Section 1.4). The particle control file can be broken down into the following command types as follows:

- Simulation Configuration
- Global Model Specifications
- Particle Group Blocks
 - Particle Motility sub-blocks
- Material Blocks
 - Layer sub-blocks
- Particle Seeding Blocks
- Model Output Blocks

Within each of these broad command types there are a wide range of different options as showcased in Figure 7.

Core to the implementation of the PTM is the concept of *particle groups* and the use of *particle group blocks*. A run can have one or more particle groups. Within a particle group, multiple *Motility sub-blocks* can be applied that allow for specialised user-specified particle motion.

Commands are applied in a cascading manner. For example, simulation and global model commands are applied to all particle groups being simulated. The particle group blocks that follow provide independent, flexible control over how each particle type is simulated. *Material blocks* allow for the assignment of spatial variability in sediment parameters. Additionally, *layer sub-blocks* that reside within material blocks can be used to vary sediment quantities and properties spatially (please note that the STM has only a single bed layer).

One or more *particle seeding blocks* can be used to control the release of particles into the model with respect to release location, time and particle type.

Lagrangian particle outputs can be specified within a PTM *model output block*.

Each of the command types and their options are detailed within Sections 2-142.5.2 to 2.6 that follow. These sections link heavily with Appendix B which details required syntax and example syntax use.

Simulation Configuration

Time commands
 Eulerian and Lagrangian time stepping
 Particle mass discretisation
 Depth limits
 Open boundaries
 Particle restart commands

Global Model Specifications

Concentration profile model
 Bed roughness model
 Bed shear model

Particle Group Blocks

Group ==
 Particle properties (d50, density)
 Initial mass discretisation
 Velocity profile model and parameters
 Horizontal dispersion model and parameters
 Vertical dispersion model and parameters
 Wind drift model and parameters
 Wave drift model and parameters
 Settling model and parameters
 Erosion model and parameters
 Deposition model and parameters

Motility model == (Motility sub-block)

Motility time commands
 Motility parameters
 End motility

End Group

Material Blocks

Material ==
 Material specific roughness parameters
 Suspended and bed load scales

Layer == 1 (Only one bed layer currently enabled)

Dry density

Group == (example group sub-block inside layer sub-block)

Fraction parameters specific to material and layer

End group

End Layer

End material

Particle Seeding Blocks

Seed particles == (seeding method)
 Particle groups
 Group mass
 Temporal commands
 Vertical release commands
 End seed

Model Output Blocks

Output == ptm_netcdf
 Output groups
 Output interval
 Output parameters
 End output

Figure 7 TUFLOW FV Particle Tracking Control File Overview

2.5.2 Simulation Configuration

The following sub-sections describe globally applied control file commands and parameters.

2.5.2.1 NScalar

The number of scalar mass constituents ([NScalar](#)) attached to each particle is globally applied across all particle groups. Unless specified using the following, the number of scalars defaults to 1. Scalar mass constituents can be thought of as separate mass ‘containers’ that reside on a particle. The total mass of a given particle is comprised of the sum of NScalar mass contributions.

```
Nscalar == 5
```

2.5.2.2 Memory Allocation

To avoid memory overallocation the particle tracking module is limited to 10,000,000 particles per simulation (the default). This can be overwritten using the [Maximum particles](#) command. Please note that the total number of particles that can be run will remain limited by the available memory on the modelling computer. Preliminary hardware benchmarking has indicated a memory usage of 0.2-0.4 GB of memory is required for every million particles, however this will vary from model to model. Therefore, model specific testing is recommended if memory usage is of concern.

2.5.2.3 Timestep commands

The PTM simulation timesteps require specification (in seconds) of both the [Lagrangian timestep](#) and [Eulerian timestep](#) timesteps. The Lagrangian timestep controls the accuracy of particle track integration. The Eulerian timestep will control the frequency with which certain environmental parameters (e.g. bed shear stress) are updated and will also control the frequency at which particle concentration fields are calculated.

```
Lagrangian timestep == 60.           ! (s)
Eulerian timestep == 600.           ! (s)
```

2.5.2.4 Sediment Transport commands (Global)

Analogous to the setup of the sediment transport control file, particle movement functionality can be specified globally and/or applied to individual particle groups. Those Parameters which can set as a function of material in the STM can also be defined spatially within a [Material](#) Block of the PTM.

```
Bed roughness model == ks           ! Nikuradse roughness
Bed roughness parameters == 0.01, 0.005 ! ksc (m), ksw (m)
Bed shear model == default          ! Soulsby
```

2.5.2.5 Depth limit commands (Global)

A set of global depth limits may be applied to control particle erosion and deposition behaviour in shallow cells using the [erosion depth limits](#), [depositional depth limits](#) and [wave depth limits](#) commands, analogous to the sediment transport module (refer to Section 1.6.1.3).

```
Erosion depth limits == 0.05, 0.10           ! (m)
Deposition depth limits == 0.01, 0.05       ! (m)
Wave depth limits == 0.05, 0.15            ! (m)
```

2.5.2.6 Restart File

Restart files may be output as part of a PTM simulation and can be used to initialise subsequent simulations. A [particle restart file](#) is always output upon successful completion of a simulation and may optionally be written at a specified frequency during the simulation using the [write restart dt](#) command. The restart file can also be optionally overwritten at the restart dt using the [restart overwrite](#) command.

Using a restart file as an initial condition requires provision of a directory path to an existing restart file:

```
particle restart file == log\PreviousRun_ptm.rst
```

2.5.3 Particle Groups

Particle [group](#) blocks define the common behaviours and attributes of a group of particles. The block must define the [initial scalar mass](#) of individual particles from the specific group when they are seeded. Sediment transport formulae require information about the [Particle density](#) and also particle size ([d50](#)) which must be specified.

The particle behaviour within a group is defined by various models and associated parameter sets. For example, the random walk component of particle horizontal velocity will be defined by the [Horizontal dispersion model](#) and associated [Horizontal dispersion parameters](#). Table 19 summarises the available models and associated parameter set requirements within the PTM [Group](#).

An example [Group](#) block is provided below:

```
Group == name
  Initial scalar mass == 1.,1.,1.,1.,1.      ! mass in grams, This example: Nscalar==5
  Particle density == 2650.                  ! density in kg/m^3
  d50 == 0.0002                             ! median diameter in m
  ...
  Other commands defining particle group behaviour, e.g.
  ...
  Horizontal dispersion model == constant    ! Constant Huz
  Horizontal dispersion parameters == 5.0    ! Dispersion coefficient (m^2/s)
  Vertical dispersion model == HD           ! Vertical dispersion via HD model
  Vertical dispersion parameters == 1.0     ! Vertical dispersion scale factor
  Wind drift model == simple
  Wind drift parameters == 0.04, -20.
```

```

Wind drift depth limits == 1, 2      !m below water surface
Wave drift model == stokes
Wave drift parameters == 1.0, 0.
Velocity profile model == logarithmic
Settling model == constant
Settling parameters == 1.0e-2
Deposition model == krone
Deposition parameters == 0.1
Erosion model == Mehta
Erosion parameters == 0.001, 0.2, 1.5

```

End Group

Table 19 PTM Group available models and associated parameter set requirements

Model category	Model Names	Model Description	Model Parameter Requirements
Velocity profile model	None <default>	Use discrete velocity profile from HD model with no reconstruction.	None
	Logarithmic	Reconstruct logarithmic profile in lowest HD model layer.	None
Horizontal dispersion model	None <default>	No horizontal random walk component.	None
	Constant	Directly specify horizontal diffusivity, Nuh.	Npar = 1 • Nuh [m ² /s] (refer Section 2.2.2)
Vertical dispersion model	None <default>	No vertical random walk component.	None
	Constant	Directly specify constant vertical diffusivity.	Npar = 1 • Nuz [m ² /s] (refer Section 2.2.2)
	Parabolic	Parabolic diffusivity distribution in lower-half of water-column, constant in top-half. nuz = kappa*ustar*zp*(1-zp/h) for zp<0.5*h nuz = 0.25*kappa*ustar*h for zp(>0.5*h)	Npar = 1 • Kappa [-]
	HD Model	Scaled vertical diffusivity from HD model.	Npar = 1 • Fac [-]
Wind drift model	None <default>	Uwind_drift = 0.	None
	Simple	Wind drift velocity proportional to 10mASL wind speed, with positive clockwise direction offset.	Npar = 2 • Fac [-] • Dir_offset [deg +ve clockwise]
	None <default>	Uwave_drift = 0.	None

Model category	Model Names	Model Description	Model Parameter Requirements
Wave drift model	Stokes	Wave drift velocity proportional to depth-averaged (Eulerian) Stokes drift, with +ve clockwise direction offset. $U_{wave_drift} = Fac * Q_{stokes} / h$ Where Q_{stokes} is the depth-averaged Stokes drift mass flux calculated from linear wave theory.	$N_{par} = 2$ <ul style="list-style-type: none"> Fac [-] Dir_offset [deg +ve clockwise]
	Stokes_Lagrangian	Wave drift velocity proportional to the depth-dependent Lagrangian Stokes velocity, with +ve clockwise direction offset. $U_{wave_drift} = Fac * U_{stokes}(z)$ Where $U_{stokes}(z)$ is the depth-dependent Lagrangian Stokes drift velocity calculated from linear wave theory.	$N_{par} = 2$ <ul style="list-style-type: none"> Fac [-] Dir_offset [deg +ve clockwise]
Settling model	None <default>	Neutrally buoyant particle, $W_s=0$.	None
	Constant	Directly specify w_s (+ve down).	$N_{par} = 1$ <ul style="list-style-type: none"> W_s [m/s]
For more advanced particle settling model options please refer to the Sediment Transport Module, Section 1.8.2, noting the setup for particles and sediment are analogous.			
Deposition model	None <default>	No deposition allowed. $Frac_deposit = 0$.	None
	W_s0	Unhindered deposition flux. $Frac_deposit = 1$.	None
For more advanced particle deposition model options please refer to the Sediment Transport Module, Section 1.8.4, noting the setup for particles and sediment are analogous.			
Erosion model (Resuspension)	None <default>	No erosion allowed. $Q_{pickup} = 0$.	None
	For more advanced particle erosion model options please refer to the Sediment Transport Module, Section 1.8.3, noting the setup for particles and sediment are analogous.		
Bed load model	None <default>	$Q_{bedload} = 0$.	None
	Bed load transport currently in development and testing for particle tracking.		

2.5.3.1 Particle Motility

Multiple particle motility behaviours can be specified in [Motility model](#) sub-blocks via. [Motility parameters](#) . An optional [Motility start time](#) (in hours) may be specified and is measured relative to the age of individual particles since seeding to allow for motility to be delayed. Where multiple motility blocks are specified for a particle group they are applied in sequence. That is, when a motility model starts the previous one (if any) ends.

The available motility models and associated parameter requirements are described in Table 20.

Example of motility model sub-blocks:

```
Group == name
  Other group block commands
  ...
  Motility model == Velocity
    Motility Parameters == -1.3, 0.5
  End motility
  Motility model == Speed, Waves
    Motility Parameters == 0.7, 0.0, 0.2, 120.
    Motility Start Time == 2.0 ! Optional start time in Hrs
  End motility
  ...
End Group
```

Table 20 Motility models and associated parameter set requirements

Motility model	Model Description	Model Parameter Requirements
None <default>	No particle motility applied.	None
Velocity	Specified horizontal velocity; u,v [m/s]	Npar = 2 <ul style="list-style-type: none"> • u [m/s] • v [m/s]
Speed, Waves	Specified horizontal speed; u_bar and direction offset relative to wave propagation.	Npar = 4 <ul style="list-style-type: none"> • Spd [m/s] • Dir_offset [degrees +ve clockwise from wave propagation direction] • Hsmin [m] • Default direction (when Hs<Hsmin)
Speed, Currents	Specified horizontal speed; u_bar and direction offset relative to currents.	Npar = 4 <ul style="list-style-type: none"> • Spd [m/s] • Dir_offset [degrees +ve clockwise from current ation direction] • U0min [m/s] • Default direction (when u0<u0min)
Speed, Target	Specified horizontal speed; u_ba towards target coordinate.	Npar = 3 <ul style="list-style-type: none"> • Spd [m/s] • X [model coordinates]

Motility model	Model Description	Model Parameter Requirements
		<ul style="list-style-type: none"> Y [model coordinates]
Crawl, Velocity	Specified horizontal velocity; u,v [m/s] only activated when particle located within dry cell.	Npar = 2 <ul style="list-style-type: none"> u [m/s] v [m/s]

2.5.4 Material Blocks

[Material](#) blocks allow for spatial definition of certain parameter sets. The material ids (numbers) are supplied to the PTM by the hydrodynamic model. Unless specified within the material block it is assumed that the [Material group](#) is equal to 1.

A material id of 0 is used to specify parameters to be applied globally, prior to overwriting with parameter sets for non-zero material ids.

Unlike the STM which support multiple bed layers, the PTM currently only supports a single active layer and has no allowances for consolidation. Within a [Material](#) block, a [Layer](#) sub-block is still required to be defined using `Layer == 1` and terminating the layer sub-block with an `End Layer` command.

Within a layer sub-block, nested [Group](#) sub-blocks can be used to allow spatial variation of particle parameters by overwriting any default Group or Material parameters that have been previously specified.

An example material block is provided as follows:

```
Material == 1,4,6 ! List of material IDs (up to 10 allowed per block)
  Material group == 2 ! Optional material group ID (default is 1)
  Bed roughness parameters == 0.02, 0.01
  Suspended Load Scale == 1.5
  Group == grp1, grp2
    Deposition parameters == 0.15
  End Group
  Layer == 1
    Dry density == 1590.0
    Group == grp1
      Erosion parameters == 0.002, 0.15, 1.5
    End Group
    Group == grp2
      Erosion parameters == 0.001, 0.3, 1.5
    End Group
  End Layer
End Material
```

2.5.5 Particle Seeding

Particle seeding blocks are used to control the spatial and temporal seeding of particles within the model domain. The block starts with a [Seed particles](#) flag followed by a seed type specification which references how the particle seeding is spatially defined. One or more [Particle groups](#) can be specified in the block.

```
Seed particles == seed_type
  Particle groups == grp1, grp3
  ...
  Commands defining seeding location, vertical distribution and mass flux
  ...
End seed
```

2.5.5.1 Particle Release

The [total number of particles](#) released into the model for each particle group for given [Seed particles](#) block is calculated using one of the following two methods:

1. Using the [Initial scalar mass](#) of the particle group and the [Group mass](#) assigned in the seed particles block as follows:

$$\text{Group Mass (g) / Initial Scalar Mass (g) = Number of Particles}$$

2. Using the [Initial scalar mass](#) of the particle group and the mass flux timeseries (refer to Section 2.5.5.8) assigned in the seed particles block as follows:

$$\text{Timeseries Mass (g/s) / Initial Scalar Mass (g) = Number of Particles / s}$$

For both cases, where $N_{\text{Scalar}} > 1$, the [Initial scalar mass](#) used to calculate particle release counts is the summation of all mass [Initial scalar mass](#) values for the particle group.

There are four options [for assigning the time period](#) over which particles will be released:

1. If no time commands are entered in the [Seed particles](#) block, all particles will be released on the first timestep of the model simulation.
2. If a [Start time](#) is specified in a [Seed particles](#) block but no [End time](#), all particles will be released at the [Start time](#).
3. If both a [Start time](#) and [End time](#) are specified, the total number of particles will be released at an even rate between the [Start time](#) and [End time](#).
4. Using a mass flux [Timeseries file](#).

Please note that Options 1-3 above require the use of the Group mass command, whilst Option 4 does not.

The sections that follow provide an example for each of the available seed particles options.

2.5.5.2 Point source

A simple seeding option is to specify a point source as per the following example. Following the seed type a set of x, y (and optional z) coordinates are specified in the model projection. If no z-coordinate is specified, then this is assumed to be 0m (depth convention). A [Group mass](#) of 1.0e+6 (in units of grams) is also specified in this example. With no specification related to timing, it is assumed that all particles are seeded during model initialisation.

```
Seed particles == point, 156.01, -27.50, -100.  
  Particle groups == grp1  
  Group mass == 1.0e+6  
End Seed
```

2.5.5.3 Polygon source

A polygon (region) can be specified that will distribute randomly seeded particles to all cells within the polygon. The [Polygon file](#) should be a .csv format with column headers X, Y and subsequent rows specifying the polygon vertices. By default, all particles will be seeded at the water surface. If particles need to be seeded as a function of depth this can be completed by including vertical distribution commands to the seed particles block (refer to Section 2.5.5.7).

```
Seed particles == polygon  
  Particle groups == grp1, grp2  
  Polygon file == polygon_file.csv  
  Group mass == 1.0e+6, 1.0e+6  
End Seed
```

2.5.5.4 Polyline source

A polyline can be specified to randomly seed individual particles along a polyline. The [Polyline file](#) should be a .csv format with column headers X, Y and subsequent rows specifying the polyline vertices. By default, all particles will be seeded at the water surface. If particles need to be seeded as a function of depth this can be completed by including vertical distribution commands to the seed particles block (refer to Section 2.5.5.7).

```
Seed particles == polyline  
  Start time == 01/01/2008 05:00  
  End time == 01/01/2008 07:00  
  Particle groups == grp3  
  Polyline file == polyline_file.csv  
  Group mass == 1.0e+6  
End Seed
```

2.5.5.5 XYZ source

The xyz option uses a [xyz file](#) specifying coordinates with column headers X, Y, Z, or with user specified headers using the [xyz header](#) command. This file is sampled row by row as each individual particle is seeded. If necessary, this file will be looped through multiple times.

```
Seed particles == xyz
```



```
Particle groups == all
xyz file == xyz_file.csv
Group mass == 1.0e+6
End Seed
```

2.5.5.6 Moving point source

A moving point source can also be specified using a mass flux timeseries file, which must also include X, Y coordinates in the timeseries (refer to Section 2.5.5.8 for more detail on the specification of a timeseries file).

```
Seed particles == moving_point
Particle groups == grp1, grp2
Timeseries file == moving_mass_flux_ts.csv
Timeseries header == Time, Lon, Lat, Flux_Grp1, Flux_Grp2
End Seed
```

2.5.5.7 Vertical distribution

The vertical distribution of particle seeding can also be specified using [Vertical coordinate type](#) and [Vertical distribution file](#) command. The default vertical coordinate type is “depth”, also available is “height”, “elevation” and “sigma”. These vertical coordinate types are analogous to those used in the HD Engine. There are also examples of depth averaging methods on the [TUFLOW FV Wiki](#).

```
Seed particles == point, 156.01, -27.50
Particle groups == grp1
Group mass == 1.0e+6
Vertical coordinate type == depth
Vertical distribution file == vert_distr.csv
End Seed
```

2.5.5.8 Mass flux timeseries

A more flexible option for specifying the time variation of particle release is to use a [Timeseries file](#) (csv format) which defines the particle release as a timeseries in units of grams per second. The [Timeseries header](#) flag can be used to specify which columns are used from the csv file. Note: Group mass is not required when using a timeseries file as the mass of particles is provided directly by the timeseries file.

```
Seed particles == point, 156.01, -27.50
Particle groups == grp1, grp2
Timeseries file == mass_flux_ts.csv
Timeseries header == Time, Flux_Grp1, Flux_Grp2
End Seed
```

2.5.6 Open Boundaries

The default PTM behaviour is that all boundaries are treated as closed, unless otherwise specified. Using this default behaviour particles will bounce off a closed boundary and therefore remain within the model domain. The [open boundary nodestring](#) can be used where particles can be free to leave the model domain, for example to flow upstream or downstream. The nodestrings should already be defined within the hydrodynamic model setup. Please note that once a particle has left the model domain the same particle cannot re-enter the model. An example command is:

```
Open boundary nodestring == 4
```

2.6 Particle Tracking Module Output

The PTM supports the output of both Lagrangian and Eulerian results. The Lagrangian outputs are handled directly by the PTM whilst Eulerian based particle concentrations are passed back to the HD Engine for output.

2.6.1 Lagrangian

The PTM Lagrangian output is fundamentally a set of x, y, z coordinates of particles output at regular timesteps. The particle groups, frequency of output and additional Lagrangian parameters (e.g. velocity, mass, particle age etc.) can be controlled within a Lagrangian output block.

Lagrangian output is stored in a compact NetCDF file format. As the number of particles being simulated can be very large, care needs to be taken with Lagrangian output specifications to avoid creating excessive storage demands. An optional [file size limit](#) can be used as a safeguard against erroneous file saving, if for example too many particles have been released due to incorrect setup.

Lagrangian results are saved to a location defined via the [output dir](#) command.

```
Output dir == ..\results
```

PTM outputs are specified using [output](#) block commands.

```
Output == ptm_netcdf
  Output groups == all
  Output interval == 900.
  Output parameters == uvw
End output
```

The ptm_netcdf output files are written in a compact NETCDF format and fundamentally define the time history of particle x, y and z coordinates. Unless specified, all output groups will be included in the file. A status variable defines whether a particle is yet to be activated (-1), active in the water column (+1), active on dry land (+2), active on the bed (+3), or has left the domain (-9). All ptm_netcdf output parameters are based on a Lagrangian definition (i.e. are defined at the individual particle locations). Available particle tracking output parameters are provided in Table 21.

2.6.2 Eulerian

Eulerian outputs from the PTM are the mass concentrations of particles within cells defined by the calling HD Engine. Note that if the user requires Eulerian output it needs to be specified via output blocks in the HD Engine .fvc file. For example, PTM_1, PTM_N, PTM_BED_1 and PTM_BED_N are the Eulerian particle concentration for particle group 1 in suspension, particle group N in suspension, particle group 1 in the bed and particle group N in the bed respectively. For full information on setting Eulerian outputs please refer to the Outputs Chapter of the [TUFLOW FV User Manual](#).

```
output == netcdf
  output parameters == h,v,d, temp, sal, Rhow, PTM_1, PTM_2, PTM_BED_1, PTM_BED_N
  output interval == 300.
```

end output

2.6.3 Visualisation libraries

There is a suite of MATLAB and Python visualisation libraries available to assist in the viewing of ptm_netcdf outputs. For Python libraries please refer to:

https://fvwiki.tuflow.com/index.php?title=FV_Python_Tools#TUFLOW_FV_Python_Package.

Table 21 Particle Tracking Output Parameters

Output Parameter	Data Type	Description	Supported Formats
Eulerian			
PTM_#	Scalar	Suspended particle concentration (mg/L).	All
PTM_BED_#	Scalar	Mass of particle in bed (kg/m ²).	All
Lagrangian ¹			
stat	Scalar	Particle state. A status variable defines whether a particle is yet to be activated (-1), active in the water column (+1), active on dry land (+2), active on the bed (+3), or has left the domain (-9).	NetCDF Only
uvw	3D Vector	3D particle velocity (m/s)	NetCDF Only
uvw_water	3D Vector	3D water velocity (m/s)	NetCDF Only
diff	3D Vector	3D particle diffusivity (m ² /s)	NetCDF Only
depth	Scalar	Depth to particle (m)	NetCDF Only
water_depth	Scalar	Total water depth (m)	NetCDF Only
age	Scalar	Age since release (hrs)	NetCDF Only
state_age	Scalar	Age in current state (hrs). States being either active in the water column, dry land or bed.	NetCDF Only
mass	Data Vector (NScalar x numParticles)	Mass of each particle with NScalar contribution if applicable.	NetCDF Only

¹ X, Y and Z particle coordinates are output by default and are always present in ptm_netcdf outputs.

3 References

- Bijker, E. W. (1967). Some considerations about scales for coastal models with moveable bed. *Publ. 50*, Delft Hydraulics Lab.
- Bijker, E.W. (1971). Longshore transport computation. *Journal of Waterways, Harbors and Coastal Engineering Division, Proceedings of ASCE*, New York, WW4, pp687-701.
- Egiazaroff, I. V. (1965). Calculation of nonuniform sediment concentrations. *Journal of the Hydraulics Division*, Vol.91, Issue 4, pp225–247.
- Garcia, M. and Parker, G. (1991). Entrainment of bed sediment into suspension. *Journal of Hydraulic Engineering, ASCE*, Vol. 117, No. 4, pp414-35.
- Meyer-Peter, E. and Müller, R. (1948). Formulas for bed-load transport. *Proceedings of 2nd meeting of the International Association for Hydraulic Structures Research*, Delft, pp39-64.
- North, E.W. Hood, R.R. Chao, S.-Y. Sanford, L.P.. (2006). Using a random displacement model to simulate turbulent particle motion in a baroclinic frontal zone: A new implementation scheme and model performance tests, *Journal of Marine Systems*, Volume 60, Issues 3–4, 2006, Pages 365-380.
- Shimizu, Y., Watanabe, Y. and Toyabe, T. (1995). Finite amplitude bed topography in straight and meandering rivers. *JSCE Journal of Hydraulic, Coastal and Environmental Engineering*, No.509/II-30, pp67-78. (in Japanese)
- Soulsby, R. (1997). Dynamics of marine sands. Thomas Telford Publications, London
- Soulsby, R. L. and Whitehouse, R. J. S. (1997). Threshold of sediment motion in coastal environments. Proceedings of the 13th Australasian Coastal and Ocean Engineering Conference and the 6th Australasian Port and Harbour Conference, Christchurch, Vol.1, pp149–154.
- Van Rijn, L.C. (1984a). Sediment Transport, Part I: Bed Load Transport. *Journal of Hydraulic Engineering, ASCE*, Vol. 110, No. 10, pp1431-56.
- Van Rijn, L.C. (1984b). Sediment Transport, Part II: Suspended Load Transport. *Journal of Hydraulic Engineering, ASCE*, Vol. 110, No. 11, pp1613-41
- Van Rijn, L.C. (1984c). Sediment Transport, Part III: Bed Forms and Alluvial Roughness. *Journal of Hydraulic Engineering, ASCE*, Vol. 110, No. 12. pp1733-54.
- Van Rijn, L. C., Walstra, D. J. R., and Van Ormondt, M. (2004). Description of TRANSPOR 2004 (TR2004) and implementation in DELFT3D-online. *Rep. Z3748*, Delft Hydraulics, Delft, The Netherlands.
- Van Rijn, L.C. (2007a), United view of sediment transport by currents and waves I: Initiation of motion, Bed roughness and bed load transport. *Journal of Hydraulic Engineering, ASCE*, Vol 133, No.6, pp649-667

Van Rijn, L.C. (2007b). United view of sediment transport by currents and waves II: Suspended transport, *Journal of Hydraulic Engineering, ASCE*, Vol. 133, No. 6, pp668-689

Van Rijn, L.C. (2007c). United view of sediment transport by currents and waves III: Graded beds. *Journal of Hydraulic Engineering, ASCE*, Vol. 133, No. 7, pp761-775

Van Rijn, L.C. (2007d). United view of sediment transport by currents and waves IV: Application of morphodynamic model. *Journal of Hydraulic Engineering, ASCE*, Vol. 133, No. 7, pp776-793

Wilcock, P. R. and Crowe, J. C. (2003). Surface-based Transport Model for Mixed-Size Sediment, *Journal of Hydraulic Engineering, ASCE*, Vol. 129, Issue 2, pp120-128.

Appendix A fvsed File Commands

SEDIMENT TRANSPORT COMMANDS

[Armor layer thickness](#) [Initial sediment concentration](#)

[Bedload model](#) [Initial mass](#)

[Bedload parameters](#) [Layer](#)

[Bed load scale](#) [Material](#)

[Bed restart file](#) [Morphological coupling](#)

[Bed roughness coupling](#) [Morfac](#)

[Bed roughness model](#) [NLayer](#)

[Bed roughness parameters](#) [Non erodible mass](#)

[Bed shear model](#) [Particle density](#)

[Bed slumping model](#) [Restart overwrite](#)

[Bed slumping parameters](#) [Sediment control file](#)

[Concentration profile model](#) [Settling model](#)

[Consolidation model](#) [Settling parameters](#)

[Consolidation parameters](#) [Start time](#)

[Critical stress model](#) [Suspended load scale](#)

[Critical stress parameters](#) [Update dt](#)

[d50](#) [Wave depth limits](#)

[Deposition depth limits](#) [Wave gamma](#)

[Deposition model](#) [Write restart dt](#)

[Deposition parameters](#) [za](#)

[Display sediment update](#)

[Dry density](#)

[Erosion depth limits](#)

[Erosion model](#)

[Erosion parameters](#)

[External model](#)

[External model parameters](#)

[Fraction](#)

[Include sediment](#)

Armour layer thickness == <amin, amax (m)>

(Optional; Default == 0,9999.)

Optional sediment simulation configuration command to allow for active layer bed armouring in sediment transport calculations. Allows bed sorting and active layer bed armouring, typically required for high energy river environments.

For example, to turn off bed armouring:

```
Armour layer thickness == 0.00, 9999. !switch off bed armouring
```

For example, to maintain a constant armour layer thickness:

```
Armour layer thickness == 0.2, 0.2. !constant armour layer of 0.2m.
```

For further details and examples please refer to Section 1.6.1.4.

Bed load model == <None, MPM, MPM_Shimizu, Soulsby_vanRijn, Soulsby_vanRijn_Total, Bijker, Bijker_Total, Wilcock-Crowe, vanRjin04, vanRjin04_Total>

(Optional; Default == None)

Optional sediment fraction block command to select one of a range of common bed load transport modules. Used in combination with the [Bed load parameters](#) command.

For example, to use the Meyer-Peter Muller bed load equation:

```
Bed load model == MPM
```

```
Bed load parameters == 8.0, 0.040, 1.5 !fac, taucr, alpha.
```

Noting the [Bed load parameters](#) assigned may vary in number and requirement as a function of the selected bed load model.

For further information please refer to Section 1.8.5 and Table 12 for a description of available options.

Bed load parameters == <blparams>

(Optional; Default == None)

Optional list of bed load model parameters input as a comma delimited list. The number and type of parameters are a function of the selected [Bed load model](#). Please refer to the [Bed load model](#) command. For further information please refer to Section 1.8.5 and Table 12 for a description of available options.

Bed load scale == <bscale>

(Optional, Default == 1)

Material block command. Optional scaling factor for bed load fluxes. May be used as a calibration factor or to switch off bed load fluxes for a given material.

For example, to switch off bed load fluxes:

```
Bed load scale == 0.0 ! Switch off bed load
```

For example, to increase bed load fluxes by 20%:

```
Bed load scale == 1.2 ! Increase bed load fluxes by 20%
```

Please note that material specific parameters can be applied via a [material](#) block (refer Section 1.9).

See also Suspended load scale and Section 1.9.1.3.

```
Bed restart file == <bed restart file name (_bed.rst)>
```

(Optional)

Optional command to load the simulation initial conditions from a bed restart file `_bed.rst` generated by a previous TUFLOW FV simulation. A bed restart file is always output upon successful completion of a simulation and may optionally be written at a specified frequency during the simulation using the [write restart dt](#) command. The restart file can also be optionally overwritten at the restart dt using the [restart overwrite](#) command.

Refer Section 1.6.2 for further information.

```
Bed roughness coupling == <0;1>
```

(Optional; Default == 0)

Optional command to pass apparent roughness height (ksa) calculated by the sediment transport module back to the hydrodynamic model to calculate bed friction terms.

Please refer to Section 1.7.1.1 for further information. See also [Bed roughness model](#).

```
Bed roughness model == <ks, d50, VanRijn>
```

(Mandatory; No Default)

Required global model specification command to select the bed roughness formulation used for sediment calculations. Used in combination with the [Bed roughness parameters](#) command.

For example, to use the ks model:

```
Bed roughness model == ks
```

```
Bed load parameters == 0.01,0.01 ! ksc, ksw
```

Noting the [Bed load parameters](#) assigned may vary in number and requirements as a function of the selected bed roughness model.

For further information please refer to Section 1.7.1.1 and Table 5 for a description of available options. See also [Bed roughness coupling](#).

```
Bed roughness parameters == <brparams>
```

(Mandatory; No Default)

Optional list of bed roughness model parameters input as a comma delimited list. The number and type of parameters are a function of the selected [Bed roughness model](#). For further information please refer to Section 1.8.5 and Table 5 for a description of available options.

Please note that material specific parameters can be applied via a [material](#) block (refer Section 1.9).

Bed shear model == <default, Bijker, vanRijn>

(Optional; Default == default)

Optional global model specification command to select the effective bed shear formulation used for sediment calculations.

For example, to use the Bijker model:

```
Bed shear model == Bijker !Bijker's method - combined wave/currents
```

For further information please refer to Section 1.7.1.2 and

Table 6 for a description of available options.

Bed slumping model == <None, Simple>

(Optional; Default == None)

Optional global model specification command to enable bed slumping during sediment calculations. Used in combination with the [Bed slumping parameters](#) command.

For example, to use the simple 'angle of repose' model:

```
Bed slumping model == Simple
```

```
Bed slumping parameters == 0.36, 0.36 ! lim. slope wet, lim. slope dry
```

Noting the [Bed slumping parameters](#) assigned may vary in number and requirements as a function of the selected bed slumping model.

For further information please refer to Section 1.7.1.3 and Table 7 for a description of available options.

Bed slumping parameters == <slope_wet, slope_dry>

(Optional; No Default)

Optional list of bed slumping model parameters input as a comma delimited list. Please note slope should be entered as a value from 0 to 1. For example:

A limiting slope of 20 degrees = $\text{Tan}(20/180*\pi) = 0.36$.

The number and type of parameters are a function of the selected [Bed slumping model](#). For further information please refer to Section 1.7.1.3 and Table 7 for a description of available options.

Please note that material specific parameters can be applied via a [material](#) block (refer Section 1.9).

Concentration profile model == <'Default', 'First-order', 'Legacy'>

(Optional; Default == Default)

Optional global model specification command to select the concentration profile model used for sediment calculations. Please note that the default is the recommended concentration profile model and should be used unless there are specific circumstances which require backwards compatibility.

The concentration profile model defines how sediment concentrations are estimated as a function of water depth by the STM. By default, a higher-order [Concentration profile model](#) reconstruction is applied within all HD model cells and for all sediment fractions as follows:

```
Concentration profile model == default
```

There are no parameters required to be input for the concentration model.

Model Names	Model Description	Model Parameter Requirements
Default	Higher-order reconstruction applied throughout water column.	None
First-order	Reconstruction only applied at bed interface.	None
Legacy	No concentration profile reconstruction.	None

Consolidation model == <None, Constant>

(Optional; Default == None)

Optional [sediment fraction block](#) command to enable bed consolidation. Used in combination with the [Consolidation parameters](#) command.

For example, to use the Constant consolidation option:

```
Consolidation model == Constant
```

```
Consolidation parameters == 0.01 !Consolidation flux g/m^2/s
```

Noting the [Consolidation parameters](#) assigned may vary in number and requirements as a function of the selected bed load model.

For further information please refer to Section 1.8.7 and Table 14 for a description of available options.

Consolidation parameters == <0;1, 0;1>

(Optional; Default == 0)

Optional list of consolidation model parameters input as a comma delimited list. The number and type of parameters are a function of the selected Consolidation model. For further information please refer to Section 1.8.7 and Table 14 for a description of available options.

Critical stress model == <None, Constant, Soulsby, Soulsby_Egiazaroff>

(Optional; Default == None)

Optional [sediment fraction block](#) command to calculate the critical stress for inception of sediment mobilisation. Used in combination with the Critical stress parameters command. For example, to use the Constant consolidation option:

```
Critical stress model == Constant
```

```
Critical stress parameters == 0.2 !Critical shear stress for motion N/m^2
```

Noting the [Critical stress parameters](#) assigned may vary in number and requirements as a function of the selected bed load model. For further information please refer to Section 1.8.6 and Table 13 for a description of available options.

Coupling the [Critical stress model](#) with the [Erosion mode](#) and/or [Bed load model](#) commands.

The calculated critical shear stress may optionally be applied to the [Erosion model](#) or [Bed load model](#) calculations (for more on coupling the critical shear stress with these models please refer to Sections 1.8.3 and 1.8.5 respectively).

Critical stress parameters == <taucr>

(Optional; Default == 0)

Optional list of consolidation model parameters input as a comma delimited list. The number and type of parameters are a function of the selected [Critical stress model](#). For further information please refer to Section 1.8.6 and Table 13 for a description of available options.

d50 == <median sieve diameter (m)>

(Mandatory, Default == 0.0)

Required [sediment fraction block](#) command to assign the median grain size of the sediment fraction.

```
d50 == 0.0002 ! [m]
```

For further information please refer to Section 1.8.1. See also [Fraction](#) and [Particle density](#).

Deposition depth limits == <limit1 (m), limit2 (m)>

(Optional, Default == 0.01, 0.05)

[Global simulation configuration command](#). The calculated deposition rate is limited to 0 if water depth is less than *limit1*. Between, *limit1* and *limit2*, the deposition rate is linearly scaled down. This is used to avoid unreasonable deposition in cells with small depths. This is particularly applicable at the wet-dry interface. Please note this will affect both bed load transport and bed pickup into the water column. For example:

```
Deposition depth limits == 0.01, 0.05
```

See also [Erosion depth limits](#), [Wave depth limits](#) and Section 1.6.1.3.

Deposition model == <None, ws0, Krone>

(Optional; Default == None)

Optional sediment fraction block command to select one of a range of common deposition models. Used in combination with the [Deposition parameters](#) command.

For example, to use the Krone deposition equation:

```
Deposition model == Krone
```

```
Deposition parameters == 0.1 !taudc N/m^2
```

Noting the [Deposition parameters](#) assigned may vary in number and requirements as a function of the selected deposition model. For further information please refer to Section 1.8.4 and Table 11 for a description of available options.

Deposition parameters == < tau dc >

(Optional; Default == 0.0)

Optional list of deposition model parameters input as a comma delimited list. The number and type of parameters are a function of the selected [Deposition model](#). For further information please refer to Section 1.8.4 and Table 11 for a description of available options.

Display sediment update == <0;1>

(Optional; Default == 1)

Optional command to enable (1) or disable (0) the screen reporting of the sediment module update.

Dry density == <bed dry density (kg/m³)>

(Mandatory, Default == 0.0)

Bed layer property that defines the dry density of the bed layer. For example:

```
Dry density == 1590.0, 1590.0 ! Dry density (kg/m^3) of each fraction in bed layer.
```

Please refer to Section 1.9.2. See also [Material](#), [Layer](#), [Nlayer](#).

Erosion depth limits == <limit1 (m), limit2 (m)>

(Optional, Default == 0.01, 0.05)

Global simulation configuration command. The calculated erosion rate is limited to 0 if water depth is less than *limit1*. Between, *limit1* and *limit2*, the erosion rate is linearly scaled down. This is used to avoid unreasonable erosion in cells with small depths. This is particularly applicable at the wet-dry interface. Please note this will affect both bed load transport and bed pickup into the water column. For example:

```
Erosion depth limits == 0.01, 0.05
```

See also [Deposition depth limits](#), [Wave depth limits](#) and Section 1.6.1.3.

Erosion model == <None, Mehta, vanRijn84 , Soulsby_vanRijn, Bijker, vanRijn04>

(Optional; Default == None)

Optional sediment fraction block command to select one of a range of common erosion models. Used in combination with the Erosion parameters command.

For example, to use the Mehta erosion model:

```
Erosion model == Mehta
```

```
Erosion parameters == 0.1, 0.2, 1.0 !Er, taucr, alpha
```

Noting the Erosion parameters assigned may vary in number and requirements as a function of the selected deposition model. For further information please refer to Section 1.8.3 and Table 10 for a description of available options.

Please note: For selected erosion models it is possible to source a calculated *taucr* via the Critical stress model. For example, to use the Mehta erosion model with a critical stress model *taucr*:

```
Erosion model == Mehta
```

```
Erosion parameters == 0.1, -1, 1.0 !Er, taucr scale factor, alpha
```

Erosion parameters == <tauce>

(Optional; Default == 0.0)

Optional list of deposition model parameters input as a comma delimited list. The number and type of parameters are a function of the selected Erosion model. For further information please refer to Section 1.8.3 and Table 10 for a description of available options.

External model == <None, vanRijn04>

(Optional; Default == None)

Optional global model specification command to use an external sediment transport model. For example:

```
external model == vanrijn04 ! vanRijn TRANSPOR model (2004)
```

```
external model parameters == 0.00012, 0.0002, 0.00035, 9.0, 3.0, 1.0, 1.0, 1.0, 1.0, 1.0 !d10, d50, d90, ur, bf_type, f_ws, f_tauc, f_current_efficiency, f_wave_efficiency, f_wave_assymetry, f_wave_streaming
```

Currently, vanRijn's TRANSPOR model is available but this can be extended if you have a specific external model of preference. If you would like to include an external model please contact support@tufLOW.com for further information.

For further information please refer to Section 1.7.1.6 and Table 8 for a description of available options.

External model parameters == <emp>

(Optional; Default == 0)

Optional list of external model parameters input as a comma delimited list. The number and type of parameters are a function of the selected [External model](#). For further information please refer to Section 1.7.1.6 and Table 8 for a description of available options.

Fraction == <fraction_name>

...
...
...

End Fraction

(Mandatory; No Default)

Optional command to include sediment transport calculations:

This command indicates the beginning of a fraction block, specifying unique sediment fraction properties. Fraction properties and fraction specific models are listed in the following rows and the 'end fraction' command is used to indicate the end of the fraction block.

Commands that can be assigned in a sediment fraction block include:

- [d50](#)
- [Particle density](#)
- [Settling model](#) and associated [Settling parameters](#)
- [Erosion model](#) and associated [Erosion parameters](#)
- [Deposition model](#) and associated [Deposition parameters](#)
- [Bed load model](#) and associated [Bed load parameters](#)
- [Critical stress model](#) and associated [Critical stress parameters](#)
- [Consolidation model](#) and associated [Consolidation parameters](#)

Example Fraction Block

```
!Sediment Fraction Commands

fraction == silt

    d50 == 0.000010          ! 10 micron

    particle density == 2650.

    settling model == constant

    settling parameters == 1.0E-03

    erosion model == Mehta

    erosion parameters == 0.1, 0.2, 1.0

    deposition model == Krone

    deposition parameters == 0.18

    bed load model == none

    critical stress model == none
```

```
consolidation model == none
```

```
end fraction
```

Please refer to Section 1.8 for further information on sediment fraction block commands and their required parameters.

```
Include sediment == <0;1, 0;1>
```

(Optional; Default == 0,0)

TUFLOW FV Control File (fvc) command to include sediment transport calculations:

- 0 = False (i.e. sediment transport is not modelled).
- 1 = True (i.e. sediment transport is modelled).

The second flag specifies whether density is a function of the modelled sediment concentration:

- 0 = False (i.e. water density is not affected by sediment concentration).
- 1 = True (i.e. water density is affected by sediment concentration).

```
Initial sediment concentration == <Sed_1, ..., Sed_Nsed>
```

(Optional)

Globally sets the initial suspended sediment concentration for each sediment fraction.

```
Initial mass == <mass (kg/m2)>
```

(Mandatory; Default == 0.0)

[Layer](#) block command used to assign quantities in kg/m² for each sediment fraction (Nsed in total). For example:

```
Initial mass == 159., 159. ! Nsed=2, 50/50 split between fines/sand
```

Please refer to Section 1.9.4. See also [Material](#), [Layer](#), [Non erodible mass](#), [Bed restart file](#).

```
Layer == <bed layer #>
```

```
...
```

```
...
```

```
...
```

```
End Layer
```

(Mandatory)

[Material](#) sub-block to vary sediment properties as a function of bed layer. This command indicates the beginning of a layer block, specifying unique sediment bed properties for cells within bed layer #. Layer properties are listed in the following rows and the 'end material' command is used to indicate the end of the material block.

By default, the STM will assume a single bed layer, however multiple layers can be specified using the [Nlayer](#) flag. Different Nlayer values can be assigned to different material blocks. Subsequent [Layer](#) sub-blocks should be used to specify layer specific properties including:

- [Dry density](#)
- [Initial mass](#)
- [Fraction](#)
- [Non erodible mass](#)

An example of layer sub-blocks is provided below.

```
Material == 3, 4, 6
...
  Nlayer == 2
  Layer == 1
    Dry density == 1590.0
  End Layer
  Layer == 2
    Dry density == 1890.0
    Fraction == fines
      Erosion parameters == 0.002, 0.15, 1.5
      Bedload parameters == 8.0, 0.15, 1.5
    End Fraction
    Fraction == sand
      Erosion parameters == 0.001, 0.3, 1.5
      Bedload parameters == 4.0, 0.3, 1.5
    End Fraction
  End Layer
...
End Material
```

For more information and examples on Layer sub-blocks please refer to Sections 1.9 and 1.9.3. See also [Material](#).

Material == <material id #>

...
...
...

End Material

(Mandatory)

This command indicates the beginning of a material block, specifying unique sediment bed properties for cells with material id #. Material properties are listed in the following rows and the 'end material' command is used to indicate the end of the material block.

Material properties related to sediment transport model include:

-
- [Bed load scale](#)
 - [Bed roughness parameters](#)
 - [Fraction](#)
 - [Layer](#)
 - [Suspended load scale](#)
 - [Nlayer](#)
 - [za](#)

For more information and examples on Material blocks please refer to Section 1.9.
See also [Layer](#), [Fraction](#).

Morphological coupling == <0;1>

(Optional; Default == 0)

Optional simulation configuration command to specify whether morphology is changed as the result of sediment transport. Set to 1 to enable morphological coupling, 0 to disable. For example:

```
Morphological Coupling == 1
```

Please refer to Section 1.6.1.2 for more information. See also [Morfac](#).

Morfac == <morphological flux factor>

(Optional, Default == 1.)

A [Morfac](#) (>1.0) can be specified in order to accelerate the evolution of bed mass composition and morphology. This might be undertaken in order to warm up the bed for a subsequent simulation or may be undertaken in order to represent a longer period of evolution than the actual simulation time.

```
Morfac == 12.0
```

Please refer to Section 1.6.1.2 for more information. See also [Morphological coupling](#).

Nlayer == <number of sediment bed layers>

(Optional, Default == 1)

By default, the STM will assume a single bed layer, however multiple layers can be specified using the [Nlayer](#) flag. Different Nlayer values can be assigned to different material blocks. Subsequent [Layer](#) sub-blocks should be used to specify layer specific properties.

Please refer to Section 1.9 for further information. See also [Material](#), [Layer](#).

Non erodible mass == <mass (kg/m²)>

(Optional; Default == 0.0)

[Layer](#) block command used to assign quantities in kg/m² for each sediment fraction (Nsed in total) that cannot be eroded from the layer. For example:

```
Non erodible mass == 20., 20. ! Nsed=2, 50/50 split between fines/sand
```

A non-erodible sediment mass can be specified within a layer in order to represent immobile (presumably coarse) sediment fractions. Note that a non-zero non-erodible mass would mean that a layer can never be completely eroded and consequently any underlying layers would not be activated. For multi-layer models a better approach would be to explicitly resolve these coarse sediment fractions.

```
Material == 10 ! List of material IDs (up to 10 allowed per block)
...
  Layer == 1
    Initial mass == 159., 159. ! Nsed=2, 50/50 split between fines/sand
  End Layer
  Layer == 2
    Non erodible mass == 9999., 9999. ! Nsed=2, Allow for bedrock.
  End Layer
...
End Material
```

See also [Material](#), [Layer](#), [Initial mass](#), [Bed restart file](#).

Particle density == <density ((kg/m³)>

Required [sediment fraction block](#) command to assign the density of the sediment fraction.

```
Particle density == 2650. ! [kg/m^3]
```

For further information please refer to Section 1.8.1. See also [Fraction](#) and [d50](#).

Restart overwrite == <0;1>

(Optional; Default == 1)

Simulation configuration command option to overwrite the [Bed restart file](#) at the time interval specified using the [Write restart dt](#) command (default) or create a series of restart files for each timestep:

- 0 = False (i.e. the restart file will not be overwritten, and a series of restart files will be generated).
- 1 = True (i.e. the single restart file will be overwritten).

Please refer to Section 1.6.2. See also [Write restart dt](#), [restart overwrite](#), [Bed restart file](#), [use restart file time](#).

Sediment control file == <file path>

(Mandatory, No Default)

TUFLOW FV Control File (.fvc) command that paths the sediment control file. Required for simulations containing sediment transport calculations.

Settling model == <None, Constant, Flocculation, Flocculation-hindered, vanRijn84, vanRijn04>

(Optional; Default == None)

Optional [sediment fraction block](#) command to select one of a range of common settling models. Used in combination with the [Settling parameters](#) command.

For example, to use the Constant deposition equation:

```
Deposition model == Constant
```

```
Deposition parameters == 0.1 !use constant settling velocity (m/s)
```

Noting the [Settling parameters](#) assigned may vary in number and requirements as a function of the selected deposition model. For further information please refer to Section 1.8.2 and Table 9 for a description of available options.

Settling parameters == <spar>

(Optional; Default == 0.0)

Optional list of deposition model parameters input as a comma delimited list. The number and type of parameters are a function of the selected [Settling model](#). For further information please refer to Section 1.8.2 and Table 9 for a description of available options.

Start time == <sediment start time (hrs)>

(Optional; Default == HD Start Time)

An optional [Start time](#) can be specified if the STM simulation should commence after the HD simulation has warmed up for a period. Please note the time format needs to be consistent with the overarching HD simulation. Specifies the start time for the simulation:

- For HD model where: `Time Format == Hours` ! Units are decimal hours
- For HD model where: `Time Format == ISODATE` ! dd/mm/yyyy HH:MM:SS.

Please refer to Section 1.6.1.1. See also [Update dt](#).

Suspended load scale == <sscale>

(Optional, Default == 1)

Material block command. Optional scaling factor for suspended load fluxes. May be used as a calibration factor or to switch off suspended load fluxes for a given material.

For example, to switch off suspended sediment fluxes:

```
Suspended load scale == 0.0 ! Switch off suspended load
```

For example, to increase increase suspended load fluxes by 20%:

```
Suspended load scale == 1.2 ! Increase suspended load fluxes by 20%
```

Please note that material specific parameters can be applied via a [material](#) block (refer Section 1.9).

See also [Bed load scale](#) and Section 1.9.1.3.

Update dt == <time (s)>

(Mandatory, Default == HD timestep)

Optional command to specify the timestep for updating the sediment transport module. If not specified, this will occur at every timestep. For example:

```
Update dt == 600. ! s
```

Please refer to Section 1.6.1.1.

Wave depth limits == < limit1 (m), limit2 (m)>

(Optional; Default == 0.01,0.05)

This optional command can be used to depth limit wave height in shallow cells, which may result in reduced wave related bed shear stress and sediment transport. The wave height is limited to 0 if the water depth is less than *limit1*. Between, *limit1* and *limit2*, the wave height is linearly scaled down. This is used to control excessive erosion in cells with small depths and where the wave model inputs may be overly coarse. For example:

```
Wave depth limits == 0.05, 0.15 ! m
```

See also [Deposition depth limits](#), [Erosion depth limits](#), [Wave gamma](#) and Section 1.6.1.3.

Wave gamma == <wgamma>

(Optional; Default == 0.8)

This optional command can be used to depth limit wave height in shallow cells. *Wgamma* is defined as the maximum limiting ratio between wave height and water depth. Unlike the [Wave depth limits](#) command the wave gamma command only reduces wave height proportional to water depth and does not reduce to 0 at a finite depth.. For example:

```
Wave gamma == 0.6 ! maximum allowable ratio of wave height to water depth
```

See also [Wave depth limits](#).

Write restart dt == <bed restart log writing interval (hrs)>

(Optional)

Writes a bed restart file (.rst) to the log directory location at the time interval specified. The restart file is binary format and contains the spatially varying conserved variables at an instant in time.

A restart file is used to specify the initial condition for subsequent TUFLOW FV simulations using the [bed restart file](#) command.

za == <reference height (m)>

(Optional, Default == 0.01)

[Material](#) block command to set the suspended sediment reference height. For example:

`za == 0.05 ! Suspended sediment reference height (m)`

Refer Section 1.9.1.1 for more information.

Appendix B fvptm File Commands

PARTICLE TRACKING COMMANDS

[End time](#)

[Eulerian timestep](#)

[File size limit](#)

[Final output](#)

[Group](#)

[Group mass](#)

[Horizontal dispersion model](#)

[Horizontal dispersion parameters](#)

[Initial scalar mass](#)

[Lagrangian timestep](#)

[Layer](#)

[Material](#)

[Maximum particles](#)

[Motility model](#)

[Motility parameters](#)

[Motility start time](#)

[NScalar](#)

[Open boundary nodestring](#)

[Output](#)

[Output dir](#)

[Particle density](#)

[Particle groups](#)

[Particle restart file](#)

[Polygon file](#)

[Polyline file](#)

[Reference time](#)

[Restart overwrite](#)

[Seed particles](#)

[Start output](#)

[Start time](#)

[Time units](#)

[Timeseries file](#)

[Timeseries header](#)

[Velocity profile model](#)

[Vertical dispersion model](#)

[Vertical dispersion parameters](#)

[Wave drift model](#)

[Wave drift parameters](#)

[Wind drift depth limits](#)

[Wind drift model](#)

[Wind drift parameters](#)

[Write restart dt](#)

[XYZ file](#)

[XYZ header](#)

End time == <end time of particle release>

(Optional; No default)

Optional [Seed particles](#) block command to control the temporal release of particles. The time convention and units should be consistent with the hydrodynamic model; in the following example ISODATE convention is being used. The particles will be released at a steady rate between the Start time and End time. For example:

```
Seed particles == point, 156.01, -27.50
  Particle groups == grp1
  Group mass == 1.0e+6
  Start time == 01/01/2008 05:00
  End time == 01/01/2008 07:00
End Seed
```

For more information please refer to Section 2.5.5.1. See also: [Start time](#), [Seed particles](#).

Eulerian timestep == <Eulerian scheme dt (s)>

(Optional; Default == HD timestep)

Optional command to specify the timestep for updating the Eulerian scheme calculations. If not specified, this will occur at every timestep. Typically, the Eulerian scheme timestep would be significantly larger than the Lagrangian timestep. For instance, a typical Eulerian timestep might be 600 seconds (ie, 10 minutes):

```
Eulerian timestep == 600. ! s
```

Please refer to Section 2.2.3. See also [Lagrangian timestep](#).

Final output == <time>

(Optional)

Optional particle [output](#) block command to specify the final time for an output Lagrangian NetCDF output request. The time format must be consistent with the simulation time format (i.e. HOURS or ISODATE). If not specified, the output final time will be consistent with the simulation end time.

Please refer to Section 2.6. See also [Output](#).

File size limit == <maximum Lagrangian output filesize (GB)>

(Optional; Default == 10 GB)

Optional particle [output](#) block command to limit the maximum output file size for Lagrangian NetCDF output. This command aims to avoid the erroneous creation of very large file sizes. To modify the default limit the user can modify the file size using the command as follows:

```
File size limit == 20. ! GB
```

Please refer to Section 2.6. See also [Output](#).

Group == <group_name>

...

...

...

End Group

(Mandatory; No Default)

This command indicates the beginning of a particle group block, specifying unique particle group properties and models. The 'end group' command is used to indicate the end of the group block.

Commands that can be assigned in a particle group block include:

- [d50](#)
- [Particle density](#)
- [Horizontal dispersion model](#) and [Horizontal dispersion parameters](#)
- [Vertical dispersion model](#) and [Vertical dispersion parameters](#)
- [Wind drift model](#) and [Wind drift parameters](#)
- [Wave drift model](#) and [Wave drift parameters](#)
- [Velocity profile model](#)
- [Settling model](#) and associated [Settling parameters](#)
- [Erosion model](#) and associated [Erosion parameters](#)
- [Deposition model](#) and associated [Deposition parameters](#)
- [Critical stress model](#) and associated [Critical stress parameters](#)
- [Motility model](#)

Example Fraction Block:

```
!Example Particle Group Block
Group == name
  Initial scalar mass == 1.,1.,1.,1.,1. ! mass in grams, Nscalar=5
  Particle density == 2650. ! density in kg/m^3
  d50 == 0.0002 ! median diameter in m
  ...
  Other commands defining particle group behaviour, e.g. Motility etc.
etc.
...
Horizontal dispersion model == constant
Horizontal dispersion parameters == 5.0
Vertical dispersion model == HD
Vertical dispersion parameters == 1.0
Wind drift model == simple
Wind drift parameters == 0.04, -20.
Wind drift depth limits ==
Wave drift model == stokes
Wave drift parameters == 1.0, 0.
Velocity profile model == logarithmic
Settling model == constant
Settling parameters == 1.0e-2
Deposition model == krone
Deposition parameters == 0.1
```

```
Erosion model == Mehta
Erosion parameters == 0.001, 0.2, 1.5
End Group
```

Please refer to Section 2.5.3 and Table 19 for further information on the available particle group models and their parameters.

```
Group mass == < grp1_mass, .... ,grpn_mass (g)>
```

(Optional; Default == 0.0)

Optional [Seed particles](#) command that is used in combination with the [Initial scalar mass](#) command to determine the number of particles to release for a given particle group. This is calculated for each respective particle group by simply dividing the group mass by the initial scalar mass as follows:

Group mass / Initial scalar mass = Number of particles for release.

Where multiple groups are specified the group mass should be entered as a comma delimited list. For example:

```
Seed particles == polygon
Particle groups == grp1, grp2
Polygon file == polygon_file.csv
Group mass == 1.0e+6, 1.0e+6
End Seed
```

Please refer to Section 2.5.5 for further information on particle seeding. See also [Seed particles](#), [Particle groups](#), [Initial scalar mass](#).

```
Horizontal dispersion model == <None, Constant>
```

(Optional; Default == None)

Optional [particle group](#) command to specify the particle dispersion model. 'None' will not apply any additional turbulent dispersion behaviour and constant will allow you to specify a fixed value for the particle group in m²/s via the [Horizontal dispersion parameters](#) command.

For example, to use the constant model

```
Horizontal dispersion model == Constant ! Constant horizontal diffusivity
Horizontal dispersion parameters == 0.2 ! Nuh m^2/s
```

For further information please refer to Section 2.5.3 and Table 19 for a description of available options.

```
Horizontal dispersion parameters == < Nuh (m2/s) >
```

(Optional; Default == 0.0)

Optional list of horizontal dispersion parameters input as a comma delimited list. The number and type of parameters are a function of the selected [Horizontal dispersion model](#).

For further information please refer to Section 2.5.3 and Table 19 for a description of available options.

Initial scalar mass == < NScalar1_mass, ... , NScalarN_mass (g) >

(Mandatory; Default == 0.0)

Required particle group command to specify the mass of individual particles. Masses are provided as a comma delimited list of length [NScalar](#). For example:

```
Initial scalar mass == 1.,1.,1.    ! mass in grams, This example Nscalar==3
```

This initial scalar mass is used in conjunction with the [Seed particles](#) block determine the number of particles released into the model which are typically applied using the [Group mass](#) or [Timeseries file](#) commands.

Please refer to Section 2.5.3 and 2.5.5 for more information.

Lagrangian timestep == <lts (s) >

(Mandatory; Default == HD timestep)

Lagrangian scheme update timestep that controls the accuracy of the particle track integration. For example:

```
Lagrangian timestep == 60.          ! (s)
```

Please refer to Section 2.2.3. See also [Eulerian timestep](#).

Layer == <bed layer #>

...
...
...

End Layer

(Mandatory)

[Material](#) sub-block command to vary particle properties as a function of bed layer. This command indicates the beginning of a layer sub-block. Layer properties are listed in rows with the 'end layer' command used to indicate the end of the layer block.

Please note that the particle tracking module only supports a single bed layer using the [Nlayer](#) flag (unlike the STM that can have multiple bed layers).

[Layer](#) sub-blocks should be used to specify layer specific properties including:

- [Dry density](#) (analogous to STM)
- [Group](#)

An example of layer sub-blocks nested within a [Material](#) block is provided as follows:

```
Material == 1,4,6 ! List of material IDs (up to 10 allowed per block)
```

...

Other commands defining particle group behaviour, e.g.

...

```
Layer == 1
```

```

Dry density == 1590.0
Group == grp1
    Erosion parameters == 0.002, 0.15, 1.5
End Group
Group == grp2
    Erosion parameters == 0.001, 0.3, 1.5
End Group
End Layer
End Material

```

For more information and examples on Layer sub-blocks please refer to Section 2.5.4. See also [Material](#).

Material == <material id #>

```

...
...
...

```

End Material

(Mandatory)

This command indicates the beginning of a material block, specifying unique particle bed properties for cells with material id #. Material properties are listed in the following rows and the 'end material' command is used to indicate the end of the material block.

Material properties related to sediment transport model include:

- [Bed roughness parameters](#)
- [Group](#)
- [Layer](#)
- [Nlayer](#) (Nlayer == 1 for particle tracking)

An example [Material](#) block is provided as follows:

```

Material == 1,4,6 ! List of material IDs (up to 10 allowed per block)
    Bed roughness parameters == 0.02, 0.01
    Group == grp1, grp2
        Deposition parameters == 0.15
    End Group
    Nlayer == 1
    Layer == 1
        Dry density == 1590.0
        Group == grp1
            Erosion parameters == 0.002, 0.15, 1.5
        End Group
        Group == grp2
            Erosion parameters == 0.001, 0.3, 1.5
        End Group
    End Layer
End Material

```

For more information and examples on Material blocks please refer to Section 2.5.4. See also [Layer](#).

Maximum particles == <max_particles>

(Optional; Default == 10,000,000)

Optional simulation configuration command to set an upper limit to the number of particles permitted to be seeded for a given run. Aimed at avoiding accidental memory allocation if a user has erroneously specified particle seeding. If you require to track more than the default value this can be specified as follows:

```
Maximum particles == 20,000,000      ! Increase max particles to 20
million.
```

Please refer to Section 2.5.2.2 for more information.

Motility model == <None, 'Speed,Waves', 'Speed,Currents', 'Speed,Target', 'Crawl, Velocity'>

(Optional; Default == None)

Optional particle group sub-block to assign one-or-more additional methods or movement. These may be applied directly to a particle or force a particle to react to environmental stimuli.

[Motility parameters](#) are applied within each motility sub-block. An optional [Motility start time](#) (in hours) may be specified and is measured relative to the age of individual particles since seeding. Where multiple motility blocks are specified for a particle group they are applied in sequence. That is, when a motility model starts the previous one (if any) ends. An example motility sub-block nested within a particle group block is provided as follows:

```
Group == name
  Other group block commands
  ...
  Motility model == Velocity
    Motility Parameters == -1.3, 0.5
  End motility
  Motility model == Speed, Waves
    Motility Parameters == 0.7, 0.0, 0.2, 120.
    Motility Start Time == 2.0 ! Optional start time in hrs
  End motility
  ...
End Group
```

Please refer to Section 2.5.3.10 for more information. The available motility models and associated parameter requirements are described in Table 20. See also: [Motility parameters](#), [Motility start time](#), [Particle group](#).

```
Motility parameters == < motility_param_#, ... ,  
motility_param_N>
```

```
(Optional; Default == 0.0)
```

Optional list of motility parameters input as a comma delimited list. The number and type of parameters are a function of the selected [Motility model](#).

Please refer to Section 0 for more information. The available motility models and associated parameter requirements are described in Table 20. See also: [Motility model](#), [Motility start time](#), Particle [group](#).

```
Motility start time == <motility_start_time (hrs)>
```

```
(Optional; Default == 0,0)
```

Optional motility sub-block command to specify the start time in hours that a given particle motility behaviour should commence. The motility start time is measured relative to the age of individual particles since seeding.

Note: Where multiple motility blocks are specified for a particle group they are applied in sequence. That is, when a motility model starts the previous one (if any) ends.

Please refer to Section 2.5.3.1 for more information. See also: [Motility model](#), [Motility parameters](#), Particle [group](#).

```
NScalar == <number of scalar constituents>
```

```
(Optional; Default == 1)
```

The number of scalar mass constituents ([NScalar](#)) attached to each particle is globally applied across all particle groups. Unless user specified, the number of scalars defaults to 1. Scalar mass constituents can be thought of as separate mass ‘containers’ that reside on a particle. The total mass of a given particle is comprised of the sum of NScalar mass contributions.

```
Nscalar == 1
```

```
Open boundary nodelist == <ns_id>
```

```
(Optional; No Default)
```

Optional command to allow particles to leave the model domain via a nodelist ID (`ns_id`). The nodelists should already be defined within the hydrodynamic model setup. The default PTM behaviour is that all boundaries are treated as closed, unless otherwise specified. Particles will bounce off a closed boundary and therefore remain within the model domain. For example, to allow particles to pass from the model via nodelist 4 (as defined in the hydrodynamic model):

```
Open boundary nodelist == 4
```

Output == <ptm_netcdf>

...
...
...

End Output

(Optional; No default)

Optional particle tracking control file command. Specifies the beginning of an output block to specify and save Lagrangian model results that are saved to a location defined via the [output dir](#) command.

```
Output dir == ..\results
```

The ptm_netcdf output files are written in a compact NETCDF format and fundamentally define the time history of particle x, y and z coordinates.. Available particle tracking output parameters are provided in Table 21. For example the follow PTM output block will output all available Lagrangian outputs:

```
! Lagrangian outputs set within the PTM Control File
Output == ptm_netcdf
  Output groups == all
  Output interval == 900.
  Output parameters == uvw
End output
```

Note that Eulerian output (e.g. particle group concentrations) is handled by the TUFLOW hydrodynamic model. For example, Eulerian outputs are specified to your hydrodynamic control file (.fvc) as follows:

```
! Eulerian outputs et within the FVC

output == netcdf

  output parameters == h,v,d, temp, sal, Rhow, PTM_1, PTM_2, PTM_BED_N,
PTM_BED_N

  output interval == 300.

end output
```

To restrict the period to time to output results the [Start output](#) and/or [Final output](#) command can be used within an output block. For example:

```
Output == ptm_netcdf
  Output groups == all
  Output interval == 900.
  Output parameters == uvw
  Start output == 02/04/2018 02:00:00
End output
```

Please refer Section 2.6 for further information on specifying result outputs.

Output dir == <>

(Optional;)

Optional particle tracking control file command to specify the output directory of Lagrangian results. Please refer to Section 2.6 for more information. See also: [Output](#).

Particle density == <density (kg/m³)>

(Mandatory, Default == 0.0)

Particle property that defines the density of the particle group. For example:

```
Particle density == 2560.0      ! Density (kg/m^3) of particles in particle
group
```

See also [Group](#) and Section 2.5.3.

Particle groups == <'grp1, grp2, ... , grpN' , 'ALL'>

(Mandatory, No Default)

Particle seeding block command that defines which particle groups are to be released. A comma delimited list can be used to seed specific particle groups. Alternatively, to release particles from all groups you can use a single 'ALL' to the right of the ==.

Example using selected particle groups *grp1* and *grp3*:

```
Seed particles == seed_type
  Particle groups == grp1, grp3
  ...
  Commands defining seeding location, vertical distribution and mass flux
  ...
End seed
```

Example that specifies *all* particle groups:

```
Seed particles == seed_type
  Particle groups == all
  ...
  Commands defining seeding location, vertical distribution and mass flux
  ...
End seed
```

See also [Seed particles](#), [Group](#) and Section 2.5.5.

Particle restart file == <restart file name (_ptm.rst)>

(Optional)

Optional command to load the simulation particle initial conditions from a restart file (_ptm.rst) generated by a previous TUFLOW FV PTM simulation.

See also [write restart dt](#) and [restart overwrite](#) commands and Section 2.5.2.6.

Polygon file == <polygon_file.csv (.csv)>

(Optional; No Default)

Particle seeding block command to seed particles evenly over a polygon. The Input .csv file contains the polygon coordinates. Header X, Y and subsequent rows specifying polygon vertices.

Example seed block:

```
Seed particles == polygon
  Particle groups == grp1, grp2
  Polygon file == polygon_file.csv
  Group mass == 1.0e+6, 1.0e+6
End Seed
```

Example contents of polyon_file.csv

x	y
159.0998	-31.3945
159.1021	-31.3949
159.1019	-31.3956
159.1009	-31.3955
159.0996	-31.3952
159.0998	-31.3945

See also [Seed particles](#) and Section 2.5.5

Polyline file == <polyline_file.csv (.csv)>

(Optional; No Default)

Particle seeding block command to seed particles evenly over a polygon. The Input .csv file contains the polyline coordinates. Header X, Y and subsequent rows specifying polyline vertices.

Example seed block:

```
Seed particles == polyline
  Particle groups == grp1, grp2
  Polygon file == polyline_file.csv
  Group mass == 1.0e+6, 1.0e+6
End Seed
```

Example contents of polyline_file.csv

x	y
159.0772	-31.366
159.0853	-31.3744

See also [Seed particles](#) and Section 2.5.5

Reference time == <BC reference time>

(Optional; For Time Format == Hours, Default == 0; For Time Format == ISODate, Default == 01/01/1990 00:00:00)

Optional [Seed particle](#) block command. Can be used to shift the time units of timeseries inputs. Analogous to the HD model reference time for boundary inputs.

Restart overwrite == <0;1>

(Optional, Default == 1)

Option to overwrite the particle restart file at the time interval specified using the [write restart dt](#) command (default) or create a series of restart files for each timestep:

- 0 = False (i.e. the restart file will not be overwritten, and a series of restart files will be generated).
- 1 = True (i.e. the single restart file will be overwritten).

See also [write restart dt](#) and [particle restart file](#) commands and Section 2.5.2.6.

Seed particles == <seed_type>

...
...
...

End Seed

(Optional; No default)

Block to assign particles in to the model where *seed_type* can equal:

- Point
- Polygon
- Polyline
- XYZ
- Moving_point

One-or-more seed particles blocks can be specified in a single simulation.

Please refer to Section 2.5.5 for further examples.

Start output == <time>

(Optional)

Optional particle [output](#) block command to specify the start time for an output Lagrangian NetCDF output request. The time format must be consistent with the simulation time format (i.e. HOURS or ISODATE). If not specified, the output start time will be consistent with the simulation start time.

Please refer to Section 2.6. See also [Output](#).

Start time == <Particle seeding start time>

(Optional; Default == HD Model Start Time)

Optional [Seed particles](#) block command to control the starting time at which particles are released. The time convention and units should be consistent with the hydrodynamic model; in the following example ISODATE convention is being used. The particles will be released at a steady rate between the [Start time](#) and [End Time](#). For example:

```
Seed particles == point, 156.01, -27.50
  Particle groups == grp1
  Group mass == 1.0e+6
  Start time == 01/01/2008 05:00
  End time == 01/01/2008 07:00
End Seed
```

If [Start time](#) is used without a corresponding [End Time](#) command, then all particles within the seed particles block will be released at the [Start time](#).

For more information please refer to Section 2.5.5.1. See also: [End Time](#), [Seed particles](#).

Time units == <Days, Hours, ISOTIME, Minutes, Seconds>

(Optional; Default == Hours)

Optional [Seed particles](#) block command to specify the time format of the input timeseries file. If ISODATE time is used in the TUFLOW FV control file, ISODATE is assumed as default, otherwise hours.

Timeseries file == <Particle seeding timeseries (.csv)>

(Optional; No Default)

Optional [Seed particles](#) block command to assign a timeseries of particle input. The timeseries file is used in combination with the [Particle groups](#) and [Timeseries header](#) commands.

Example seed particles block using a mass flux timeseries:

```
Seed particles == point, 156.01, -27.50
  Particle groups == grp1, grp2
  Timeseries file == mass_flux_ts.csv
  Timeseries header == Date, Apples, Oranges
End Seed
```

Example contents of *mass_flux_ts.csv*

Date	Apples	Oranges
1/05/2011 0:00	1	2
1/05/2011 1:00	1	2
1/05/2011 1:00	0	0

Noting all units are in grams/second.

See also [Timeseries header](#), [Time units](#), [Seed particles](#), [Particle groups](#) and Section 2.5.5.8.

Timeseries header == <header1,header2, ... headerN>

(Optional; No Default)

Seed particles block command that allows the user to specify the .csv input file column headers. This should immediately follow the timeseries file command.

See also [Timeseries file](#), [Time units](#), [Seed particles](#), [Particle groups](#) and Section 2.5.5.8.

Velocity profile model == <None, Logarithmic>

(Optional; Default == None)

Optional Particle group command to reconstruct the vertical profile of velocity from the depth averaged velocity (in the case of a 2D HD model) or from the bottom most 3D layer. If using 'None' the discrete velocity profile from HD model is used with no reconstruction. If 'Logarithmic' a reconstructed profile is used to as particle forcing.

For example:

```
Group == name
  Initial scalar mass == 1.          ! mass in grams, This example:
Nscalar==1
  Particle density == 2650.         ! density in kg/m^3
  d50 == 0.0002                    ! median diameter in m
...
...
Other commands defining particle group behaviour, e.g.
...
  Velocity profile model == logarithmic
End Group
```

See also, [Group](#) and Table 19.

Vertical dispersion model == <None, Constant, Parabolic, HDModel>

(Optional; Default == None)

Optional particle [Group](#) command to influence or set dispersion of particles in the vertical plane. The various options are detailed as follows:

- None: Will turn off the vertical random walk component
- Constant: Applies a constant vertical diffusivity in units of m²/s via the Vertical dispersion parameters command.
- Parabolic: Parabolic diffusivity distribution in lower-half of water-column, constant in top-half. Kappa value input via the Vertical dispersion parameters command.
- HDModel: Will use the vertical diffusivity from the HD model. The vertical diffusivity can then optionally be scaled via the Vertical dispersion parameters command.

For example:

```
Group == name
  Initial scalar mass == 1.          ! mass in grams, This example:
Nscalar==1
  Particle density == 2650.         ! density in kg/m^3
  d50 == 0.0002                    ! median diameter in m
...
...
Other commands defining particle group behaviour, e.g.
...
  Vertical dispersion model == HDMODEL ! Vertical dispersion via HD model
  Vertical dispersion parameters == 1.2 ! Increase HD Vertical dispersion
by 20%
End Group
```

See also [Group](#), [Vertical dispersion parameters](#) and Table 19.

Vertical dispersion parameters == <>

(Optional; Default == 0.)

Optional particle Group command used in combination with the Vertical dispersion model command. Refer [Vertical dispersion model](#).

Wave drift model == <None, Stokes, Stokes_Lagrangian>

(Optional; Default == None)

Optional particle [Group](#) command to allow particles to drift in response to wave processes. The various options are detailed as follows:

- None: Wave drift disabled.
- Stokes: Wave drift velocity proportional to depth-averaged (Eulerian) Stokes drift, with +ve clockwise direction offset. Requires input of two comma delimited [Wave drift parameters](#), a scale_factor and dir_offset.
- Stokes_Lagrangian: Wave drift velocity proportional to Lagrangian Stokes velocity, with +ve clockwise direction offset. Requires input of two comma delimited [Wave drift parameters](#), a scale_factor and dir_offset.

For example:

```
Group == name
  Initial scalar mass == 1.           ! mass in grams, This example:
Nscalar==1
  Particle density == 2650.           ! density in kg/m^3
  d50 == 0.0002                       ! median diameter in m
...
...
Other commands defining particle group behaviour, e.g.
...
Wave drift model == stokes
Wave drift parameters == 1.0, 0.
End Group
```

See also [Group](#), [Wave drift parameters](#) and Table 19.

Wave drift parameters == <>

(Optional; Default == 0.)

Optional particle Group command used in combination with the Wave drift model command.

Refer [Wave drift model](#).

Wind drift depth limits ==<limit1 (m), limit2 (m)>

(Optional, Default == 1.0, 2.0)

Global simulation configuration command. The calculated wind drift rate is linearly scaled down to zero between limit1 and limit2 to avoid applying wind drift to particles too far below the water surface. For example:

```
Wind drift depth limits == 0.5, 1.0      ! Depth from surface (m)
```

See also [Erosion depth limits](#), [Wave depth limits](#), and [Deposition depth limits](#).

Wind drift model == <None, Simple>

(Optional; Default == None)

Optional particle [Group](#) command to allow particles to drift in response to wind forcing. The various options are detailed as follows:

- None: Wind drift disabled.
- Simple: Wind drift velocity proportional to 10mASL wind speed, with a positive clockwise direction offset. Requires input of two comma delimited [Wind drift parameters](#), a scale_factor and dir_offset.

For example:

```
Group == name
  Initial scalar mass == 1.          ! mass in grams, This example:
Nscalar==1
  Particle density == 2650.          ! density in kg/m^3
  d50 == 0.0002                     ! median diameter in m
  ...
  ...
  Other commands defining particle group behaviour, e.g.
  ...
  wind drift model == simple
  wind drift parameters == 0.04, -20.
End Group
```

See also [Group](#), [Wind drift parameters](#) and Table 19.

Wind drift parameters == <0;1, 0;1>

(Optional; Default == 0.)

Optional particle Group command used in combination with the Wind drift model command.

Refer [Wind drift model](#).

Write restart dt == <time (hours)>

(Optional; No Default)

Writes a particle restart file (_ptm.rst) to the log directory location at the time interval specified. The particle restart file is binary format and contains the particle locations and behaviour at an instant in time. A particle restart file is used to specify the initial condition for subsequent TUFLOW FV PTM simulations using the particle restart file command.

See also [Particle restart file](#), [Restart overwrite](#) and Section 2.5.2.6.

XYZ file == <>

(Optional; No default)

Optional Seed particle block command used in combination with the *xyz seed_type* refer [Seed particles](#). The xyz file is sampled row by row as each individual particle is seeded. If necessary, this file will be looped through multiple times until all particles specified by the [Group mass](#) and [Initial scalar mass](#) are released.

An example seed particles block is provided as follows:

```
Seed particles == xyz
  Particle groups == carrots
  xyz file == particle_release.csv
  Group mass == 1.0e+6
End Seed
```

The format of the file *particle_release.csv* is provided below:

X	Y	Z
159.1268	-31.4404	0
159.0976	-31.3966	0
159.0777	-31.3681	-2

The default header can be optionally modified using the XYZ header command.

See also [Seed particles](#), [Particle groups](#), [Group mass](#), [Initial scalar mass](#), [XYZ Header](#) and Section 2.5.5.8.

XYZ header == <XHeader, YHeader, ZHeader>

(Optional; Default == 'X', 'Y', 'Z')

Optional particle seeding block command that can be used in combination with XYZ file.

An example seed particles block is provided as follows:

```
Seed particles == xyz
  Particle groups == carrots
  xyz file == particle_release.csv
  xyz header == Lon, Lat, Elev
  Group mass == 1.0e+6
End Seed
```

The format of the file *particle_release.csv* is provided below:

Lon	Lat	Elev
159.1268	-31.4404	0
159.0976	-31.3966	0
159.0777	-31.3681	-2

See also [Seed particles](#), [Particle groups](#), [Group mass](#), [Initial scalar mass](#), [XYZ file](#) and Section 2.5.5.8.