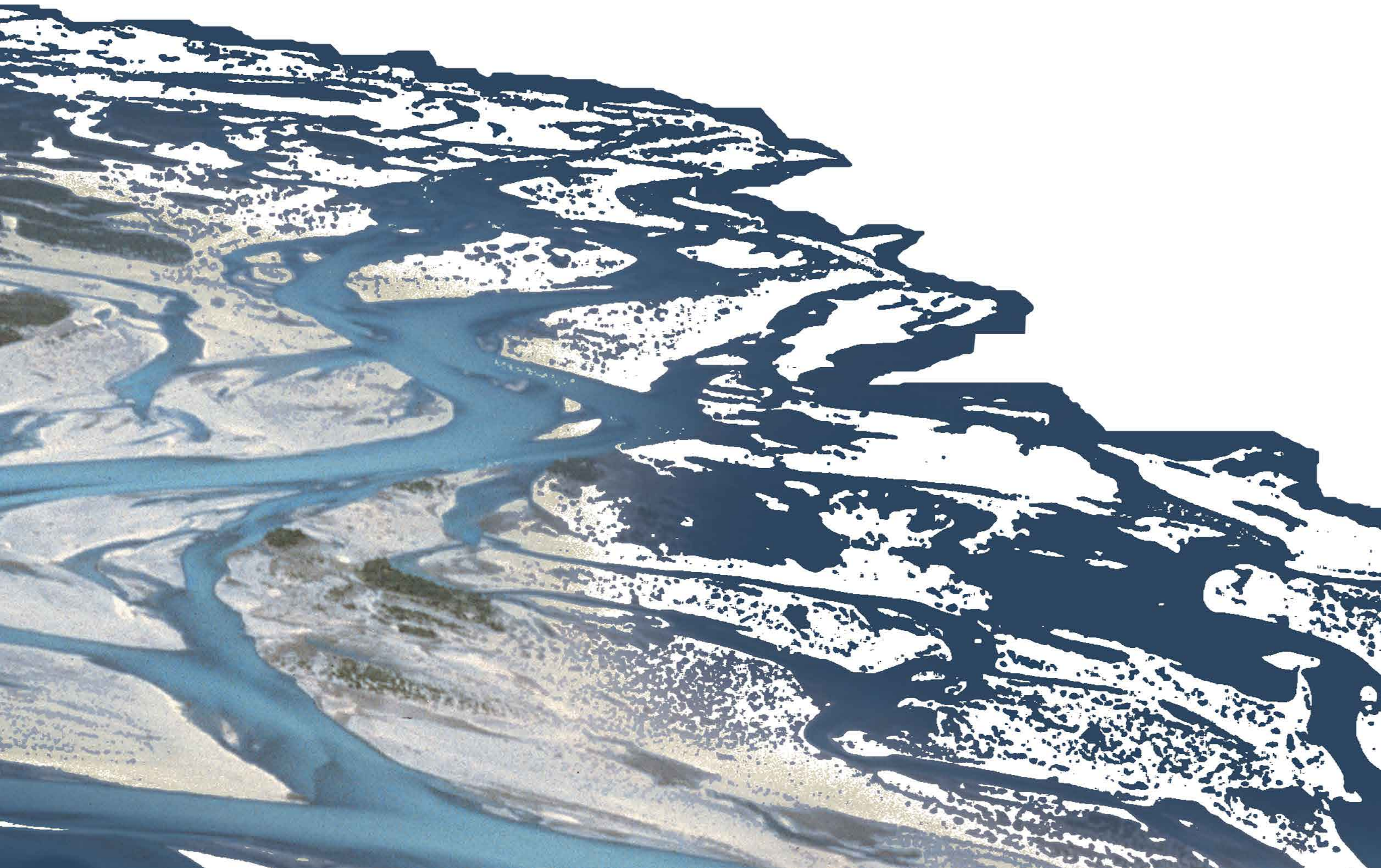
BASEMENT

BASIC SIMULATION ENVIRONMENT
FOR SIMULATION OF ENVIRONMENTAL FLOW
AND NATURAL HAZARD SIMULATION

SYSTEM MANUALS

VERSION 3.1
November 2020



Preamble

VERSION 3.1.1

March 2021

Credits

Current Team Members

Software Development, Documentation and Test (alphabetical)

M. Bürgler, MSc. ETH Environmental Eng.
F. Caponi, MSc. Environmental Eng.
Dr. D. Conde, MSc. Civil Eng.
E. Gerke, MSc. ETH Civil Eng.
S. Kammerer, MSc. ETH Environmental Eng.
Dr.techn. M. Weberndorfer, MSc.

Scientific Board

Prof. Dr. R. Boes, Director VAW, Member of Project Board
Dr. A. Siviglia, MSc, Scientific Adivisor
Dr. D. Vanzo, MSc. Environmental Eng., Scientific Adivisor
Dr. D. Vetsch, Dipl. Ing. ETH, Project Director

Former Project Members

See https://www.basement.ethz.ch/people

Commissioned and co-financed by

Swiss Federal Office for the Environment (FOEN)

Contact

Website: https://www.basement.ethz.ch
User forum: https://people.ee.ethz.ch/~basement/forum

For list of contributors see https://www.basement.ethz.ch

Laboratory of Hydraulics,
Hydrology and Glaciology

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Citation Advice

For System Manuals:

Vetsch D., Siviglia A., Bacigaluppi P., Bürgler M., Caponi F., Conde D., Gerke E., Kammerer S., Koch A., Peter S., Vanzo D., Vonwiller L., Weberndorfer M. 2021. System Manuals of BASEMENT, Version 3.1.1. Laboratory of Hydraulics, Glaciology and Hydrology (VAW). ETH Zurich. Available from https://www.basement.ethz.ch. [date of access].

For Website:

BASEMENT – Basic Simulation Environment for Computation of Environmental Flow and Natural Hazard Simulation, 2021. https://www.basement.ethz.ch

For Software:

BASEMENT – Basic Simulation Environment for Computation of Environmental Flow and Natural Hazard Simulation. Version 3.1.1 © ETH Zurich, VAW, 2021.

License

End-User License Agreement (EULA)

THIS EULA IS INTENDED FOR COMMERCIAL AND NON-COMMERCIAL PURPOSES. FOR QUESTIONS RELATED TO THIS AGREEMENT PLEASE CONTACT: Dr. David Vetsch, basement@ethz.ch

This End-User License Agreement ("EULA") is a legal agreement between you ("You") (an individual or acting on behalf of a company) and ETH Zurich, Raemistrasse 101, 8092 Zurich (Switzerland) ("ETH Zurich") for the binary software code of **BASEMENT** and associated media, and may include "online" or electronic documentation ("SOFTWARE").

The SOFTWARE simulates water flow, sediment and scalar transport in rivers and according interaction in consideration of movable boundaries and morphological changes. Further information and description of the SOFTWARE is available here: https://basement.ethz.ch/

The SOFTWARE is protected by copyright laws. The SOFTWARE is hereby licensed, not sold.

In order to install and use the SOFTWARE, You must indicate agreement with the following terms and conditions by clicking "ACCEPT" at the end of this EULA during the installation process.

1 LICENSE GRANT

(i) ETH Zurich hereby grants to You, and in case You are acting on behalf of a company also to the employees of such company, a free-of-charge, single, non-exclusive, world-wide, non-transferable, non-sublicensable right to install, execute and display the SOFTWARE on device(s) running a validly licensed copy of the operating system for which the SOFTWARE was designed. Such rights are granted for commercial and non-commercial purposes.

(ii) With respect to electronic documents included with the SOFTWARE, You may make an unlimited number of copies (either in hardcopy or electronic form), provided that such copies shall be used only for internal purposes and are not republished or distributed to any third party.

2 USE OF RESULTS GENERATED BY THE SOFTWARE

You are allowed to use the content generated by the SOFTWARE ("SOFTWARE RESULTS") for commercial and non-commercial purposes.

Note that any attribution (e.g. ETH Zurich logo) on the SOFTWARE RESULTS must be retained. You are not allowed to alter, cancel or fade, after a few seconds, such attribution.

3 THIRD PARTY CODE

The SOFTWARE may contain other program code from third parties. A list of other third party code and libraries used by this SOFTWARE is available here: https://basement.ethz.ch/about/thirdpartysoftware

Their license applies to such third party code and libraries contained herein. Refer to the above internet site for the licenses and copyrights.

4 DURATION OF LICENSE AND TERMINATION

This EULA enters into effect on the date of acceptance of this EULA by You. This EULA,

(i) may be terminated by ETH Zurich at any time for any reason;

(ii) will terminate automatically without notice from ETH Zurich if (a) You fail to comply with any term(s) of this EULA or (b) You refuse, after the notification in accordance with clause 10 (i), to accept the new EULA term and conditions provided by ETH Zurich;

(iii) is terminated as soon as You cease to use the SOFTWARE and destroy all copies, full or partial, of the SOFTWARE;

Upon termination pursuant to (i) and (ii), you must cease all use of the SOFTWARE and destroy all copies, full or partial, of the SOFTWARE.

5 OBLIGATIONS OF YOU

(i) You may not remove or alter any copyright notices on any and all copies of the SOFTWARE.

(ii) You may not distribute or assign the SOFTWARE or any copy thereof to third parties. You may not rent, lease, sell, lend, transfer, redistribute, or sublicense the SOFTWARE to any third party.

(iii) You may not reverse engineer, decompile or disassemble the SOFTWARE, except and only to the extent that such activity is expressly permitted by applicable law despite this limitation.

(iv) You may not reproduce, modify or adapt the SOFTWARE, except and only to the extent that such activity is expressly permitted by applicable law despite this limitation.

(v) You must comply with all applicable laws.

6 OWNERSHIP

Except as expressly licensed to You in this EULA, ETH Zurich and its licensors retains all right, title, and interest in and to the SOFTWARE. All title and copyrights in and to the SOFTWARE (including but not limited to any images, photographs, animations, video, audio, music, text, and "applets" incorporated into the SOFTWARE), the accompanying materials, and any copies of the SOFTWARE are owned by ETH Zurich and its licensors. The SOFTWARE is protected by copyright laws. Therefore, You must treat the SOFTWARE like any other copyrighted material. All rights not expressly granted are reserved by ETH Zurich.

7 MAINTENANCE, SUPPORT, UPGRADES OR NEW RELEASES

ETH Zurich has no obligation to provide maintenance, support, upgrades, new releases, enhancements or modifications and disclaims all costs associated with service, repair or correction of the SOFTWARE. If any supplemental software code is provided to You by ETH Zurich, this supplemental software code shall be considered part of the SOFTWARE and is subject to the terms and conditions of this EULA if not otherwise explicitly written. It is expressly acknowledged by You that no rights to receive maintenance, support, upgrades, new releases, enhancements or modifications may be derived from this EULA.

8 NO WARRANTY

YOU EXPRESSLY ACKNOWLEDGE AND AGREE THAT USE OF THE SOFTWARE IS AT YOUR SOLE RISK AND THAT THE ENTIRE RISK AS TO SATISFACTORY QUALITY, PERFORMANCE, ACCURACY, AND EFFORT IS WITH YOU. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, THE SOFTWARE AND ANY SERVICES PERFORMED OR PROVIDED BY THE SOFTWARE ARE PROVIDED "AS IS" AND "AS AVAILABLE", WITH ALL FAULTS AND WITHOUT WARRANTY OF ANY KIND, AND ETH ZURICH HEREBY DISCLAIMS ALL WARRANTIES AND CONDITIONS WITH RESPECT TO THE SOFTWARE AND ANY SERVICES, EITHER EXPRESS, IMPLIED, OR STATUTORY, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES AND/OR CONDITIONS OF MERCHANTABILITY, OF SATISFACTORY QUALITY, OF FITNESS FOR A PARTICULAR PURPOSE, OF ACCURACY, OF QUIET ENJOYMENT, AND OF NON-INFRINGEMENT OF THIRD-PARTY RIGHTS. ETH ZURICH DOES NOT WARRANT AGAINST INTERFERENCE WITH YOUR ENJOYMENT OF THE SOFTWARE, THAT THE FUNCTIONS CONTAINED IN OR SERVICES PERFORMED OR PROVIDED BY THE SOFTWARE WILL MEET YOUR REQUIREMENTS, THAT THE OPERATION OF THE SOFTWARE OR SERVICES WILL BE UNINTERRUPTED OR ERROR-FREE, OR THAT DEFECTS IN THE SOFTWARE OR SERVICES WILL BE CORRECTED. NO ORAL OR WRITTEN INFORMATION OR ADVICE GIVEN BY ETH ZURICH OR ITS AUTHORIZED REPRESENTATIVE SHALL CREATE A WARRANTY. SHOULD THE SOFTWARE OR SERVICES PROVE DEFECTIVE, YOU ASSUME THE ENTIRE COST OF ALL NECESSARY SERVICING, REPAIR, OR CORRECTION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES OR LIMITATIONS ON APPLICABLE STATUTORY RIGHTS OF A CONSUMER, SO THE ABOVE EXCLUSION AND LIMITATIONS MAY NOT APPLY TO YOU.

9 LIABILITY

In no event shall ETH Zurich be liable for any damages (including, without limitation, lost profits, business interruption, or lost information) arising from the use of or inability to use the SOFTWARE through You or the employees of the company You are legally representing. In no event will ETH Zurich be liable for loss of data or for indirect, special, incidental, consequential (including loss of profit), or other damages based in contract, tort or otherwise.

The above limitation of liability shall not be construed to amend or limit any party's statutory liability.

10 GENERAL PROVISIONS

(i) ETH Zurich reserves the right to change the terms and conditions of this EULA at any point in time. In such event, ETH Zurich will notify You in due time of the changes to the terms of the EULA.

(ii) Rights and duties derived from this EULA shall not be transferred to third parties without the written acceptance of the ETH Zurich.

(iii) You shall not infer from this EULA any other rights, including licenses, than those that are explicitly stated herein.

(iv) This EULA shall exclusively be governed by and interpreted in accordance with the laws of Switzerland, without reference to its conflict of laws principles. The exclusive place of jurisdiction is Zurich (Switzerland).

11 AKNOWLEDGMENT

You acknowledge that you have read this EULA, understand it, and had an opportunity to seek independent legal advice prior to agreeing to it. In consideration of ETH Zurich agreeing to provide the SOFTWARE, You agree to be bound by the terms and conditions of this EULA. You further agree that it is the complete and exclusive statement of the agreement between you and ETH Zurich, which supersedes any proposal or prior agreement, oral or written, and any other communication between you and ETH Zurich relating to the subject of this EULA.

Notice:

Third party software copyright notices and third party software licenses can be found in the appendix.

BASIC SIMULATION ENVIRONMENT
FOR SIMULATION OF ENVIRONMENTAL FLOW
AND NATURAL HAZARD SIMULATION

REFERENCE MANUAL

VERSION 3.1
November 2020

Contents

1

Mathematical Models

1.1 Hydrodynamics

1.1.1 Introduction

Mathematical models of the so-called *shallow water* type govern a wide variety of physical phenomena. Especially the one-dimensional (1D) de Saint-Venant equations (SVE) or two-dimensional (2D) shallow water equations (SWE) are of practical interest with regard to water flows with a free surface under the influence of gravity. Applications of the models include e.g.:

- River hydrodynamics
- Propagation of flood waves
- Dam break waves
- Flooding and inundation
- Ecological assessment based on flow quantities

The 2D SWE are based on the following set of hypotheses:

- the water is assumed to be incompressible; i.e. the water density ρ is constant
- the vertical acceleration of the water particles are assumed to be small compared to the longitudinal component of the acceleration. As a consequence the pressure distribution is hydrostatic;
- the bottom slope is small enough for the longitudinal coordinate to coincide with the horizontal axis;
- the flow regime is turbulent. As a consequence the head loss, mainly due to friction against the bottom, is proportional to the square of the flow velocity.

1.1.2 Governing Equations

The governing equations are obtained under shallow water conditions imposing mass conservation for the fluid and solid phases and the momentum principle to a flow in an open channel with a fixed bottom.

Introducing a Cartesian reference system (x, y, z) in which the z axis is vertical and the $x - y$ plane is horizontal with respect to gravity g , the system of governing equations can be written as

$$\begin{cases} \frac{\partial h}{\partial t} + \frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y} = S_h \\ \frac{\partial q_x}{\partial t} + \frac{\partial}{\partial x}\left(\frac{q_x^2}{h} + \frac{1}{2}gh^2\right) + \frac{\partial}{\partial y}\left(\frac{q_x q_y}{h}\right) + gh\left(S_{bx} + S_{fx}\right) = 0 \\ \frac{\partial q_y}{\partial t} + \frac{\partial}{\partial x}\left(\frac{q_y q_x}{h}\right) + \frac{\partial}{\partial y}\left(\frac{q_y^2}{h} + \frac{1}{2}gh^2\right) + gh(S_{by} + S_{fy}) = 0 \ , \end{cases} \tag{1.1}$$

where:

h	$[m]$	water depth
g	$[m/s^2]$	gravity acceleration
u (v)	$[m/s]$	depth averaged velocity in x (y) direction
q_x (q_y)	$[m^2/s]$	discharge per unit width in x (y) direction
S_h	$[m/s]$	lateral inflow/outflow discharge per unit width
S_{fx} (S_{fy})	$[-]$	friction terms in x (y) direction .

The bed slope source terms

$$S_{bx}, S_{by}$$

are evaluated as follows:

$$S_{bx} = -\frac{\partial z_B}{\partial x} \quad ; \quad S_{by} = -\frac{\partial z_B}{\partial y} \tag{1.2}$$

1.1.3 Closure Relations

In order to solve system (eq. 1.1) we need to specify the closure relations for the friction terms S_{fx}, S_{fy} and the value of lateral inflow/outflow discharge per unit width S_h.

1.1.3.1 Friction Terms

The governing equations (eq. 1.1) have been derived under the hypothesis of turbulent flow, hence the friction terms can be assumed proportional to the square of the depth-averaged velocity and can be written as:

Several formulae are available for S_f. All these formulae use hypothesis (H3) of a turbulent flow regime, hence the assumption that the slope of the energy line is proportional to the square of the flow velocity u. The most frequently used laws are

Adopting a quadratic friction law, the friction term is proportional to the square of the depth-averaged velocity and can be written as:

$$S_{fx} = \frac{u|\vec{u}|}{gc_f^2 h} \quad ; \quad S_{fy} = \frac{v|\vec{u}|}{gc_f^2 h} \tag{1.3}$$

where g is the gravity acceleration, u and v are the depth averaged velocities in x and y direction, $|\vec{u}| = \sqrt{u^2 + v^2}$ is the magnitude of the velocity vector and c_f is the dimensionless friction coefficient.

Several formulae are available for the dimensionless friction coefficient c_f. Here it is quantified using both a power or a logarithmic for which are described in the next sections.

1.1.3.1.1 Power Law

The Manning-Strickler power law is widely used in practice and it requires that either the Strickler's k_{str} $[m^{1/3}/s]$ or the Manning's n coefficients ($k_{str} = n^{-1}$) is specified.

In this case the dimensionless friction coefficient c_f is calculated as

$$c_f = \frac{k_{str} h^{1/6}}{\sqrt{g}} \tag{1.4}$$

1.1.3.1.2 Logarithmic Law

The following approaches are implemented to determine the friction coefficient c_f:

Chézy:

$$\begin{aligned} c_f &= 5.75 \log \left(12 \frac{R}{K_s}\right) && \text{for} \quad R > K_s \\ c_f &= 5.75 \log (12) && \text{for} \quad R < K_s \ , \end{aligned} \tag{1.5}$$

where K_s $[m]$ is the bed roughness height which is commonly taken to be proportional to a representative sediment size d_x. For rivers, K_s can be assumed $K_s = n_k d_{90}$ where $n_k = 2 \div 3$.

Bezzola:

In this closure relation, proposed by Bezzola (2002), c_f is given as a function of the roughness sublayer height y_R $[m]$ (usually for rivers $y_R \approx 1.0 d_{90}$ is a good approximation). This approach is also valid for small values of the relative submergence h/y_r Bezzola (2002).

$$\begin{cases} c_f = 2.5\sqrt{1 - \frac{y_R}{h}} \ \ln\left(10.9\frac{R}{y_R}\right), & \text{for} \quad \frac{h}{y_R} > 2 \\ c_f = 1.25\sqrt{\frac{h}{y_R}} \ \ln\left(10.9\frac{R}{y_R}\right), & \text{for} \quad 0.5 \le \frac{h}{y_R} \le 2 \\ c_f = 1.5, & \text{for} \quad \frac{h}{y_R} < 0.5 \end{cases} \tag{1.6}$$

1.1.3.2 Lateral Inflow/Outflow

S_h is used to represent additional sources of water like rainfall and springs or water abstraction (sink) and are allocated on a set of elements defined by regions. The external

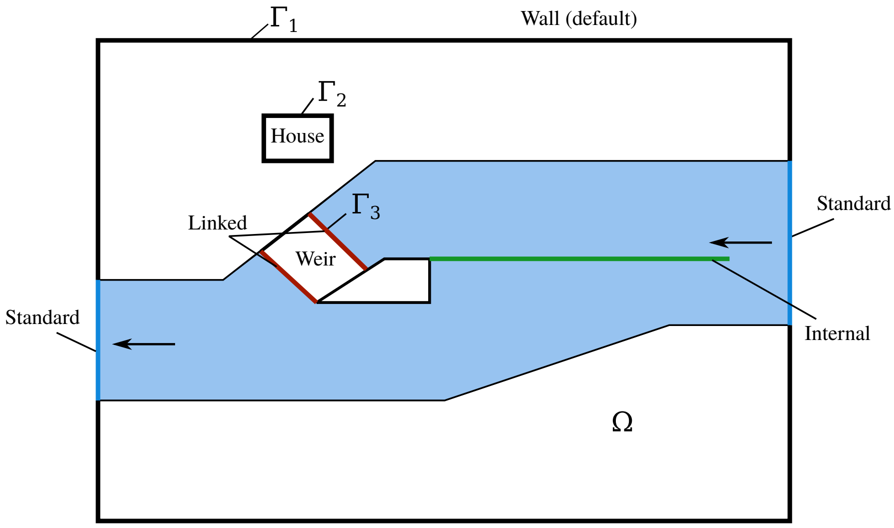


Figure 1.1 *Modeling domain and types of boundary conditions available. The flow is from right to left and a side weir (green line) divides the channel into a lower and an upper channel through the weir.* ***Standard*** *or* ***linked*** *boundary conditions must be provided at* Γ_1, Γ_2 *and* Γ_3 *while* ***internal*** *boundary conditions can be specified in any place within* Ω

source can be specified as total discharge $[m^3/s]$ or distributed over time $[mm/h]$. Different approaches are used to manage the behaviour of the external sources:

- Exact: The specified water volume is added or extracted (non conservative)
- Available: The specified water volume to extract is limited by the available water volume in the elements (conservative)
- Infinity: All available water will be abstracted (conservative)

Addition of water always follows the "Exact" behaviour as there is no upper limit. The abstraction of water could also follow the "Exact" behaviour but the simulation might end abruptly if the available water volume is smaller than the volume prescribed. Therefore, the "Available" behaviour aims to avoid this situation. The "Infinity" behaviour abstracts all available water volume.

1.1.4 Boundary Conditions

After the specification of the *closure relations* there are now three equations and three unknowns, namely h, q_x and q_y. In principle, given initial and boundary conditions, one should be able to solve system (eq. 1.1) for h, q_x and q_y as functions of space x, y, and time t. Given the modeling domain described in Figure 1.1, boundary conditions are required at the domain boundary Γ and optionally can be specified within the interior domain Ω.

Therefore, three different types of boundary conditions can be defined:

- Standard boundary conditions: located at the domain boundary Γ_i
- Linked boundary conditions: located at the domain boundary Γ_i or inside the domain Ω

- Internal boundary conditions: located inside the domain Ω

Standard boundaries (at Γ) represent the limits of the computational domain possibly including also buildings, weirs or structures for water intake (see Figure 1.1).

1.1.4.1 Standard Boundary Conditions

At the standard boundaries, two different types of boundary conditions can be specified: wall or flow boundaries. Flow boundary conditions allow the flow to enter or leave the domain while wall boundary conditions express no mass flux over the boundary. By default, the extneral boundaries of the domain are set as wall boundaries.

1.1.4.1.1 Wall Boundaries

The *Wall* or *reflective* boundary consider the boundary at Γ_i and suppose it physically consists of a fixed, reflective impermeable wall. Then the physical situation is modelled imposing that:

$$\rho\vec{u}\cdot\vec{n} = 0 \quad ; \quad \frac{\partial\vec{u}}{\partial\vec{n}} = 0 \tag{1.7}$$

Where $\vec{n}$ is the outward directed unit vector perpendicular to the wall and $\vec{u} = (u, v)^T$ is the velocity vector. The static pressure is assumed to be zero.

1.1.4.1.2 Flow Boundaries

The *Flow* boundary conditions are defined as *inflow* if they let water entering or as *outflow* if they let water leaving the domain. Flow boundaries are further distinguished into *Standard* and *Linked*. The former are applied on the boundary domain Γ, while the latter establish a *link* between two portions of the domain.

Standard

Inflow boundaries:

This boundary requires the specification of a value for the total volume discharge Q, $[m^3/s]$, which is then divided by the length of the boundary Γ and projected orthogonally to the boundary to obtain the values of q_x and q_y. In case of supercritical flow the following possibilities to specify the value of the water depth h are possible:

- *uniform_in*: h is calculated assuming that local uniform flow conditions. The calculation proceeds as follows:

  $$h = \sqrt[3]{\frac{(Q/b)^2}{gc_f^2 s}} \tag{1.8}$$

  where c_f is the Chézy coefficient, b is the entire length of the boundary Γ and s is the value of the local bed slope that must be specified.

- *froude_in*: In this case the flow depth h is calculated as follows:

$$h = \sqrt[3]{\frac{(Q/b)^2}{gFr^2}} \tag{1.9}$$

where b is the entire length of the boundary and Fr is the value of the local Froude number that must be specified

- *zhydrograph*: The water surface elevation (wse) at the boundary must be specified by the user. The depth is calculated as:

$$h = wse - z_B \tag{1.10}$$

where z_B is the bottom elevation at the boundary. The flow velocity at the boundary is set to zero.

Outflow boundaries:

At the outflow boundaries a value for the water depth h must be specified. These are the possible options:

- *uniform_out*: the water depth h is calculated using equation (eq. 1.8) specifying a value for the total discharge Q and a local bed slope s. Uniform flow is calculated based on given slope and cell state at boundary (eq. 1.8).

- *weir_out_constant* and *weir_out_dynamic*: These boundary conditions establishe a relation between the approaching discharge q constant and the water depth using the Poleni weir formula:

$$q = \frac{2}{3}\mu\sqrt{2g(h_{up} - w)^3} \tag{1.11}$$

where h_{up} is the water depth of the approaching flow and w is the weir elevation. The Poleni factor μ can be either set as constant ($\mu = 0.75$ by default) or dynamically evaluated as:

$$\mu = \frac{0.611}{a}\frac{0.75}{b}\frac{h_{up} - z_w}{w} \tag{1.12}$$

where a and b must be specified by the user in the case of *weir_out_dynamic* (default values are $a = 0.611$ and $b = 0.075$).

- *hqrelation_out*: The discharge is determined as a function of the water surface elevation, thus a stage-discharge-relation has to be specified.

- *zhydrograph*: Sets a fixed water surface elevation (wse) at the boundary. The wse [m] at the boundary must be specified by the user. The depth is calculated as:

$$h = wse - z_B \tag{1.13}$$

where z_B is the bottom elevation at the boundary. The flow velocity is calculated with the Riemann solver (HLLC).

- *zero_gradient_out* (scientific use only): Transmissive, or transparent boundaries allow the passage of waves without any effect on them. This is mathematically obtained imposing over the entire length of the boundary that:

$$\rho\vec{u}\cdot\vec{n} = \text{const} \quad ; \quad \frac{\partial \vec{u}}{\partial \vec{n}} = 0 \tag{1.14}$$

  In this case there is no need to specify further parameters.

  Note: This is boundary condition should ***not*** *be used for practical problems and is intended for scientific use only.*

1.1.4.2 Linked Boundary Conditions

This type of boundaries establish a *link* between within a certain region of the domain where equations are not solved. Once this domain portion is identified the two boundaries, between which the link is established, must be specified. Let us call them Γ_{in} and Γ_{out}. Then, one inflow boundary condition must be specified at Γ_{in} and one outflow boundary condition at Γ_{out} while in the remaining boundaries wall conditions are automatically assigned. Not necessarily, Γ_{in} and Γ_{out} must have the same number of elements.

Linked boundaries can describe a $h - Q$ relation or a weir, i.e.:

- *weir_linked_constant* and *weir_linked_dynamic*: Similar to the standard weir boundary, the weir height w has to be specified. No kinetic energy is considered.
- *hqrelation_linked*: The flux is calculated given a h-Q relation (see description of the h-Q relation for standard boundaries).
- *2way_hqrelation_linked*: The internal boundary works as dynamic wall that is controlled by water surface elevation thresholds. If the upper water surface elevation threshold is reached, the internal boundary is removed until the water level reaches the lower water surface elevation, where the wall is re-established.
- *zhydrograph_linked*: Sets a fixed water surface elevation (WSE) at the upstream boundary. The flux is caculated with the Riemann solver and used as inflow in the downstream boundary. Input of kinetic energy is neglected at the downstream boundary.
- *zhydrograph_linked_kinE*: Sets a fixed water surface elevation (WSE) at the upstream boundary. The flux is caculated with the Riemann solver and used as inflow in the downstream boundary. Input of kinetic energy is taken into account at the downstream boundary, using the flow depths of the cells adjacent to the boundary.

1.1.4.3 Internal Boundary Conditions

The internal boundary condition allows a direct cell-cell relation due to the exact same number of elements on the left and on right side of the boundary. Internal boundary conditions can be used to specify internal walls, dynamic walls or an h-Q relation.

- *wall_internal*: The wall conditions (eq. 1.7) are applied on both sides of the internal boundary.

- *dynamic_wall_internal*: The wall conditions are applied on the internal boundary until reaching a threshold value (time or water depth) after which the wall is removed.
- *hqrelation_internal*: A stage-discharge relation is applied on one side of the internal boundary, while on the other side, wall conditions apply (unidirectional flow).

1.1.5 Flood Tracking

The flood tracking aims at extracting the flood arrival time, the maximum water depth, flow velocity, specific discharge and bed shear stress along the numerical simulation and over a selected domain area. The flood tracking provides outputs within a tracking time step defined by the user.

1.2 Morphodynamics

1.2.1 Introduction

Morphodynamic models provide scientific frameworks for advancing our understanding of river systems. The research on involved topics is an important and socially relevant undertaking regarding our environment. Nowadays numerical models are used for different purposes, from answering questions about basic morphodynamic research to managing complex river engineering problems. Due to increasing computer power and the development of advanced numerical techniques, morphodynamic models are now more and more used to predict the bed patterns evolution to a broad spectrum of spatial and temporal scales. The development and the success of application of such models are based upon a wide range of disciplines from applied mathematics for the numerical solution of the equations to geomorphology for the physical interpretation of the results.

Applications of morphodynamic models include:

- Damming of river basins
- Morphological changes due to width changes (e.g. River widenings)
- Effects of sediment mining
- River straightening

1.2.2 Bedload Sediment Transport

1.2.2.1 Governing Equations for Uniform Sediment Transport

The governing equations are obtained under shallow water conditions imposing mass conservation for the fluid and solid phases and the momentum principle to a flow in an open channel with a cohesionless bottom.

Introducing a Cartesian reference system $(x; y; z)$ in which the z axis is vertical and the $x - y$ plane is horizontal, the system of governing equations is described by the system of equations (eq. 1.1) for hydrodynamics coupled with one equation for the conservation of the total sediment mass. The conservation of sediment mass is ensured by the Exner equation (eq. 1.15), named after the Austrian sedimentologist Felix M. Exner (Exner, 1925). The Exner equation allows to describe the bed evolution due to erosion or deposition, which results in the elevation change of the actual bed level z_B:

$$(1-p)\frac{\partial z_B}{\partial t} + \frac{\partial q_{B_x}}{\partial x} + \frac{\partial q_{B_y}}{\partial y} - Sl_b = 0 \tag{1.15}$$

where p is the porosity, Sl_b is the source term per unit width specifying local input or output of sediment material (e.g. slope collapse or excavation) and $\vec{q}_B = \begin{pmatrix} q_{B_x} \\ q_{B_y} \end{pmatrix}$ is the specific bedload flux. The Exner equation describes the bed evolution due to erosion or deposition processes, which results in changes of the bed level z_B.

The Exner equation is solved in a decoupled way, meaning that the shallow water equations and the Exner equation are solved in sequence. This approach makes the assumption that the bedload flux is much slower than the water flow velocity (Soares-Frazão and Zech, 2011).

1.2.2.2 Threshold Conditions for Sediment Movement

The key dimensionless parameter quantifying sediment mobility is the Shields parameter defined as:

$$\theta = \frac{|\vec{\tau_b}|}{(\rho_s - \rho)gd} \tag{1.16}$$

where $vec\tau_b$ is the bed shear stress (drag force acting on the particle), d is the sediment diameter, ρ and ρ_s are the water and sediment density, respectively. The Shields parameter can be interpreted as the ratio scaling the impelling force of flow drag acting on a particle to the Coulomb force resisting motion acting on the same particle.

The bed shear stress $\vec{\tau_b} = \begin{pmatrix} \tau_{bx} \\ \tau_{by} \end{pmatrix}$ is usually estimated by a closure condition using an empirical or semi-empirical formula. Here we use the quadratic friction law which relates the depth-averaged velocities to the bed shear stress as follows:

$$\tau_{bx} = \rho\frac{|\vec{u}|u}{c_f^2} \quad ; \quad \tau_{by} = \rho\frac{|\vec{u}|v}{c_f^2} \tag{1.17}$$

where $\vec{u} = \begin{pmatrix} u \\ v \end{pmatrix}$ is the flow velocity vector, ρ is the density of water and c_f is the dimensionless Chézy friction coefficient as defined in Section 1.1.3.1.

When a granular bed is subjected to a turbulent flow, it is found that virtually no motion of the grains is observed below a critical value (θ_{cr}) of the Shields parameter. According to the Shields' theory Shields (1936), θ_{cr} can be expressed as a function of the Reynolds number $Re^* = \frac{du_*}{\nu}$. Alternatively, the diagram of incipient motion (see Figure 1.2) can be plotted as a function of the dimensionless grain diameter D^* ($\theta_{cr} = f(D^*)$), where

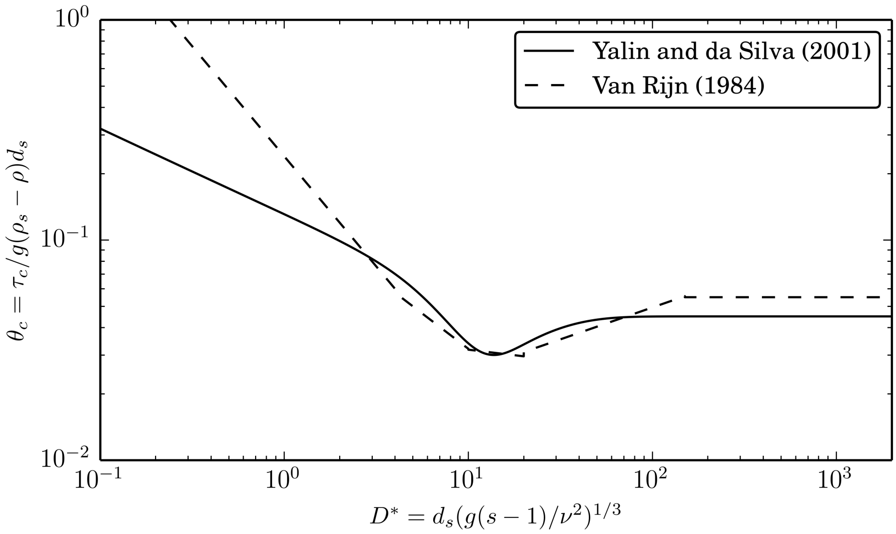


Figure 1.2 *Modified Shields diagram for initiation of sediment motion*

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

.

The curve representing the particle incipient motion ($\theta = \theta_{cr}$) can be divided into three parts in the log-log graph:

- for $D^* \leq 3$, can be approximated by a linear segment;
- for $3 \leq D^* \leq 100$ this is represented by a curve with a relative minimum;
- for \$D^* > 100 \$ by a constant trend.

An approximation of the original Shields diagram was proposed by van Rijn (1984):

$$\begin{array}{lll} \theta_{cr} = 0.24(D^*)^{-1} & \text{for} & 1 \leq D^* \leq 4 \\ \theta_{cr} = 0.14(D^*)^{-0.64} & \text{for} & 4 < D^* \leq 10 \\ \theta_{cr} = 0.04(D^*)^{-0.1} & \text{for} & 10 < D^* \leq 20 \\ \theta_{cr} = 0.013(D^*)^{0.29} & \text{for} & 20 < D^* \leq 150 \\ \theta_{cr} = 0.055 & \text{for} & D^* > 150 \end{array} \tag{1.18}$$

Another explicit formulation of the Shields curve was proposed by Yalin and Silva (2001). It reads

$$\theta_{cr} = 0.13 D^{*-0.392} \exp(-0.015 D^*) + 0.045\left(1 - \exp(-0.068 D^*)\right) \tag{1.19}$$

1.2.2.2.1 Influence of Local Slope on Incipient Motion

The threshold condition for incipient motion of grains developed by Shields is valid for almost horizontal bed. In case of sloped bed in flow direction or transverse to it, the stability of grains is either increased or reduced due to the gravity. The critical shear stress value can be adapted consequently to account for the influence of local slopes. One approach is to multiply the critical shear stress for almost horizontal bed θ_{cr} with the correction factors k_l and k_t for the local bed slope in the longitudinal and transversal flow direction. In the following, the critical Shields stress corrected for arbitrary bed slope δ is referred to as $\theta_{c,\delta}$, defined as:

$$\frac{\theta_{c,\delta}}{\theta_{cr}} = k_l k_t = k \tag{1.20}$$

The correction factor k_l and k_t are calculated as suggested by van Rijn (1989):

$$k_l = \cos\delta_l \left(1 - \frac{\tan\delta_l}{\tan\gamma}\right) \tag{1.21}$$

$$k_t = \cos\delta_t \sqrt{1 - \frac{\tan^2\delta_t}{\tan^2\gamma}} \tag{1.22}$$

where δ_l is the angle between the horizontal and the bed along flow direction, δ_t is the slope angle transversal to the flow direction and γ is the angle of repose of the sediment material.

Other formulations are also available, as for example the one proposed by Chen et al. (2010):

$$k = \frac{1}{\tan\gamma}\left(\cos^2\left(\frac{\pi}{2} - \delta_l\right) - 1 + \frac{1 + \tan^2\gamma}{(1 + \tan^2\delta_l + \tan^2\delta_t)}\right)^{0.5} + \cos\left(\frac{\pi}{2} - \delta_l\right) \tag{1.23}$$

1.2.2.3 Closure Relations for Bedload Transport

In order to solve system (eq. 1.1) and equation (eq. 1.15) we need to specify the closure relations. For the friction terms S_{fx}, S_{fy} and the value of lateral inflow/outflow discharge per unit width S_h we can use the relations already introduced in the Hydrodynamic part (Section 1.1.3). For the Exner equation we need relations quantifying the bedload discharge. Let us now introduce the dimensionless bedload transport rate Φ also known as the Einstein bedload number, first introduced by Hans Albert Einstein in 1950, and given by

$$\Phi = \frac{q_B}{\sqrt{(s-1)gd^3}} \tag{1.24}$$

where $s = \rho_s/\rho$.

It is common practice to quantify bedload transport empirically relating Φ with either the Shields stress θ or the excess of the Shields stress θ above some appropriately defined "critical" Shields stress $(\theta - \theta_{cr})$. The critical Shields stress θ_{cr} is defined so as to fit experimental or field data and provide a threshold for which the bedload transport rate is too low to be of interest. The Shields parameter, takes the following form

$$\theta = \frac{h\sqrt{S_{fx}^2 + S_{fy}^2}}{(s-1)d} \tag{1.25}$$

where h is the water depth, S_{fx} and S_{fy} the friction slope in x- and y-direction respectively, $s = \rho_s/\rho_w$, and d is the grain size diameter. Note that Eq. 1.16 and Eq. 1.25 are equivalent.

In what follows, we describe the bedload transport formulas that are implemented to calculate the transport capacity $q_B = |\vec{q_B}|$, where the specific bedload flux vector $\vec{q_B} = (q_{B_x}, q_{B_y})$ generally has the same direction as the water flow.

For practical purposes, the bedload transport formula can be calibrated by an additional pre-factor (*factor*). The bedload transport capacity is obtained from the closure relation scaled by this pre-factor.

1.2.2.3.1 Meyer-Peter and Müller (1948)

The bedload transport formula of Meyer-Peter and Müller (Meyer-Peter and Müller, 1948) defines the specific bedload transport rate q_B as:

$$q_B = \alpha(\theta - \theta_{cr})^m \sqrt{(s-1)gd^3} \tag{1.26}$$

Herein, α denotes the bedload coefficient (originally $\alpha = 8$), m the bedload exponent (originally $m = 1.5$), θ is the dimensionless bed shear stress (Shields parameter), θ_{cr} is the critical dimensionless bed shear stress, d is the grain diameter, $s = \rho_s/\rho$ and g stands for the gravitational acceleration. Meyer-Peter and Müller observed in their experiments that the first grains moved already for $\theta_{cr} = 0.03$. But as their experiments took place with steady conditions they used a value for which already 50% of the grains where moving. They proposed the value of $\theta_{cr} = 0.047$. The formula of Meyer-Peter and Müller is applicable in particular for coarse sand and gravel with grain diameters larger than 1 mm (Malcherek, 2001).

The bedload coefficient α, the exponent m and the critical Shields parameter θ_{cr} can be adapted by the user in the MPM-like formula.

1.2.2.3.2 Grass Formula

The Grass formula (Grass, 1981) proposes a simple bedload transport formula, where q_b is a function of the flow velocity u and a dimensional constant α and does not require the evaluation of the Shields stress:

$$q_B = \alpha(u - u_c)^m \cdot \sqrt{(s-1)gd^3} \tag{1.27}$$

where $\alpha \in [0, 1]$ is a dimensional constant that encompasses the effects of grain size and kinematic viscosity and is usually determined from experimental data, u_c is the critical velocity and the exponent m is usually set to $m = 3$. The threshold condition for incipient motion of grains is typically set to zero, meaning that the bedload transport and the fluid motion start simultaneously. The coefficient α characterizes the interaction between the bed and the fluid. If $\alpha = 0$, no sediment transport occurs. If $\alpha = 1$ the interaction between the bed and fluid is the largest.

1.2.2.3.3 Engelund and Hansen (1972)

Engelund and Hansen (1972) proposed a transport formula for uniform bed material taking into account at the same time the presence of both bed- and suspended-load. This formula is commonly used as a bulk load formula and reads

$$q_B = 0.05\sqrt{(s-1)gd^3} \cdot c_f^2 \theta^{2.5} \tag{1.28}$$

where d denotes the median sediment size of the bed material, c_f the dimensionless Chézy friction coefficient and θ is the dimensionless bed shear stress (see eq. 1.25). The Engelund and Hansen formula for bedload transport does not consider the critical shear stress as threshold condition for incipient motion.

1.2.2.3.4 Smart & Jäggi (1983)

Smart and Jaeggi (1983) developed a bedload transport formula for steep channels using their own experimental results and the results of Meyer-Peter and Müller (Meyer-Peter and Müller, 1948). The specific bedload transport rate q_B is defined as:

$$q_B = \frac{\alpha}{(s-1)} \left(\frac{d_{90}}{d_{30}}\right)^{0.2} J^{0.6} |\vec{q}| (J - J_{cr}) \tag{1.29}$$

where s is the sediment density coefficient ($s = \rho_s/\rho$), $|\vec{q}|$ is the magnitude of the specific discharge and d_{30} and d_{90} are the characteristic grain size diameters, i.e. 30 % resp. 90 % (by weight) of the bed material are smaller. The energy slope J and the critical slope for the initiation of the bedload transport J_{cr} calculated as

$$J = \frac{\theta(s-1)d_m}{h} \tag{1.30}$$

$$J_{cr} = \frac{\theta_{cr}(s-1)d_m}{h} \tag{1.31}$$

where θ is the dimensionless bed shear stress (see eq. 1.25), θ_{cr} the critical dimensionless bed shear stress, d_m the mean grain size diameter and h the water depth. Smart and Jaeggi (1983) recommend values of $\alpha = 4$ and $\theta_{cr} = 0.05$. The scope of applictation is for bed slopes $0.005 \leq$ J ≤ 0.2 (Smart and Jaeggi, 1983).

1.2.2.4 Correction of Bedload Direction

The 2D projection of the solid discharge along x and y is obtained through standard procedures, that are mostly based on empirical basis and which account for the downward effect of gravity on sediment particles due to local bed slope and the presence of spiral flow motion in curved reaches.

1.2.2.4.1 Lateral Bed Slope Effect

Empirical bedload formulas were originally derived for situations where bed slope equals flow direction. However, in case of lateral bed slope with respect to flow direction, the bedload direction differs from the flow direction due to gravity acting on the bed material.

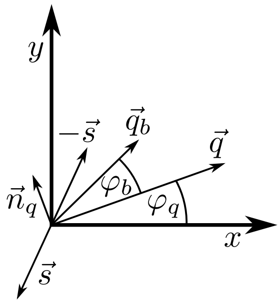


Figure 1.3 *Bed load transport deviation angle φ_b from the flow direction $\vec{q}$ due to the lateral bed slope $\vec{s}$ (Vonwiller, 2017)*

Figure 1.3 illustrates the deviation of the bedload transport direction due to lateral bed slope in a Cartesian coordinate system.

The bedload direction is corrected for lateral bed slope based on the following approach (e.g. see Ikeda (1982) and Talmon et al. (1995)):

$$\tan\varphi_b = -f(\theta)\cdot\ \vec{s}\cdot\ \vec{n}_q \quad \text{for} \quad \vec{s}\cdot\vec{n}_q < 0 \tag{1.32}$$

$$f(\theta) = N_l\left(\frac{\theta_{cr}}{\theta}\right)^{M_l} \tag{1.33}$$

where φ_b = bedload direction with respect to the flow vector $\vec{q}$, N_l = lateral transport factor ($0.75 \leq N_l \leq 2.63$), M_l = lateral transport exponent (typically $M_l = 0.5$), $\vec{s} = \left(\frac{\partial z_B}{\partial x}, \frac{\partial z_B}{\partial y}\right)$ bed slope (positive uphill, negative downhill), $\vec{n}_q$ = unit vector perpendicular to $\vec{q}$ pointing in downhill direction ($\vec{s}\cdot\vec{n}_q < 0$) , θ = effective dimensionless shear stress and θ_{cr} = critical dimensionless shear stress of sediment.

The direction of the bedload transport under the influence of lateral bed slope is written as:

$$\frac{q_{B_y}}{q_{B_x}} = \tan(\varphi_b + \varphi_q) \tag{1.34}$$

1.2.2.4.2 Curvature Effect

Curvature in rivers may cause deviation of the bedload direction from the depth averaged flow direction. Due to three dimensional spiral flow motion, the bedload direction tends to point towards the inner side of the curve, while the flow direction points towards the outer side (Figure 1.4). This curvature effect is taken into account according to an approach proposed by Engelund (1974) , where the deviation angle φ_c of the bottom shear stress $\vec{\tau_b}$ (positive counterclockwise and vice versa) from the main flow direction is determined as

$$\tan\varphi_c = \frac{|\vec{\tau}_{bn}|}{|\vec{\tau}_{bs}|} = -N_*\frac{h}{R} \tag{1.35}$$

where $\vec{\tau}_{bn}$ and $\vec{\tau}_{bs}$ are the bed shear stress normal to and in the flow direction respectively, h denotes the water depth, N_* is a curvature factor, and R denotes the radius of the river

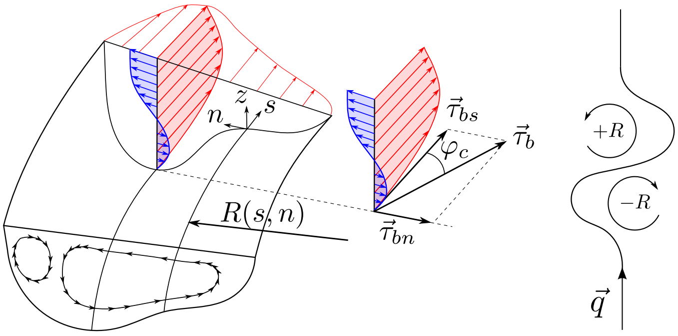


Figure 1.4 *Effect of spiral motion in river bend on bed shear stress $\vec{\tau}_b$ with deviation angle from main flow direction φ_c (Vonwiller, 2017)*

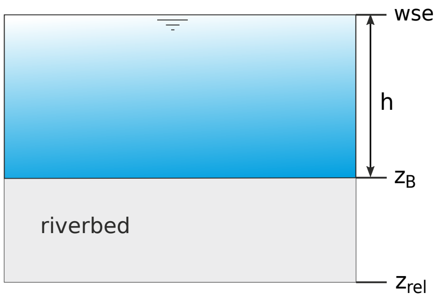


Figure 1.5 *Fixed bed concept and definition*

bend (positive for curvature in counterclockwise direction and vice versa).

Note that the curvature factor N_* mainly depends on bed roughness. Therefore, $N_* \approx 7$ for natural streams (Engelund, 1974), and values up $N_* \approx 11$ for laboratory channels (Rozovskii, 1961).

1.2.2.5 Fixed Bed Concept

Morphodynamic simulations generate deposition and erosion patterns of the riverbed. Erosion processes, if not limited, can proceed indefinitely in the vertical direction. This limit can be imposed by defining a non-erodible fixed bed elevation z_{rel}, below which the river bed is considered as *fixed*. This threshold also determines the amount of sediment available for transport (see Figure 1.5). The fixed bed elevation can either be assigned via regions or via a separate .2dm mesh file. When the fixed bed elevation is specified via regions, the fixed bed elevation must be provided relative to the initial bottom elevation z_B with $z_{rel} \leq 0$. When the fixed bed elevation is specified via a mesh, the elevation in the separate .2dm mesh file must correspond to the absolute fixed bed elevation z_{fix} [m]. Moreover, the fixed bed mesh file must have the exact same topology as the original computational mesh. In case the specified mesh only has elevation information on the nodes, the elevation is interpolated with the same method as specified in the INTERPOLATION block (default: mean). If the fixed bed elevation of the fixed bed mesh exceeds the bottom elevation of the computational mesh, the fixed bed elevation is defined at the elevation of the computational mesh.

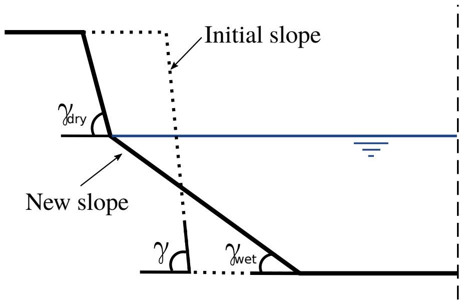


Figure 1.6 *Critical failure angles for slope collapse*

The accuracy of the fixed bed correction is guaranteed by defining the maximal overshoot below the fix bed elevation and the maximal number of iterations required for the correction.

1.2.2.6 Gravitational Transport

Gravitational induced riverbank or sidewall failures are significant aspects concerning erosion and transport modelling. Such processes may play an important role in many situations, such as meandering streams, river widenings or failures of erodible embankment structures due to overtopping waters. Such slope failure processes take place mostly discontinuous and can deliver significant contributions to the total sum of transported material. The modes of slope failures can differ largely (falls, topples, slides, etc.) and depend on the soil material, the degree of soil compaction and the pore pressures within the soil matrix. Here, a simplified, geometric approach is applied to be able to consider some aspects of this purely gravitational induced transport. The main idea of the implemented geometrical approach is to assume that a slope failure takes place if the local bed slope γ becomes steeper than a critical slope γ_{cr} (Figure 1.6).

$$q_{B,grav} = \begin{cases} 0 & \text{if}\,(\gamma \leq \gamma_{cr}) \\ f(\gamma, \gamma_{cr}) & \text{if}\,(\gamma > \gamma_{cr}) \end{cases} \tag{1.36}$$

The sliding material is moved from the sediment element with higher elevation to the lower situated element until the stable condition: $\gamma \leq \gamma_{cr}$ is reached. Two characteristic critical slope angles are defined in this approach to have some flexibility in modelling the complex geotechnical aspects. The critical angles can be characterized as:

- critical angle for dry or partially saturated bank material γ_{dry}, which may greatly exceed the material's angle of repose (up to nearly vertical walls) due to negative pore pressures,
- critical angle for fully saturated and over flown material γ_{wet} which is in the range of the material's angle of repose

1.2.2.6.1 Calculation Proceedure

The flux due to gravitational transport $q_{B,grav}$ is calculated with the following proceedure, by looping over each element of the computational grid:

1. In a first steep, the local bed slope γ_i is calculated with respect to each neighbouring element i, where z is the bed elevation of the main element, z_i is the bed elevation of the neighbour element i and d_i is the distance between the element centers.

$$\gamma_i = arctan\left(\frac{z - z_i}{d_i}\right) \tag{1.37}$$

2. For local bed slopes γ_i exceeding the critical slope γ_{cr}, the new bed elevations z_{new} and $z_{i,new}$ are determined such that the stable condition $\gamma_i = \gamma_{cr}$ is reached for all neighbouring cells i. The critical angle γ_{cr} is selected according to eq. 1.38, where h is the water depth in the main element and h_{min} is the user-specified minimum water depth.

$$\gamma_{cr} = \begin{cases} \gamma_{wet} & \text{if}\, h \geq h_{min} \\ \gamma_{dry} & \text{if}\, h < h_{min} \end{cases} \tag{1.38}$$

3. If the calculated bed elevation change of the main element $\delta_z = z - z_{new}$ is smaller than the user-specified parameter *min_bed_change* (default: 0.001 m) $\delta_{z,min}$, no gravitational transport occurs to avoid oscillatory behaviour and to reduce the computational effort. If the minimum bed elevation change $\delta_{z,min}$ is exceeded, the specific gravitational flux $q_{B,grav,i}$ [m^2/s] to each neighbour element i is calculated from the bed elevation change in element i according to eq. 1.39, where A_i is the area of element i, and l_i is the length of the edge connecting the main element to its i^{th} neighbour element. If the calculated bed elevation change of the main element $\delta_z = z - z_{new}$ is larger than the user-defined maximum bed elevation change $\delta_{z,max}$, the gravitational flux is limited, such that $\delta_z = \delta_{z,max}$. The maximum bed elevation change $\delta_{z,max}$ is calculated by eq. 1.40, where $r_{b,max}$ is the user-specified parameter *max_bed_change_rate* and Δt is the current update time step.

$$q_{B,grav,i} = \begin{cases} 0 & \text{if}\, \delta_z < \delta_{z,min} \\ \frac{(z_{i,new} - z_i) \cdot A_i}{l_i} & \text{if}\, \delta_z \geq \delta_{z,min} \\ \frac{(z_{i,new} - z_i) \cdot A_i}{l_i} \cdot \frac{\delta_{z,max}}{\delta_z} & \text{if}\, \delta_z \geq \delta_{z,max} \end{cases} \tag{1.39}$$

$$\delta_{z,max} = r_{b,max} \cdot \Delta t \tag{1.40}$$

4. If a non-erodbile fixed bed elevation z_{fix} is specified, the gravitational transport flux is corrected the same way as the bedload transport flux (see Section 1.2.2.5).

5. Finally, the balancing of the gravitational fluxes and the determination of the new bed elevations z is achieved by solving the Exner equation using the same numerical approaches as outlined for the bed load transport. This procedure ensures that fixed bed elevations are taken into account and the mass continuity is fulfilled.

1.2.2.6.2 Time Scale of the Gravitational Transport Process

Since the sediment movement due to gravitational transport during one update time step is limited to adjacent elements, it may take many update time steps to reach a stable condition on a larger scale, e.g. a bank slope spanning over multiple elements. The time scale until a globally stable condition is reached, is influenced by the update time step (update_time), the maximum bed change rate (max_bed_change_rate) and the grid resolution. The parameter *update_time* (default: 0.0 s) determines at which frequency the gravitational transport proceedure (steps 1-5 above) is executed. Generally, a smaller update time step reduces the time scale until a globally stable condition is reached. The default behaviour is to set the update time step value to 0.0 seconds, which results in the gravitational transport proceedure being executed at the same time step as determined from the hydraulic CFL-criterium (see Section 2.3.4).

Furthermore, the speed of the gravitational transport process can be limited with the parameter *max_bed_change_rate* (default: 1.0 m/s). This parameter represents a maximum rate at which the bed elevation of a cell can be lowered due to gravitational transport and determines the maximum bed elevation change during one update time step. Generally, a smaller value may increase the time scale until a globally stable condition is reached.

Since sediment movement due to gravitational transport is limited to adjacent elements, also the grid resolution effects the time scale until a globally stable condition is reached. A finer grid resolution (smaller elements) increases the necessary number of update cycles the reach a stable slope over a specific length. However, a finer grid resolution may also decrease the hydraulic time step and therefore may the increase frequency at which the gravitational transport procedure is executed.

1.2.2.6.3 Bed slope at Stable Condition

The implemented approach for modelling gravitational transport processes ensures that the local bed slope does not exceed the user-specified critical angle. Due to the spatial discretization with constant bed elevations within an element, the local bed slope is calculated between two element centers. On a larger scale (e.g. a bank slope spanning over multiple elements), the bed slope may deviate from the user-specified critical angle after reaching stable conditions. The stable bed slope over multiple elements, generally exceeds the critical bed slope.

To illustrate this effect, a simple test case was simulated. The rectangular computational domain is initially split into two parts: the left side with a bed elevation of 2 meters and the right side with a bed elevation of 0 meters. The domain is completely dry and the critical angle for dry material is set to $\gamma_{dry} = 35°$. The simulation only considers only gravitational tranport and is stopped after a stable condition is reached. The initial and final bed elevation profile along the centerline are illustrated in Figure 1.7. The initial profile exhibits the nearly vertical bed step. The final profile at stable conditions exhibits a slope of approximately 41°, which exceeds the critical angle of $\gamma_{dry} = 35°$. The reason for this deviation on a larger scale is that the distance which is considered for the calculation of the slope is not a straight line, but a line connecting the element centers. This is illustrated in Figure Figure 1.8. Calculating the large scale slope considering the length of the green line results in a slope of approximately 41°, while considering the length of the red line results in a slope of approx. 35°.

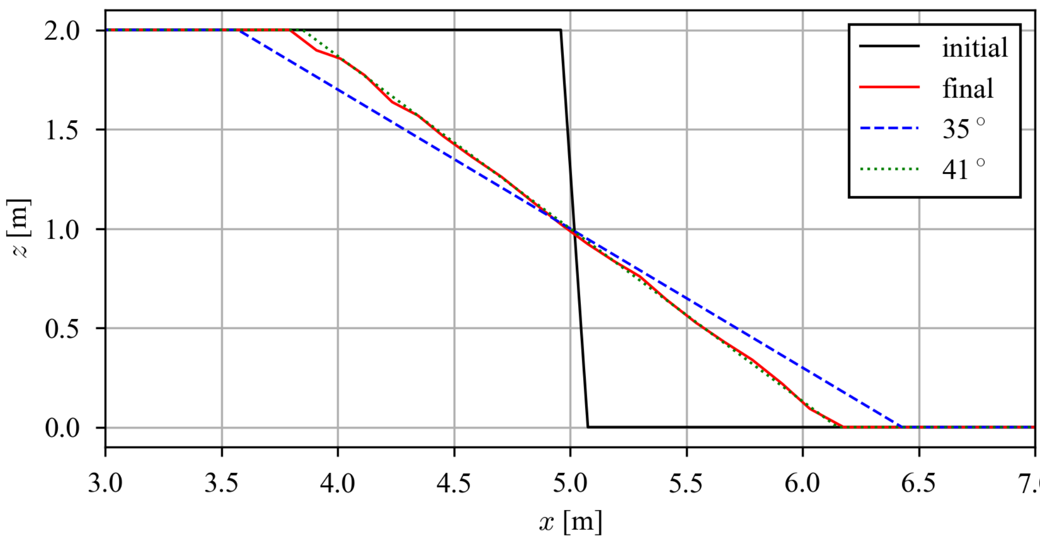


Figure 1.7 *Initial and final bed elevation profiles along the centerline are compared to profiles with angles of 35° and 41°. The slope of the bed profile at stable conditions (final) exceeds the critical angle of* γ_{dry} *= 35°.*

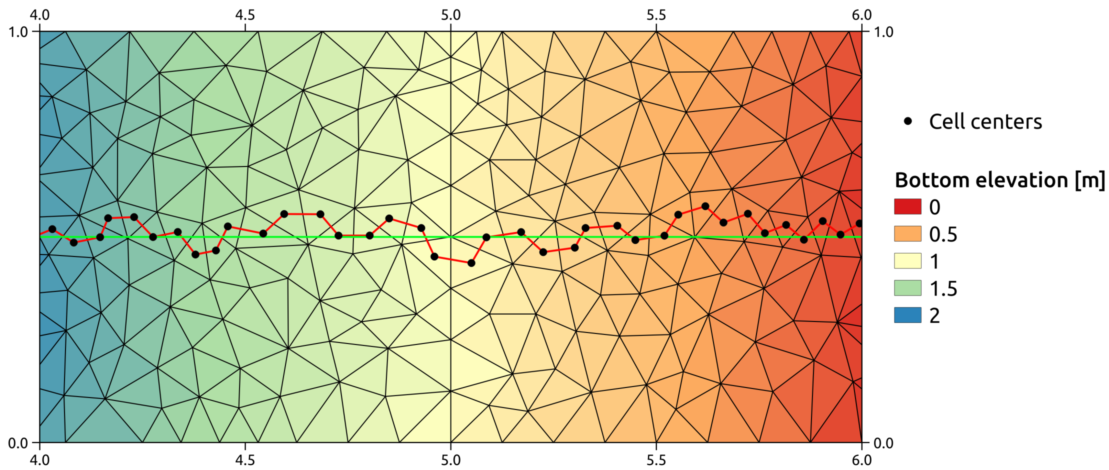


Figure 1.8 *The red line connects the element centers and represents the distance, which is taken into account for the* ***local slope*** *calculation and thus for the gravitational transport. The green line represents the direct distance, which is taken into account for the* ***large scale slope****. The slope along the red line corresponds to 35°, while the slope along the green line corresponds to 41°.*

1.2.2.7 External Sources Terms

The source term Sl_b represents additional sediment mass input or output (sink) that occurs locally on the computational domain on a set of elements defined by regions. The source can be specified as total volume flux including porosity $[m^3/s]$. Different approaches are used to manage the behaviour of the external sources in case of a negative flux (sink):

- Exact: The specified sediment volume is added or extracted (non conservative)
- Available: The specified sediment volume to extract is limited by the defined fixed bed elevation of the elements (conservative)
- Infinity: All available sediment will be abstracted (conservative)

Addition of sediment always follows the "Exact" behaviour as there is no upper limit. The abstraction of sediment could also follow the "Exact" behaviour but the simulation might end abruptly if the available sediment volume is smaller than the volume abstracted. Therefore, the "Available" behaviour aims to avoid this situation. The "Infinity" behaviour abstracts all available sediment volume.

1.2.2.8 Boundary Conditions

After the specification of the *closure relations* for the sediment transport, the system of governing equations (eq. 1.1) and (eq. 1.15) can be solved within the modeling domain described in Figure 1.1, provided boundary conditions (morphologic boundary conditions) are specified at the domain boundary Γ. For the sediment transport only *external boundaries* that allow sediment flowing into or out of the domain can be specified. A morphologic boundary condition can be co-located with a hydraulic boundary condition. In case no hydraulic boundary condition is specified, the boundary will behave as a wall and sediment transport will not occur.

1.2.2.9 Upstream Boundary Condition

- *equilibrium_in*: After erosion or deposition up to a user specified reference bed elevation (*reference_bed_elevation*) this upstream boundary condition grants a equilibrium condition, i.e. the same amount of sediment leaving the first computational cell in flow direction enters the cell from the upstream boundary. This leads to a constant bed elevation at the boundary condition.
- *sedimentograph*: based on a sediment hydrograph describing the bedload inflow as function of time (constant or variable). The bedload is defined as a volumetric flow rate $Q_b = \dfrac{\mu_s}{\rho_s}$ $[m^3/s]$, where μ_s is the sediment mass flow rate $[kg/s]$ and ρ_s the sediment density $[kg/m^3]$. Notice that the porosity is not considered in the bedload input and is specified separately as own parameter value. The volumetric flow rate is either distributed using a geometrical weighting (*sedimentograph*), using wetted area weighting (*sedimentograph_warea*) or using wetted conveyance weighting (*sedimentograph_conveyance*). **Note:** When using the pre-factor (*factor*) described in Section 1.2.2.3, it is automatically applied to the volumetric flow rate at the boundary for these types of boundary conditions, i.e. the sediment hydrograph is scaled by the pre-factor.

- *transport_capacity*: the sediment inflow is defined by calculating the equilibrium transport capacity according to the hydraulic state at the boundary. The bedload is defined as a compact volumetric flow rate (without porosity) Q_b $[m^3/s]$. The volumetric flow rate is either distributed using a geometrical weighting (*transport_capacity*), using wetted wetted area weighting (*transport_capacity_warea*) or using wetted conveyance weighting (*transport_capacity_conveyance*). **Note:** When using the pre-factor (*factor*) described in Section 1.2.2.3, the pre-factor is implicitly included in the volumetric flow rate at the boundary for this type of boundary condition, i.e. the transport capacity at the boundary is scaled by the pre-factor. Additionally, an independent scaling factor can be specified (*boundary_factor*), only applying to these types of the boundaries.

For the sediment discharge and transport capacity boundary condition types, the specific sediment discharge q_b is distinguished by three weighting schemes:

1. Geometrical weighting with respect to the total nodestring length L_n.

$$q_b = \frac{Q_b}{L_n} \qquad \left[\frac{m^3}{s \cdot m}\right] \tag{1.41}$$

2. Wetted area weighting

$$q_b = \frac{Q_b}{A_{w,tot}} \cdot h \qquad \left[\frac{m^3}{s \cdot m^2}\right] \tag{1.42}$$

3. Conveyance weighting

$$q_b = \frac{Q_b}{K_{tot}} h\sqrt{c_f h} \qquad \left[\frac{m^3}{s \cdot m}\right] \tag{1.43}$$

with $K_{tot} = A_{w,tot}\sqrt{c_f h}$ the total conveyance and c_f the friction coefficient.

1.2.2.10 Downstream Boundary Condition

One downstream boundary condition is available:

- *equilibrium_out*: After erosion or deposition up to a user specified reference bed elevation (*reference_bed_elevation*) this downstream boundary condition grants a equilibrium condition, i.e. all sediment entering the last computational cell will leave the cell over the downstream boundary. This leads to a constant bed elevation at the boundary condition.

1.2.2.11 Linked Boundary Condition

One linked boundary condition is available:

- *equilibrium_linked*: At the upsteam boundary, erosion or deposition is possible up to a user specified reference bed elevation (*reference_bed_elevation*). After reaching the reference elevation, this boundary condition grants a equilibrium condition, i.e. all sediment leaving the computational cells on the upstream side is entering at the downstream boundary with a lag of one timestep. This leads to a constant bed elevation at the upstream boundary.

1.3 Passive tracers

1.3.1 Introduction

A multitude of dissolved species are present in environmental flows. In the context of hydraulic and environmental engineering, numerical modelling of scalar transport becomes a relevant tool mostly because of its versatility. In terms of advective phenomena, the most common applications include

- Pollutant fate and transport
- Accumulation or depletion of nutrients
- Calculation of water residence times
- Flow visualization

1.3.2 Transport of passive species

1.3.2.1 Governing Equations for passive specie transport

The governing equations are obtained under the shallow water framework and impose mass conservation for both fluid and dissolved phases.

In a Cartesian frame of reference $(x; y; z)$ in which the z axis is vertical and the horizontal lies in the $x - y$ plane, the system of governing equations is formed by (eq. 1.1) for hydrodynamics and is coupled with multiple equations for the conservation of the total tracer masses. The conservation of each tracer mass is ensured by the scalar continuity equation (eq. 1.44), which is tightly coupled to the shallow water equations. This equation allows to describe the evolution of the specie concentration as:

$$\frac{\partial q_S}{\partial t} + \frac{\partial}{\partial x}\left(\frac{q_x q_S}{h}\right) + \frac{\partial}{\partial y}\left(\frac{q_y q_S}{h}\right) - S_s = 0 \tag{1.44}$$

where $q_S = h\phi_s$ is the volume of specie s per unit width, with ϕ_s being it's volumetric concentration, and S_s is the source term per unit width, specifying local input or output of the specie s.

1.3.2.2 External Sources Terms

The source term S_s conveys an input or output (sink) that occurs locally on the computational domain over elements limited by regions. It can be specified as total volumetric flux $[m^3/s]$. Different approaches define the behaviour of external sources if it imposes negative fluxes (sink):

- Exact: The specified volume is added or extracted (non conservative)
- Available: The specified volume to extract is limited by the available tracer volume (conservative)
- Infinity: All available tracer volume will be extracted (conservative)

Additionally, there is the option of forcing a target concentration ϕ_s^f homogeneously across all cells defined by the region. In such case the value of q_s is set directly as $h\phi_s^f$ (non conservative).

1.3.2.3 Boundary Conditions

The system of governing equations (eq. 1.1) and (eq. 1.44) can be solved within the modeling domain described in Figure 1.1, provided that boundary conditions (hydrodynamical and tracer boundary conditions) are specified at the domain boundary Γ. For the tracer transport only *external boundaries* that allow tracer flowing into or out of the domain can be specified. A tracer boundary condition should be co-located with a hydraulic boundary condition. Otherwise, the boundary will behave as a wall and tracer transport will not occur.

1.3.2.4 Upstream Boundary Condition

Two upstream boundary conditions are available:

- *discharge_in* and *discharge_in_warea*: based on a tracer discharge inflow, either as a constant or a function of time, the prescribed volumetric flow rate Q_s $[m^3/s]$ is imposed at the boundary. The volumetric flow rate is either distributed using a geometrical weighting or a wet-area weighting, and the specific tracer discharge q_s is thus given by:

1. Geometrical weighting with respect to the total nodestring length L_n.

$$q_s = \frac{Q_s}{L_n} \qquad \left[\frac{m^3}{s \cdot m}\right]$$

2. Wetted-area weighting

$$q_s = \frac{Q_s}{A_{w,tot}} \cdot h \qquad \left[\frac{m^3}{s \cdot m}\right]$$

- *concentration_in*: the tracer inflow is defined by forcing a target tracer concentration ϕ_s^f. The volumetric flow rate is thus given by the target concentration paired with the total hydrodynamic mass flux q through that boundary as $q_s = q\phi_s^f$.

Note: If the hydrodynamical mass flow at the boundary is not inward directed then no tracer flux is imposed.

1.3.2.5 Downstream Boundary Condition

One downstream boundary condition is available:

- *zero_gradient_out*: this downstream boundary allows the free outflow of any tracer quantities in the flow. **Note:** If this condition is not prescribed, a wall condition is assumed and the tracer quantities will be retained (no outflow).

2

Numerical Models

2.1 General View

The governing equations of hydro- and morphodynamics are conservation laws expressing conservation of mass and momentum. The aim of the numerical simulation is to solve these equations over the computational domain and for a given time. The computational domain is discretized by a computational mesh (Figure 2.1) consisting of elements (often having triangular shape) and conservation equations are applied on each domain element. In order to numerically solve the conservation equations, the mathematical model is approximated by numerical schemes, i.e. the numerical approximation consists of the spatial and temporal discretization of the conservation equations including an algorithm that solves the discretized equations.

The conservation equations can be formulated either in integral or differential form. Different numerical schemes exist to discretise the equations:

- Finite difference: The discrete values are considered as point values defined at mesh points
- Finite element: The discrete values are determined in terms of the nodal values of the mesh
- Finite volume: The discrete values are averaged over finites volumes of the mesh

In BASEMENT, the spatial discretisation of the domain is based on an unstructured mesh made of triangular elements. For the conservation equations, the spatial discretisation follows the finite volume scheme, while for the temporal discretisation an explicit first order Euler scheme is used. The numerical model processes the hydro- and morphodynamic equations in a decoupled way (Figure 2.2).

The discretization and the solution method for the hydro- and morphodynamic equations will be presented in the following sections.

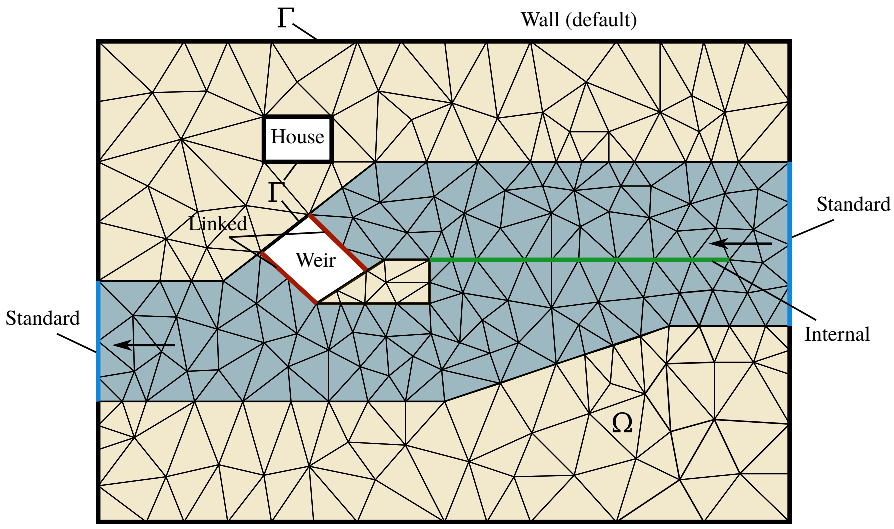


Figure 2.1 *Modeling domain, types of boundary conditions and computational mesh. The flow is from right to left and a side weir (green line) divides the channel into a lower and an upper channel through the weir.* External *boundary conditions must be provided at* Γ_1*,* Γ_2 *and* Γ_3 *while* internal *boundary conditions can be specified in any place within* Ω

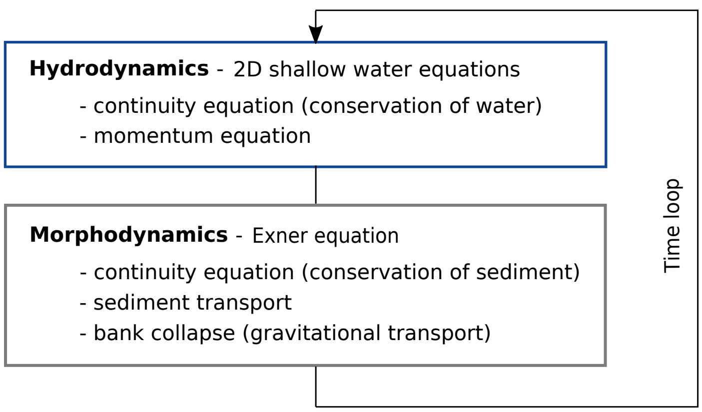


Figure 2.2 *Overview of the numerical model*

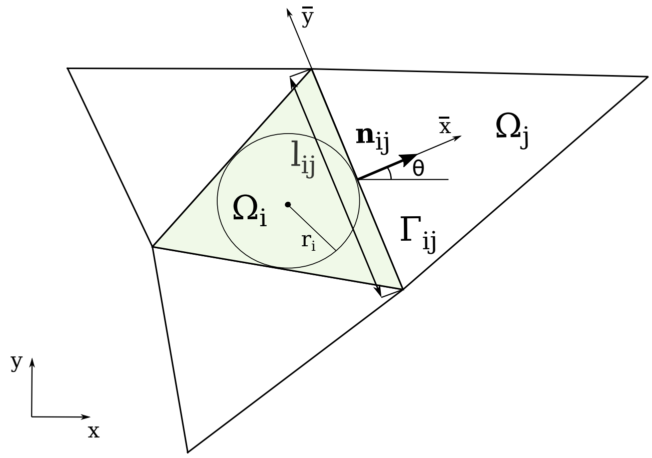


Figure 2.3 *Element (shaded triangle) of unstructured triangular mesh and used notation*

2.2 Discretization

The problem is discretised adopting a finite volume approach over unstructured triangular meshes. A conforming triangulation T_Ω of the computational domain $\Omega \subset \mathbb{R}^2$ by elements Ω_i such that $T_\Omega = \bigcup \Omega_i$, is assumed. Hereafter we will use the following notation: given a finite volume Ω_i, $j = 1, 2, 3$ is the set of indexes such that Ω_j is a neighbour of Ω_i; Γ_{ij} is the common edge of two neighbour cells Ω_i and Ω_j, and l_{ij} its length. $\mathbf{n}_{ij} = (n_{ij,x}, n_{ij,y})$ is the unit vector which is normal to the edge Γ_{ij} and points toward the cell Ω_j (see Figure 2.3). Data are represented by cell averages $\boldsymbol{U}_i^n$ and the numerical solution sought at time $t^{n+1} = t^n + \Delta t$, is denoted by $\boldsymbol{U}_i^{n+1}$.

2.3 Numerical solution of Hydrodynamics

2.3.1 Vectorial Form of the Governing Equations

For numerical convenience, the system of governing equations (eq. 1.1) is rewritten in vectorial form in terms of the water surface elevation $H = h + z_B$. It now reads:

$$\frac{\partial \boldsymbol{U}}{\partial t} + \frac{\partial \boldsymbol{F}_x}{\partial x} + \frac{\partial \boldsymbol{F}_y}{\partial y} = \boldsymbol{S} \tag{2.1}$$

where the vector of unknowns is

$$\boldsymbol{U} = \begin{pmatrix} H \\ q_x \\ q_y \end{pmatrix} \tag{2.2}$$

the vector fluxes are

$$\boldsymbol{F}_x = \begin{pmatrix} q_x \\ uq_x + \frac{1}{2}g(H^2 - 2Hz_b) \\ uq_y \end{pmatrix} \quad ; \quad \boldsymbol{F}_y = \begin{pmatrix} q_y \\ vq_x \\ vq_y + \frac{1}{2}g(H^2 - 2Hz_b) \end{pmatrix} \tag{2.3}$$

and the vector of source terms is

$$\boldsymbol{S} = \begin{pmatrix} S_h \\ gHS_x \\ gHS_y \end{pmatrix} \ . \tag{2.4}$$

The motivation of using H instead of h lies in the fact that it is easier to develop numerical schemes which preserve depth positivity and satisfy the well-balanced property.

2.3.2 Spatial Discretisation

In order to discretise the system of governing equations, the domain is meshed by a set of triangular elements. The spatial discretization of the conservation equations is carried out by the finite volume method, where the differential equations are integrated over the single elements, i.e. control volumes. The water surface elevation is defined at the element center and is equally distributed over the element.

By integrating the governing system of equations eq. 2.1 in the control volume $V = [\Omega_i] \times [t^n, t^{n+1}]$, we obtain

$$\boldsymbol{U}_i^{n+1} = \boldsymbol{U}_i^n - \frac{\Delta t}{|\Omega_i|} \sum_{j=1}^{3} l_{ij} \left[\boldsymbol{F}_{ij}\right] + \Delta t \boldsymbol{S}_i \ . \tag{2.5}$$

2.3.3 Flux Estimation

2.3.3.1 Rotational Invariance of the Shallow Water Equations

The flux $\boldsymbol{F}_{ij}$ are evaluated taking advantage of the rotational invariance property of the shallow water equations. According to this property the two-dimensional homogeneous shallow water equations satisfy the following equality (Toro, 2009):

$$\boldsymbol{n}_{ij} \cdot [\boldsymbol{F}_x(\boldsymbol{U}), \boldsymbol{F}_y(\boldsymbol{U})] = \boldsymbol{T}^{-1}(\theta)\boldsymbol{F}_x[\boldsymbol{T}(\theta)\boldsymbol{U}] \tag{2.6}$$

where θ is the angle between the vector $\boldsymbol{n}_{ij}$ and x-axis, measured counter clockwise from the x-axis (see Figure 2.3) and

$$\boldsymbol{T}(\theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{pmatrix} \tag{2.7}$$

being

$\boldsymbol{T}^{-1}(\theta) =$ inverse of $\boldsymbol{T}(\theta)$.

2.3.3.2 Computation of the Flux

The flux $\boldsymbol{F}_{ij}$ is obtained at every edge of the finite volume mesh, as the solution of the one-dimensional projected Riemann problem along the normal direction of the two conservation laws eq. 2.1. The computational steps can be summarized as follows:

- First, the vector of conserved variables $\boldsymbol{U}$ is transformed into the local coordinate system $(\bar{x}, \bar{y})$ (see Figure 2.3) at the edge with the operation $\boldsymbol{T}(\theta)\boldsymbol{U}$.
- A one-dimensional, local Riemann problem is formulated and solved in the normal direction of the edge. From this calculation the new flux vector over the edge $\boldsymbol{F}[\boldsymbol{T}(\theta)\boldsymbol{U}]$ is defined.
- The flux vector, formulated in the local coordinate system is transformed back to the global coordinates (Cartesian) with $\boldsymbol{T}^{-1}\boldsymbol{F}[\boldsymbol{T}(\theta)\boldsymbol{U}]$. The sum of the fluxes of all edges of an element gives the total fluxes in the x- and y directions.

The fluxes are calculated in the normal direction of the element edges. The normal direction of the edge is defined positive from element i (L) to element j regarding the edge direction.

2.3.3.3 The HLLC approximated Rieman Solver

The HLLC approximate Riemann solver (Toro, 1994) is a modified HLL (Harten, Lax and van Leer) approximate Riemann solver that includes the shear wave.

The numerical flux at the cell interface is computed as follows:

$$\boldsymbol{F}_{ij}^{HLLC} = \begin{cases} \boldsymbol{F}_i & if \quad 0 \leq S_i, \\ \boldsymbol{F}_{*i} = \boldsymbol{F}_i + S_i(\boldsymbol{U}_{*L} - \boldsymbol{U}_i) & if \quad S_i \leq 0 \leq S_*, \\ \boldsymbol{F}_{*j} = \boldsymbol{F}_j + S_j(\boldsymbol{U}_{*R} - \boldsymbol{U}_j) & if \quad S_* \leq 0 \leq S_j, \\ \boldsymbol{F}_j & if \quad 0 \geq S_j. \end{cases} \tag{2.8}$$

The wave speed velocities are estimated as:

$$S_i = u_i - \sqrt{gh_i}\xi_i \; ; \; S_j = u_j + \sqrt{gh_j}\xi_j \tag{2.9}$$

where $\xi_{K=(i,j)}$ is defined as:

$$\xi_K = \begin{cases} \sqrt{\frac{1}{2}\left[\frac{(h_*+h_K)h_*}{h_K^2}\right]} & if \quad h_* > h_K, \\ 1 & if \quad h_* \leq h_K. \end{cases} \tag{2.10}$$

with h_*, an estimate for the exact solution of the water depth in the star region obtained using the depth positivity condition. It reads as

$$h_* = \frac{1}{2}(h_L + h_R) - \frac{1}{4}(u_R - u_L)(h_L - h_R)/(\sqrt{gh_L} + \sqrt{gh_R}) \tag{2.11}$$

In case of dry-bed conditions, the wave speeds are estimated as the exact dry front speed, i.e.:

$$S_i = \begin{cases} u_i - 2\sqrt{gh_i} & if \quad h_i = 0, \\ \text{usual estimate} & if \quad h_i > 0, \end{cases}$$
$$S_j = \begin{cases} u_j + 2\sqrt{gh_j} & if \quad h_j = 0, \\ \text{usual estimate} & if \quad h_j > 0. \end{cases} \tag{2.12}$$

And the middle estimated wave speed S_* corresponds to the front wave speed in case of dry-bed problem.

The expression of the states $\boldsymbol{U}_{*i}, \boldsymbol{U}_{*j}$ and the middle wave speed S_* can be found in the book of Toro (2009).

2.3.4 Numerical Stability

Numerical stability is assured by choosing the time step Δt for time integration such that it obeys the Courant-Friedrichs-Lewy (CFL) condition. In 2-D the Courant number (CFL) can be defined as follows:

$$CFL = \frac{(\sqrt{u^2 + v^2} + c)\Delta t}{r_i} \tag{2.13}$$

where r_i is the radius of the inscribed circle that defines the element center (Figure 2.3), u, v are the corresponding velocities of the element and $c = \sqrt{gh}$. The HLLC scheme is stable for

$$0 < CFL \leq 1 \tag{2.14}$$

2.3.5 Discretisation of Source terms

2.3.5.1 Bed Slope Source Term

The bed slope source term (eq. 1.2) is discretized using the robust modified-state approach proposed by Duran et al. (2013). The discretization presents a motionless steady states-preserving scheme:

$$\boldsymbol{S}_{b,i} = \sum_{j=1}^{m} l_{ij} \boldsymbol{S}_{b,ij} = \sum_{j=1}^{m} l_{ij} \begin{pmatrix} 0 \\ gH^*_{ij}(z_i - \bar{z}_{ij})\vec{\boldsymbol{n}}_{ij} \end{pmatrix} \tag{2.15}$$

where $\bar{z}_{ij} = \check{z}_{ij} - \Delta_{ij}$ with $\check{z}_{ij} = \max(z_{bi}, z_{bj})$ the maximum bed elevation between cells i and j and $\Delta_{ij} = \max(0, \check{z}_{ij} - H_i)$. H^*_{ij} is the approximated value of the water surface elevation H at the cell interface Γ_{ij}.

2.3.5.2 Friction Source Term

We handle the inhomogeneous character of system eq. 1.1 due to the presence of frictional source terms by adopting a robust splitting technique Toro (2001). We initially consider the initial value problem (IVP)

$$\left.\begin{array}{llll} PDE: & \mathcal{A}(\boldsymbol{U}) & = & \mathcal{S}(\boldsymbol{U}) \\ IC: & \boldsymbol{U}(x,y,0) & = & \boldsymbol{U}_i^n \end{array}\right\}\text{IVP} .$$

where $\mathcal{A}$ represents the advective operator

$$\mathcal{A}(\boldsymbol{U}) = \frac{\partial \boldsymbol{U}}{\partial t} + \frac{\partial \boldsymbol{F}_x}{\partial x} + \frac{\partial \boldsymbol{F}_y}{\partial y} = \boldsymbol{0} ,$$

and $\mathcal{S}$ represents the frictional source term operator.

The numerical solution is then obtained by subsequently integrating *two* initial value problems (IVPs):

$$\left.\begin{array}{llll} ODEs: & \frac{\mathrm{d}\boldsymbol{U}}{\mathrm{d}t} & = & \mathcal{S}(\mathbf{U}) \\ ICs: & \boldsymbol{U}(x,y,0) & = & \boldsymbol{U}_i^n \end{array}\right\} \stackrel{\Delta t}{\Longrightarrow} \overline{\boldsymbol{U}}_i \quad \text{IVP1} ,$$

$$\left.\begin{array}{llll} PDEs: & \mathcal{A}(\mathbf{U}) & = & 0 \\ ICs: & \boldsymbol{U}(x,y,0) & = & \overline{\boldsymbol{U}}_i \end{array}\right\} \stackrel{\Delta t}{\Longrightarrow} \boldsymbol{U}_i^{n+1} \quad \text{IVP2} ,$$

The initial condition (IC) for IVP1 is $\mathbf{U}_i^n$, corresponding to the initial condition of the full problem IVP. The solution of IVP1 is obtained solving a system of ordinary differential equations (ODEs) after integration by a time step Δt and is denoted by $\overline{\boldsymbol{U}}_i$. IVP2 is then integrated by a time step Δt, with initial condition given by the solution of IVP1 $\overline{\boldsymbol{U}}_i$. The solution of IVP2 $\boldsymbol{U}_i^{n+1}$ is obtained solving an hyperbolic homogeneous system of partial differential equations (PDEs) and represents the approximate solution of the full problem IVP.\ Since we adopt an implicit second-order Runge-Kutta method for solving the ODEs systems IVP1 and an explicit finite volume method for solving IVP2, the integration time step Δt is determined accordingly with the CFL stability condition for IVP2.

2.3.5.3 External Source Term

An external source is defined as specific mass flux δ (m/s), uniformly distributed over a number of elements of the domain with a specific surface area. The external source can either be specified as discharge (m^3/s) or precipitation intensity (mm/h) for a specific region of the domain. The external source value is divided among the cells composing the region and converted to cell specific mass flux δ_i. The volume allocated is characterized by different behaviors:

$$\begin{array}{llll} \text{Exact:} & S_{h,i} = \delta_i & & \\ \text{Available:} & S_{h,i} = \delta_i & \text{if} & \delta_i \cdot \Delta t > 0 \\ & S_{h,i} = \max(\delta_i, -h_i) & \text{if} & \delta_i \cdot \Delta t < 0 \\ \text{Infinity:} & S_{h,i} = \delta_i & \text{if} & \delta_i \cdot \Delta t > 0 \\ & S_{h,i} = -h_i & \text{if} & \delta_i \cdot \Delta t < 0 \end{array} \tag{2.16}$$

Where h_i is the water depth of the element i. The external source volume is added to the initial water volume.

$$h_i^{t+1} = h_i^t + S_{h,i} \cdot \Delta t \tag{2.17}$$

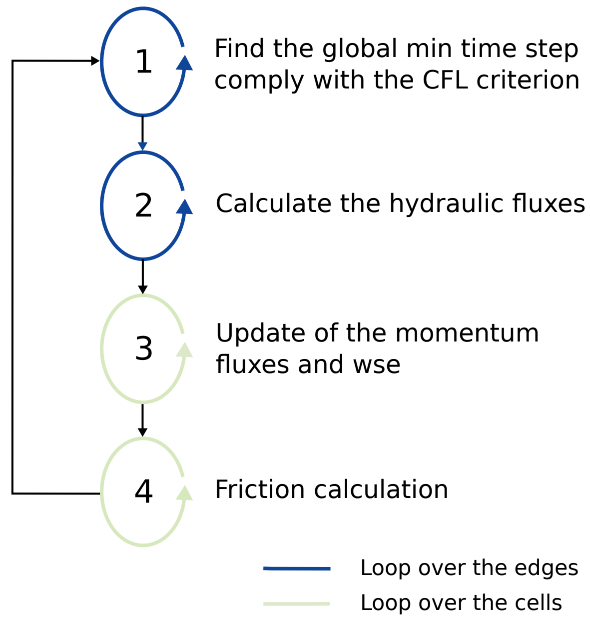


Figure 2.4 *Numerical solution procedure of hydrodynamic simulation for each time step* Δt

2.3.6 Solution Procedure

The numerical solution procedure of BASEMENT explains how the discretised shallow water equation (eq. 1.1) is solved inside a defined time step Δt through a sequence of loops over the edges or cells (Figure 2.4).

First, a global minimum time step Δt should be defined. Then, the hydraulic fluxes (liquid mass, x-momentun and y-momentum) are calculated with a HLLC Riemann solver at the element edges according to the initial states of the left and right cells (Section 2.3.3). Subsequently, the hydraulic state variables i.e. cell centered quantities are updated and finally, the friction (source term) is calculated using an implicit scheme, thus looping twice over the cell.

2.4 Numerical solution of Morphodynamics

2.4.1 Numerical solution of the Exner equation

2.4.1.1 Fundamentals

The Exner equation assures that sediment mass is conserved in the bed and is used to model the riverbed time evolution. The rate of sediment transport is determined using a closure equation. The cell centered finite volume approach is used to discretise the Exner equation and in particular the HLL approximate Riemann solver with a wave speed estimator defined in Soares-Frazão and Zech (2011) is adopted. The shallow water and the Exner equations create a system of equations that is solved in a decoupled way (Figure 2.2). This approach makes the assumption that the bed load flux is much slower than the water flow velocity (Soares-Frazão and Zech, 2011).

2.4.1.2 Spatial discretization

In order to discretise the the Exner equation, we use the same unstructured mesh adopted for the hydrodynamic part and the same finite volume approach. As a consequence, the bed level z_B is defined at the element center and is equally distributed over the element.

By integrating the Exner equation in the control volume $V = [\Omega_i] \times [t^n, t^{n+1}]$, we obtain

$$z_{Bi}^{n+1} = z_{Bi}^{n} - \frac{\Delta t}{|\Omega_i|} \sum_{j=1}^{3} \left[q_{Bij \cdot l_{ij}} \right] + \Delta t \boldsymbol{S}_i \ . \tag{2.18}$$

The calculation of the sediment flux at the cell interface proceeds as follows:

1. loop over the cells and calculate:

    1. correction terms for the bed-load vector directions (if selected by the user), therefore:

    - calculation of the local bed slope, for the lateral-transport correction (see section Section 1.2.2.4.1)
    - calculation of the local curvature of the flow field, for the spiral flow correction (see section Section 1.2.2.4.2)

2. loop over the cell interfaces and:

    1. calculate the flux projection along the normal vector ($n_{ij,x}$,$n_{ij,y}$ of edge Γ_{ij}, i.e.: $q_{Bi,n} = q_{Bi,x} \cdot n_{ij,x} + q_{Bi,y} \cdot n_{ij,x}$ and $q_{Bj,n} = q_{Bj,x} \cdot n_{ij,x} + q_{Bj,y} \cdot n_{ij,x}$ with j=1,2,3
    2. compute the flux at the interface using the approximate HLL Riemann solver at the interface

    - Evaluate the wave speeds at the interface. this is obtained following the approach proposed by Soares-Frazão and Zech (2011), for which the wave speeds can be calculated as an approximation of the smallest eigenvalue of the system of governing equations, i.e. Shallow water and Exner. They read:

$$\lambda_1 = 1/2(u_n - c - \sqrt{(u_n - c)^2 + 4a_2c^2}) \tag{2.19}$$

$$\lambda_2 = 1/2(u_n - c + \sqrt{(u_n - c)^2 + 4a_2c^2}) \tag{2.20}$$

where $u_n = u \cdot n_{ij,x} + v \cdot n_{ij,y}$, $c = \sqrt{gh}$ and $a_2 = \dfrac{\partial q_{b,n}}{\partial q_n}$ which is the derivative of the bed load discharge in the normal flow direction with respect to the hydraulic flux direction. Then the speeds estimate are

$$S^- = \min(\lambda_{1,L}, \lambda_{1,R}) \tag{2.21}$$

and

$$S^+ = \max(\lambda_{2,L}, \lambda_{2,R}) \tag{2.22}$$

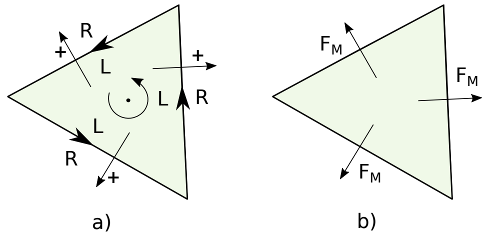


Figure 2.5 *a) Sign convention for the edge direction: counterclockwise b) Positive morphological flux direction at edges: from left (L) to right (R)*

- Flux calculation:

$$q_{Bij}^{HLL} = \begin{cases} q_{Bi,n} & if \quad S^{-} \geq 0, \\ \dfrac{q_{Bi,n}S^{+} - q_{Bj,n}S^{-} + S^{-}S^{+}\left(z_{Bj} - z_{Bi}\right)}{S^{+} - S^{-}} & if \quad S^{-} < 0 < S^{+}, \\ q_{Bj,n} & if \quad S^{+} \leq 0. \end{cases}$$

The convention for the positive bed load flux direction is the same as for the hydrodynamic flux and is presented on Figure 2.5

2.4.1.3 Discretization of External Source Term

The source term Sl_b describes a local input or removal of sediment mass into a river.

An external source is defined as specific mass flux δ (m/s), uniformly distributed over a number of elements of the domain (region) with a specific surface area. The external source can be specified as the total volume flux (m^3/s) for a specific region of the domain. The external source value is divided among the cells composing the region and converted to cell specific mass flux δ_i. The volume allocated is characterized by different behaviors:

$$\begin{array}{llll} \text{Exact:} & S_{b,i} = \delta_i & & \\ \text{Available:} & S_{b,i} = \delta_i & \text{if} & \delta_i \cdot \Delta t > 0 \\ & S_{b,i} = \max(\delta_i, -(z_{Fix} - z_i) & \text{if} & \delta_i \cdot \Delta t < 0 \\ \text{Infinity:} & S_{b,i} = \delta_i & \text{if} & \delta_i \cdot \Delta t > 0 \\ & S_{b,i} = -(z_{Fix} - z_i) & \text{if} & \delta_i \cdot \Delta t < 0 \end{array} \tag{2.23}$$

Where z_i is the bottom elevation and z_{Fix} the fixed bed elevation of the element i. The external source volume is added to the initial bottom elevation of element i.

$$z_i^{t+1} = z_i^t + S_{b,i} \cdot \Delta t \tag{2.24}$$

2.4.2 Solution procedure

The numerical solution procedure of BASEMENT explains how the discretised Exner equation (eq. 1.15) is solved through a sequence of loops over the edges or cells (Figure 2.6).

In the numerical simulation, the hydrodynamic and morphodynamic simulations are performed in a decoupled way. The morphodynamic simulation is executed after the

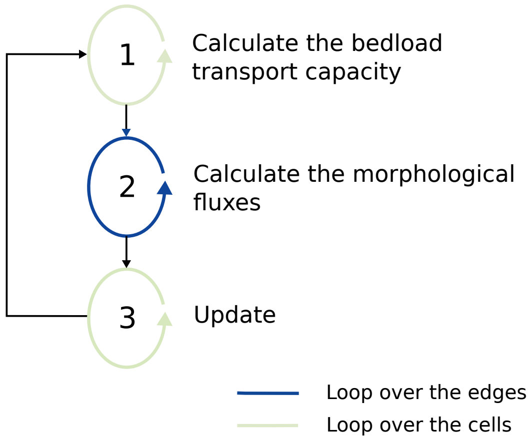


Figure 2.6 *Numerical solution procedure of morphodynamic simulation for each time step* Δt

hydrodynamic simulation, using the hydraulic fluxes to calculate the morphological fluxes. This approach assumes that the sediment transport is much slower than the water velocity, which is an accurate assumption for the numerical modelling of slow flood with morphological changes occurring over a long period (Soares-Frazão and Zech, 2011). The numerical solution procedure of Figure 2.6 is performed after the step 4 of Figure 2.4 inside the same time step Δt.

The numerical solution of the Exner equation starts with a loop over the cells in order to find the bedload transport capacity q_b with a potential correction due to a curvature effect or lateral bed slope. Then, the morphological fluxes F_M are calculated at the element edges and finally, the bed elevation z_b is updated over the cells.

2.5 Numerical solution of Passive Tracers

2.5.1 Numerical solution of the scalar advection equation

2.5.1.1 Fundamentals and spatial discretization

The scalar advection equation assures that scalar masses are conserved in the flow column. The same cell cell-centered finite volume approach used for the SWE is employed in scalar advection, namely the HLLC approximate Riemann solver. In order to discretize the scalar advection equation, the same unstructured mesh adopted for the hydrodynamic and morphodynamic parts is used, as is the same finite volume approach. As a consequence, the scalar concentration ϕ_s is defined at the element center and is equally distributed over the element.

By integrating the scalar advection equation in the control volume $V = [\Omega_i] \times [t^n, t^{n+1}]$, we obtain

$$q_{Si}^{n+1} = q_{Si}^{n} - \frac{\Delta t}{|\Omega_i|} \sum_{j=1}^{3} [f_{Sij} \cdot l_{ij}] + \Delta t \boldsymbol{S}_{s_i} \; . \tag{2.25}$$

where f_{Sij} is the intercell scalar flux. The calculation of the scalar fluxes at the cell interface proceeds as follows:

1. loop over the cell interfaces and compute the flux at the interface using a simplified approximate HLLC Riemann solver at each interface:
   - Retrieve the hydrodynamic fluxes $\boldsymbol{F}_{ij}$ at the interface between cells i and j and extract the fist component of the flux vector (mass fluxes) to the variable q_m.
   - For each specie, perform the flux calculation through a simplified version of the HLLC solver for solute transport:

   $$f_{Sij} = \frac{q_{Si}}{h} q_m \;\; \text{if} \;\; q_m > 0$$
   $$f_{Sij} = \frac{q_{Sj}}{h} q_m \;\; \text{if} \;\; q_m < 0$$

   this approach significantly reduces the numerical effort involved in the computation of the numerical fluxes for each species without sacrificing accuracy.

2. loop over the cells and update the conserved quantities with the fluxes at each of its interfaces:
   - Retrieve the hydrodynamic fluxes f_{Sij} at each of the three interfaces.
   - For each specie, perform the update as prescribed in eq. 2.25.

2.5.1.2 Discretization of External Source Term

The source term S_s describes a local input or removal of scalar mass into a river. An external source is defined as specific mass flux δ (m/s), uniformly distributed over a number of elements of the domain (region) with a specific surface area. The external source can be specified as the total volume flux (m^3/s) for a specific region of the domain. The external source value is divided among the cells composing the region and converted to cell specific mass flux δ_i. The volume allocated is characterized by different behaviors:

$$\begin{array}{lll} \text{Exact:} & S_{S,i} = \delta_i & \\ \text{Available:} & S_{S,i} = \delta_i & \text{if} \quad \delta_S \cdot \Delta t > 0 \\ & S_{S,i} = \max(\delta_i, -q_{Si}) & \text{if} \quad \delta_i \cdot \Delta t < 0 \\ \text{Infinity:} & S_{S,i} = \delta_i & \text{if} \quad \delta_i \cdot \Delta t > 0 \\ & S_{S,i} = -q_{Si} & \text{if} \quad \delta_i \cdot \Delta t < 0 \end{array}$$

The external source volume is then added to the initial bottom elevation of element i through a first-order Euler approach

$$q_{Si}^{t+1} = q_{Si}^{t} + S_{S,i} \cdot \Delta t$$

3

References

Bezzola, G.R. (2002). Fliesswiederstand und Sohlenstabilität natürlicher Gerinne [PhD thesis]: Eidgenössische Technische Hochschule Zürich.

Chen, X., Ma, J. and Dey, S. (2010). Sediment transport on arbitrary slopes: Simplified model. *Journal of Hydraulic Engineering-ASCE*, 136(5): 311–317.

Duran, A., Liang, Q. and Marche, F. (2013). On the well-balanced numerical discretization of shallow water equations on unstructured meshes. *Journal of Computational Physics*, 235: 565–586.

Engelund, F. (1974). Flow and bed topography in channel bends. *Journal of the Hydraulics Division ASCE*, 100(11): 1631–1648.

Engelund, F. and Hansen, E. (1972). A monograph on sediment transport in alluvial streams. *Teknisk Forlag, Copenhagen*,

Exner, F.M. (1925). Ueber die wechselwirkung zwischen wasser und geschiebe in fluessen. Akademie der Wissenschaften, Mathematische Naturwissenschaft Abt. IIa, Wien, Austria.

Grass, A.J. (1981). Sediment transport by waves and currents. *University College, London, Dept. of Civil Engineering*, London, UK.

Ikeda, S. (1982). Lateral Bed-Load Transport on Side Slopes. *Journal of the Hydraulics Division-Asce*, 108(11): 1369–1373.

Malcherek, A. (2001). Sedimenttransport und Morphodynamik. Vorlesungsskript der Universität der Bundeswehr München, München.

Meyer-Peter, E. and Müller, R. (1948). Formulas for Bed-Load Transport, 2nd Meeting IAHR, Stockholm, Sweden.

Rozovskii, I.L. (1961). Flow of Water in Bends of Open Channels. *Academy of Science of the Ukrainian S.S.R, Institute of Hydrology; Hydraulic Engineering*,

Shields, A. (1936). Anwendungen der Ähnlichkeitsmechanik und der Turbulenzforschung auf die Geschiebebewegungen. Mitteilung der Preussischen Versuchsanstalt für Wasserbau und Schiffbau. Berlin, Deutschland.

Smart, G. and Jaeggi, M. (1983). Sediment Transport on Steep Slopes. *VAW-Mitteilung* 64, Versuchsanstalt für Wasserbau,Hydrologie und Glaziologie (VAW). Zürich, ETH Zürich.

Soares-Frazão, S. and Zech, Y. (2011). HLLC scheme with novel wave-speed estimators appropriate for two-dimensional shallow-water flow on erodible bed. *International Journal for Numerical Methods in Fluids*, 66.

Talmon, A., Struiksma, N. and van Mierlo, M. (1995). Laboratory measurements of the direction of sediment transport on transverse alluvial-bed slopes. *Journal of Hydraulic Research*, 33(4): 495–517.

Toro, S., E. F. (1994). Restoration of the contact surface in the HLL-Riemann solver. *Shock Waves*, 4: 25–34.

Toro, E.F. (2009). Riemann Solvers and Numerical Methods for Fluid Dynamics. *Springer-Verlag*, Berlin.

Toro, E.F. (2001). Shock-Capturing Methods for Free-Surface Shallow Flows. *John Wiley*, Chichester, New York.

van Rijn, L.C. (1989). Handbook Sediment Transport by Current and Waves. *Delft Hydraulics Laboratory*, Delft, The Netherlands.

van Rijn, L.C. (1984). Sediment Transport, Part I: Bed Load Transport. *Journal of Hydraulic Engineering, ASCE*, 110(10): 1431–1456.

Vonwiller, L. (2017). Numerical Modeling of Morphological Response of Gravel-Bed Rivers to Sediment Supply. *VAW-Mitteilung* 246, Versuchsanstalt für Wasserbau, Hydrologie und Glaziologie (VAW). Zürich, ETH Zürich.

Yalin, M. and Silva, A. da (2001). Fluvial processes. *International Association of Hydraulic Engineering; Research (IAHR), Delft, The Netherlands*,

BASIC SIMULATION ENVIRONMENT
FOR SIMULATION OF ENVIRONMENTAL FLOW
AND NATURAL HAZARD SIMULATION

APPENDIX

VERSION 3.1
November 2020

1

Third Party Software

1.1 Third party software copyright notices

Abseil

```
Copyright Google Inc.

License: Apache 2.0
```

Bzip2

```
This program, "bzip2", the associated library "libbzip2", and all
documentation, are copyright (C) 1996-2010 Julian R Seward.  All
rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

2. The origin of this software must not be misrepresented; you must
   not claim that you wrote the original software.  If you use this
   software in a product, an acknowledgment in the product
   documentation would be appreciated but is not required.

3. Altered source versions must be plainly marked as such, and must
   not be misrepresented as being the original software.

4. The name of the author may not be used to endorse or promote
   products derived from this software without specific prior written
   permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS
OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED.  IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE
GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
```

Double-conversion

Egl-registry

Freetype

```
The contributed BDF and PCF drivers  come with a license similar to that
of the X Window System.  It is compatible to the above two licenses (see
file src/bdf/README and  src/pcf/README).  The same holds  for the files
`fthash.c' and  `fthash.h'; their  code was  part of  the BDF  driver in
earlier FreeType versions.

The gzip module uses the zlib license (see src/gzip/zlib.h) which too is
compatible to the above two licenses.

The MD5 checksum support (only used for debugging in development builds)
is in the public domain.
```

Glibc-queue

Google Test

Harfbuzz

```
HarfBuzz is licensed under the so-called "Old MIT" license.  Details follow.
For parts of HarfBuzz that are licensed under different licenses see individual
files names COPYING in subdirectories where applicable.

Copyright © 2010,2011,2012,2013,2014,2015,2016,2017,2018,2019  Google, Inc.
Copyright © 2019  Facebook, Inc.
Copyright © 2012  Mozilla Foundation
Copyright © 2011  Codethink Limited
Copyright © 2008,2010  Nokia Corporation and/or its subsidiary(-ies)
Copyright © 2009  Keith Stribley
Copyright © 2009  Martin Hosken and SIL International
Copyright © 2007  Chris Wilson
Copyright © 2006  Behdad Esfahbod
Copyright © 2005  David Turner
Copyright © 2004,2007,2008,2009,2010  Red Hat, Inc.
Copyright © 1998-2004  David Turner and Werner Lemberg

For full copyright notices consult the individual files in the package.


Permission is hereby granted, without written agreement and without
license or royalty fees, to use, copy, modify, and distribute this
software and its documentation for any purpose, provided that the
above copyright notice and the following two paragraphs appear in
all copies of this software.

IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE TO ANY PARTY FOR
DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES
ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN
IF THE COPYRIGHT HOLDER HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.

THE COPYRIGHT HOLDER SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING,
BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE.  THE SOFTWARE PROVIDED HEREUNDER IS
ON AN "AS IS" BASIS, AND THE COPYRIGHT HOLDER HAS NO OBLIGATION TO
PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.
```

HDF5

```
HDF5 (Hierarchical Data Format 5) Software Library and Utilities
Copyright (c) 2006-2018, The HDF Group.

NCSA HDF5 (Hierarchical Data Format 5) Software Library and Utilities
Copyright (c) 1998-2006, The Board of Trustees of the University of Illinois.

All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted for any purpose (including commercial purposes)
provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice,
   this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice,
   this list of conditions, and the following disclaimer in the documentation
   and/or materials provided with the distribution.

3. Neither the name of The HDF Group, the name of the University, nor the
   name of any Contributor may be used to endorse or promote products derived
   from this software without specific prior written permission from
   The HDF Group, the University, or the Contributor, respectively.

DISCLAIMER:
THIS SOFTWARE IS PROVIDED BY THE HDF GROUP AND THE CONTRIBUTORS
```

HighFive

JSON for Modern C++

Libjpeg-turbo

```
libjpeg-turbo Licenses
======================

libjpeg-turbo is covered by three compatible BSD-style open source licenses:

- The IJG (Independent JPEG Group) License, which is listed in
  [README.ijg](README.ijg)

  This license applies to the libjpeg API library and associated programs
  (any code inherited from libjpeg, and any modifications to that code.)

- The Modified (3-clause) BSD License, which is listed below

  This license covers the TurboJPEG API library and associated programs, as
  well as the build system.

- The [zlib License](https://opensource.org/licenses/Zlib)

  This license is a subset of the other two, and it covers the libjpeg-turbo
  SIMD extensions.


Complying with the libjpeg-turbo Licenses
=========================================

This section provides a roll-up of the libjpeg-turbo licensing terms, to the
best of our understanding.

1.  If you are distributing a modified version of the libjpeg-turbo source,
    then:

    1.  You cannot alter or remove any existing copyright or license notices
        from the source.

        **Origin**
        - Clause 1 of the IJG License
        - Clause 1 of the Modified BSD License
        - Clauses 1 and 3 of the zlib License

    2.  You must add your own copyright notice to the header of each source
        file you modified, so others can tell that you modified that file (if
        there is not an existing copyright header in that file, then you can
        simply add a notice stating that you modified the file.)

        **Origin**
        - Clause 1 of the IJG License
        - Clause 2 of the zlib License

    3.  You must include the IJG README file, and you must not alter any of the
        copyright or license text in that file.

        **Origin**
        - Clause 1 of the IJG License

2.  If you are distributing only libjpeg-turbo binaries without the source, or
    if you are distributing an application that statically links with
    libjpeg-turbo, then:

    1.  Your product documentation must include a message stating:

        This software is based in part on the work of the Independent JPEG
        Group.

        **Origin**
        - Clause 2 of the IJG license

    2.  If your binary distribution includes or uses the TurboJPEG API, then
        your product documentation must include the text of the Modified BSD
```

```
        License (see below.)

        **Origin**
        - Clause 2 of the Modified BSD License

3.  You cannot use the name of the IJG or The libjpeg-turbo Project or the
    contributors thereof in advertising, publicity, etc.

    **Origin**
    - IJG License
    - Clause 3 of the Modified BSD License

4.  The IJG and The libjpeg-turbo Project do not warrant libjpeg-turbo to be
    free of defects, nor do we accept any liability for undesirable
    consequences resulting from your use of the software.

    **Origin**
    - IJG License
    - Modified BSD License
    - zlib License


The Modified (3-clause) BSD License
===================================

Copyright (C)2009-2019 D. R. Commander.  All Rights Reserved.
Copyright (C)2015 Viktor Szathmáry.  All Rights Reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.
- Neither the name of the libjpeg-turbo Project nor the names of its
  contributors may be used to endorse or promote products derived from this
  software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS",
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED.  IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.


Why Three Licenses?
===================

The zlib License could have been used instead of the Modified (3-clause) BSD
License, and since the IJG License effectively subsumes the distribution
conditions of the zlib License, this would have effectively placed
libjpeg-turbo binary distributions under the IJG License.  However, the IJG
License specifically refers to the Independent JPEG Group and does not extend
attribution and endorsement protections to other entities.  Thus, it was
desirable to choose a license that granted us the same protections for new code
that were granted to the IJG for code derived from their software.
```

Libpng

```
COPYRIGHT NOTICE, DISCLAIMER, and LICENSE
=========================================
```

```
libpng-0.96, and are distributed according to the same disclaimer and
license as libpng-0.96, with the following individuals added to the
list of Contributing Authors:

    Tom Lane
    Glenn Randers-Pehrson
    Willem van Schaik

libpng versions 0.89, June 1996, through 0.96, May 1997, are
Copyright (c) 1996-1997 Andreas Dilger, are derived from libpng-0.88,
and are distributed according to the same disclaimer and license as
libpng-0.88, with the following individuals added to the list of
Contributing Authors:

    John Bowler
    Kevin Bracey
    Sam Bushell
    Magnus Holmgren
    Greg Roelofs
    Tom Tanner

Some files in the "scripts" directory have other copyright owners,
but are released under this license.

libpng versions 0.5, May 1995, through 0.88, January 1996, are
Copyright (c) 1995-1996 Guy Eric Schalnat, Group 42, Inc.

For the purposes of this copyright and license, "Contributing Authors"
is defined as the following set of individuals:

    Andreas Dilger
    Dave Martindale
    Guy Eric Schalnat
    Paul Schmidt
    Tim Wegner

The PNG Reference Library is supplied "AS IS".  The Contributing
Authors and Group 42, Inc. disclaim all warranties, expressed or
implied, including, without limitation, the warranties of
merchantability and of fitness for any purpose.  The Contributing
Authors and Group 42, Inc. assume no liability for direct, indirect,
incidental, special, exemplary, or consequential damages, which may
result from the use of the PNG Reference Library, even if advised of
the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this
source code, or portions hereof, for any purpose, without fee, subject
to the following restrictions:

 1. The origin of this source code must not be misrepresented.

 2. Altered versions must be plainly marked as such and must not
    be misrepresented as being the original source.

 3. This Copyright notice may not be removed or altered from any
    source or altered source distribution.

The Contributing Authors and Group 42, Inc. specifically permit,
without fee, and encourage the use of this source code as a component
to supporting the PNG file format in commercial products.  If you use
this source code in a product, acknowledgment is not required but would
be appreciated.
```

OP2

```
Copyright (c) 2011, Gihan Mudalige, Istvan Reguly, Mike Giles, and others.
Please see the AUTHORS file in  the main source directory for details.
All rights reserved.

License: BSD 2.0
```

OpenMesh

PCRE2

```
PCRE2 LICENCE
-------------

PCRE2 is a library of functions to support regular expressions whose syntax
and semantics are as close as possible to those of the Perl 5 language.

Release 10 of PCRE2 is distributed under the terms of the "BSD" licence, as
specified below, with one exemption for certain binary redistributions. The
documentation for PCRE2, supplied in the "doc" directory, is distributed under
the same terms as the software itself. The data in the testdata directory is
not copyrighted and is in the public domain.

The basic library functions are written in C and are freestanding. Also
included in the distribution is a just-in-time compiler that can be used to
optimize pattern matching. This is an optional feature that can be omitted when
the library is built.


THE BASIC LIBRARY FUNCTIONS
---------------------------

Written by:       Philip Hazel
Email local part: ph10
Email domain:     cam.ac.uk

University of Cambridge Computing Service,
Cambridge, England.

Copyright (c) 1997-2017 University of Cambridge
All rights reserved.


PCRE2 JUST-IN-TIME COMPILATION SUPPORT
--------------------------------------

Written by:       Zoltan Herczeg
Email local part: hzmester
Emain domain:     freemail.hu

Copyright(c) 2010-2017 Zoltan Herczeg
All rights reserved.


STACK-LESS JUST-IN-TIME COMPILER
--------------------------------

Written by:       Zoltan Herczeg
Email local part: hzmester
Emain domain:     freemail.hu

Copyright(c) 2009-2017 Zoltan Herczeg
All rights reserved.
```

```
THE "BSD" LICENCE
-----------------

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

    * Redistributions of source code must retain the above copyright notices,
      this list of conditions and the following disclaimer.

    * Redistributions in binary form must reproduce the above copyright
      notices, this list of conditions and the following disclaimer in the
      documentation and/or other materials provided with the distribution.

    * Neither the name of the University of Cambridge nor the names of any
      contributors may be used to endorse or promote products derived from this
      software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.


EXEMPTION FOR BINARY LIBRARY-LIKE PACKAGES
------------------------------------------

The second condition in the BSD licence (covering binary redistributions) does
not apply all the way down a chain of software. If binary package A includes
PCRE2, it must respect the condition, but if package B is software that
includes package A, the condition is not imposed on package B unless it uses
PCRE2 independently.
```

Qt

```
Copyright (C) 2018 The Qt Company Ltd and other contributors

License: LGPL 3.0
```

SimpleXdmf

```
Copyright (c) 2017 hsimyu

License: MIT
```

TCLAP

```
Copyright (c) 2003 Michael E. Smoot
Copyright (c) 2004 Daniel Aarno
Copyright (c) 2017 Google Inc.

License: MIT
```

Vcpkg

```
Copyright (c) Microsoft Corporation

All rights reserved.

License: MIT
```

Zlib

1.2 Third party software licenses

LGPL 3.0

```
Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License
without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a
facility refers to a function or data to be supplied by an Application
that uses the facility (other than as an argument passed when the
facility is invoked), then you may convey a copy of the modified
version:

 a) under this License, provided that you make a good faith effort to
 ensure that, in the event an Application does not supply the
 function or data, the facility still operates, and performs
 whatever part of its purpose remains meaningful, or

 b) under the GNU GPL, with none of the additional permissions of
 this License applicable to that copy.

3. Object Code Incorporating Material from Library Header Files.

The object code form of an Application may incorporate material from
a header file that is part of the Library.  You may convey such object
code under terms of your choice, provided that, if the incorporated
material is not limited to numerical parameters, data structure
layouts and accessors, or small macros, inline functions and templates
(ten or fewer lines in length), you do both of the following:

 a) Give prominent notice with each copy of the object code that the
 Library is used in it and that the Library and its use are
 covered by this License.

 b) Accompany the object code with a copy of the GNU GPL and this license
 document.

4. Combined Works.

You may convey a Combined Work under terms of your choice that,
taken together, effectively do not restrict modification of the
portions of the Library contained in the Combined Work and reverse
engineering for debugging such modifications, if you also do each of
the following:

 a) Give prominent notice with each copy of the Combined Work that
 the Library is used in it and that the Library and its use are
 covered by this License.

 b) Accompany the Combined Work with a copy of the GNU GPL and this license
 document.

 c) For a Combined Work that displays copyright notices during
 execution, include the copyright notice for the Library among
 these notices, as well as a reference directing the user to the
 copies of the GNU GPL and this license document.

 d) Do one of the following:

     0) Convey the Minimal Corresponding Source under the terms of this
     License, and the Corresponding Application Code in a form
     suitable for, and under terms that permit, the user to
     recombine or relink the Application with a modified version of
     the Linked Version to produce a modified Combined Work, in the
     manner specified by section 6 of the GNU GPL for conveying
     Corresponding Source.

     1) Use a suitable shared library mechanism for linking with the
```

Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

5. Combined Libraries.

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

GPL 3.0

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <https://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this

```
License.  Each licensee is addressed as "you".  "Licensees" and
"recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work
in a fashion requiring copyright permission, other than the making of an
exact copy.  The resulting work is called a "modified version" of the
earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based
on the Program.

To "propagate" a work means to do anything with it that, without
permission, would make you directly or secondarily liable for
infringement under applicable copyright law, except executing it on a
computer or modifying a private copy.  Propagation includes copying,
distribution (with or without modification), making available to the
public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other
parties to make or receive copies.  Mere interaction with a user through
a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices"
to the extent that it includes a convenient and prominently visible
feature that (1) displays an appropriate copyright notice, and (2)
tells the user that there is no warranty for the work (except to the
extent that warranties are provided), that licensees may convey the
work under this License, and how to view a copy of this License.  If
the interface presents a list of user commands or options, such as a
menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work
for making modifications to it.  "Object code" means any non-source
form of a work.

A "Standard Interface" means an interface that either is an official
standard defined by a recognized standards body, or, in the case of
interfaces specified for a particular programming language, one that
is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other
than the work as a whole, that (a) is included in the normal form of
packaging a Major Component, but which is not part of that Major
Component, and (b) serves only to enable use of the work with that
Major Component, or to implement a Standard Interface for which an
implementation is available to the public in source code form.  A
"Major Component", in this context, means a major essential component
(kernel, window system, and so on) of the specific operating system
(if any) on which the executable work runs, or a compiler used to
produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all
the source code needed to generate, install, and (for an executable
work) run the object code and to modify the work, including scripts to
control those activities.  However, it does not include the work's
System Libraries, or general-purpose tools or generally available free
programs which are used unmodified in performing those activities but
which are not part of the work.  For example, Corresponding Source
includes interface definition files associated with source files for
the work, and the source code for shared libraries and dynamically
linked subprograms that the work is specifically designed to require,
such as by intimate data communication or control flow between those
subprograms and other parts of the work.

The Corresponding Source need not include anything that users
can regenerate automatically from other parts of the Corresponding
Source.
```

```
The Corresponding Source for a work in source code form is that
same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of
copyright on the Program, and are irrevocable provided the stated
conditions are met.  This License explicitly affirms your unlimited
permission to run the unmodified Program.  The output from running a
covered work is covered by this License only if the output, given its
content, constitutes a covered work.  This License acknowledges your
rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not
convey, without conditions so long as your license otherwise remains
in force.  You may convey covered works to others for the sole purpose
of having them make modifications exclusively for you, or provide you
with facilities for running those works, provided that you comply with
the terms of this License in conveying all material for which you do
not control copyright.  Those thus making or running the covered works
for you must do so exclusively on your behalf, under your direction
and control, on terms that prohibit them from making any copies of
your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under
the conditions stated below.  Sublicensing is not allowed; section 10
makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological
measure under any applicable law fulfilling obligations under article
11 of the WIPO copyright treaty adopted on 20 December 1996, or
similar laws prohibiting or restricting circumvention of such
measures.

When you convey a covered work, you waive any legal power to forbid
circumvention of technological measures to the extent such circumvention
is effected by exercising rights under this License with respect to
the covered work, and you disclaim any intention to limit operation or
modification of the work as a means of enforcing, against the work's
users, your or third parties' legal rights to forbid circumvention of
technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you
receive it, in any medium, provided that you conspicuously and
appropriately publish on each copy an appropriate copyright notice;
keep intact all notices stating that this License and any
non-permissive terms added in accord with section 7 apply to the code;
keep intact all notices of the absence of any warranty; and give all
recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey,
and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to
produce it from the Program, in the form of source code under the
terms of section 4, provided that you also meet all of these conditions:

  a) The work must carry prominent notices stating that you modified
  it, and giving a relevant date.

  b) The work must carry prominent notices stating that it is
  released under this License and any conditions added under section
  7.  This requirement modifies the requirement in section 4 to
  "keep intact all notices".
```

```
  c) You must license the entire work, as a whole, under this
  License to anyone who comes into possession of a copy.  This
  License will therefore apply, along with any applicable section 7
  additional terms, to the whole of the work, and all its parts,
  regardless of how they are packaged.  This License gives no
  permission to license the work in any other way, but it does not
  invalidate such permission if you have separately received it.

  d) If the work has interactive user interfaces, each must display
  Appropriate Legal Notices; however, if the Program has interactive
  interfaces that do not display Appropriate Legal Notices, your
  work need not make them do so.

A compilation of a covered work with other separate and independent
works, which are not by their nature extensions of the covered work,
and which are not combined with it such as to form a larger program,
in or on a volume of a storage or distribution medium, is called an
"aggregate" if the compilation and its resulting copyright are not
used to limit the access or legal rights of the compilation's users
beyond what the individual works permit.  Inclusion of a covered work
in an aggregate does not cause this License to apply to the other
parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms
of sections 4 and 5, provided that you also convey the
machine-readable Corresponding Source under the terms of this License,
in one of these ways:

  a) Convey the object code in, or embodied in, a physical product
  (including a physical distribution medium), accompanied by the
  Corresponding Source fixed on a durable physical medium
  customarily used for software interchange.

  b) Convey the object code in, or embodied in, a physical product
  (including a physical distribution medium), accompanied by a
  written offer, valid for at least three years and valid for as
  long as you offer spare parts or customer support for that product
  model, to give anyone who possesses the object code either (1) a
  copy of the Corresponding Source for all the software in the
  product that is covered by this License, on a durable physical
  medium customarily used for software interchange, for a price no
  more than your reasonable cost of physically performing this
  conveying of source, or (2) access to copy the
  Corresponding Source from a network server at no charge.

  c) Convey individual copies of the object code with a copy of the
  written offer to provide the Corresponding Source.  This
  alternative is allowed only occasionally and noncommercially, and
  only if you received the object code with such an offer, in accord
  with subsection 6b.

  d) Convey the object code by offering access from a designated
  place (gratis or for a charge), and offer equivalent access to the
  Corresponding Source in the same way through the same place at no
  further charge.  You need not require recipients to copy the
  Corresponding Source along with the object code.  If the place to
  copy the object code is a network server, the Corresponding Source
  may be on a different server (operated by you or a third party)
  that supports equivalent copying facilities, provided you maintain
  clear directions next to the object code saying where to find the
  Corresponding Source.  Regardless of what server hosts the
  Corresponding Source, you remain obligated to ensure that it is
  available for as long as needed to satisfy these requirements.

  e) Convey the object code using peer-to-peer transmission, provided
  you inform other peers where the object code and Corresponding
  Source of the work are being offered to the general public at no
```

```
  charge under subsection 6d.

A separable portion of the object code, whose source code is excluded
from the Corresponding Source as a System Library, need not be
included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any
tangible personal property which is normally used for personal, family,
or household purposes, or (2) anything designed or sold for incorporation
into a dwelling.  In determining whether a product is a consumer product,
doubtful cases shall be resolved in favor of coverage.  For a particular
product received by a particular user, "normally used" refers to a
typical or common use of that class of product, regardless of the status
of the particular user or of the way in which the particular user
actually uses, or expects or is expected to use, the product.  A product
is a consumer product regardless of whether the product has substantial
commercial, industrial or non-consumer uses, unless such uses represent
the only significant mode of use of the product.

"Installation Information" for a User Product means any methods,
procedures, authorization keys, or other information required to install
and execute modified versions of a covered work in that User Product from
a modified version of its Corresponding Source.  The information must
suffice to ensure that the continued functioning of the modified object
code is in no case prevented or interfered with solely because
modification has been made.

If you convey an object code work under this section in, or with, or
specifically for use in, a User Product, and the conveying occurs as
part of a transaction in which the right of possession and use of the
User Product is transferred to the recipient in perpetuity or for a
fixed term (regardless of how the transaction is characterized), the
Corresponding Source conveyed under this section must be accompanied
by the Installation Information.  But this requirement does not apply
if neither you nor any third party retains the ability to install
modified object code on the User Product (for example, the work has
been installed in ROM).

The requirement to provide Installation Information does not include a
requirement to continue to provide support service, warranty, or updates
for a work that has been modified or installed by the recipient, or for
the User Product in which it has been modified or installed.  Access to a
network may be denied when the modification itself materially and
adversely affects the operation of the network or violates the rules and
protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided,
in accord with this section must be in a format that is publicly
documented (and with an implementation available to the public in
source code form), and must require no special password or key for
unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this
License by making exceptions from one or more of its conditions.
Additional permissions that are applicable to the entire Program shall
be treated as though they were included in this License, to the extent
that they are valid under applicable law.  If additional permissions
apply only to part of the Program, that part may be used separately
under those permissions, but the entire Program remains governed by
this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option
remove any additional permissions from that copy, or from any part of
it.  (Additional permissions may be written to require their own
removal in certain cases when you modify the work.)  You may place
additional permissions on material, added by you to a covered work,
for which you have or can give appropriate copyright permission.
```

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under

```
this License.  If your rights have been terminated and not permanently
reinstated, you do not qualify to receive new licenses for the same
material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or
run a copy of the Program.  Ancillary propagation of a covered work
occurring solely as a consequence of using peer-to-peer transmission
to receive a copy likewise does not require acceptance.  However,
nothing other than this License grants you permission to propagate or
modify any covered work.  These actions infringe copyright if you do
not accept this License.  Therefore, by modifying or propagating a
covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically
receives a license from the original licensors, to run, modify and
propagate that work, subject to this License.  You are not responsible
for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an
organization, or substantially all assets of one, or subdividing an
organization, or merging organizations.  If propagation of a covered
work results from an entity transaction, each party to that
transaction who receives a copy of the work also receives whatever
licenses to the work the party's predecessor in interest had or could
give under the previous paragraph, plus a right to possession of the
Corresponding Source of the work from the predecessor in interest, if
the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the
rights granted or affirmed under this License.  For example, you may
not impose a license fee, royalty, or other charge for exercise of
rights granted under this License, and you may not initiate litigation
(including a cross-claim or counterclaim in a lawsuit) alleging that
any patent claim is infringed by making, using, selling, offering for
sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this
License of the Program or a work on which the Program is based.  The
work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims
owned or controlled by the contributor, whether already acquired or
hereafter acquired, that would be infringed by some manner, permitted
by this License, of making, using, or selling its contributor version,
but do not include claims that would be infringed only as a
consequence of further modification of the contributor version.  For
purposes of this definition, "control" includes the right to grant
patent sublicenses in a manner consistent with the requirements of
this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free
patent license under the contributor's essential patent claims, to
make, use, sell, offer for sale, import and otherwise run, modify and
propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express
agreement or commitment, however denominated, not to enforce a patent
(such as an express permission to practice a patent or covenant not to
sue for patent infringement).  To "grant" such a patent license to a
party means to make such an agreement or commitment not to enforce a
patent against the party.

If you convey a covered work, knowingly relying on a patent license,
and the Corresponding Source of the work is not available for anyone
```

```
to copy, free of charge and under the terms of this License, through a
publicly available network server or other readily accessible means,
then you must either (1) cause the Corresponding Source to be so
available, or (2) arrange to deprive yourself of the benefit of the
patent license for this particular work, or (3) arrange, in a manner
consistent with the requirements of this License, to extend the patent
license to downstream recipients.  "Knowingly relying" means you have
actual knowledge that, but for the patent license, your conveying the
covered work in a country, or your recipient's use of the covered work
in a country, would infringe one or more identifiable patents in that
country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or
arrangement, you convey, or propagate by procuring conveyance of, a
covered work, and grant a patent license to some of the parties
receiving the covered work authorizing them to use, propagate, modify
or convey a specific copy of the covered work, then the patent license
you grant is automatically extended to all recipients of the covered
work and works based on it.

A patent license is "discriminatory" if it does not include within
the scope of its coverage, prohibits the exercise of, or is
conditioned on the non-exercise of one or more of the rights that are
specifically granted under this License.  You may not convey a covered
work if you are a party to an arrangement with a third party that is
in the business of distributing software, under which you make payment
to the third party based on the extent of your activity of conveying
the work, and under which the third party grants, to any of the
parties who would receive the covered work from you, a discriminatory
patent license (a) in connection with copies of the covered work
conveyed by you (or copies made from those copies), or (b) primarily
for and in connection with specific products or compilations that
contain the covered work, unless you entered into that arrangement,
or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting
any implied license or other defenses to infringement that may
otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License.  If you cannot convey a
covered work so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you may
not convey it at all.  For example, if you agree to terms that obligate you
to collect a royalty for further conveying from those to whom you convey
the Program, the only way you could satisfy both those terms and this
License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have
permission to link or combine any covered work with a work licensed
under version 3 of the GNU Affero General Public License into a single
combined work, and to convey the resulting work.  The terms of this
License will continue to apply to the part which is the covered work,
but the special requirements of the GNU Affero General Public License,
section 13, concerning interaction through a network will apply to the
combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of
the GNU General Public License from time to time.  Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number.  If the
```

```
  This program is distributed in the hope that it will be useful,
  but WITHOUT ANY WARRANTY; without even the implied warranty of
  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
  GNU General Public License for more details.

  You should have received a copy of the GNU General Public License
  along with this program.  If not, see <https://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short
notice like this when it starts in an interactive mode:

  <program>  Copyright (C) <year>  <name of author>
  This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
  This is free software, and you are welcome to redistribute it
  under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License.  Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
<https://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program
into proprietary programs.  If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library.  If this is what you want to do, use the GNU Lesser General
Public License instead of this License.  But first, please read
<https://www.gnu.org/licenses/why-not-lgpl.html>.
```

GPL 2.0

```
                    GNU GENERAL PUBLIC LICENSE
                       Version 2, June 1991

 Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
 Everyone is permitted to copy and distribute verbatim copies
 of this license document, but changing it is not allowed.

                            Preamble

  The licenses for most software are designed to take away your
freedom to share and change it.  By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users.  This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it.  (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.)  You can apply it to
your programs, too.

  When we speak of free software, we are referring to freedom, not
price.  Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

  To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.
```

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

```
    c) If the modified program normally reads commands interactively
    when run, you must cause it, when started running for such
    interactive use in the most ordinary way, to print or display an
    announcement including an appropriate copyright notice and a
    notice that there is no warranty (or else, saying that you provide
    a warranty) and that users may redistribute the program under
    these conditions, and telling the user how to view a copy of this
    License.  (Exception: if the Program itself is interactive but
    does not normally print such an announcement, your work based on
    the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole.  If
identifiable sections of that work are not derived from the Program,
and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works.  But when you
distribute the same sections as part of a whole which is a work based
on the Program, the distribution of the whole must be on the terms of
this License, whose permissions for other licensees extend to the
entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Program.

In addition, mere aggregation of another work not based on the Program
with the Program (or with a work based on the Program) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

  3. You may copy and distribute the Program (or a work based on it,
under Section 2) in object code or executable form under the terms of
Sections 1 and 2 above provided that you also do one of the following:

    a) Accompany it with the complete corresponding machine-readable
    source code, which must be distributed under the terms of Sections
    1 and 2 above on a medium customarily used for software interchange; or,

    b) Accompany it with a written offer, valid for at least three
    years, to give any third party, for a charge no more than your
    cost of physically performing source distribution, a complete
    machine-readable copy of the corresponding source code, to be
    distributed under the terms of Sections 1 and 2 above on a medium
    customarily used for software interchange; or,

    c) Accompany it with the information you received as to the offer
    to distribute corresponding source code.  (This alternative is
    allowed only for noncommercial distribution and only if you
    received the program in object code or executable form with such
    an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it.  For an executable work, complete source
code means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to
control compilation and installation of the executable.  However, as a
special exception, the source code distributed need not include
anything that is normally distributed (in either source or binary
form) with the major components (compiler, kernel, and so on) of the
operating system on which the executable runs, unless that component
itself accompanies the executable.

If distribution of executable or object code is made by offering
access to copy from a designated place, then offering equivalent
access to copy the source code from the same place counts as
distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

  4. You may not copy, modify, sublicense, or distribute the Program
```

```
except as expressly provided under this License.  Any attempt
otherwise to copy, modify, sublicense or distribute the Program is
void, and will automatically terminate your rights under this License.
However, parties who have received copies, or rights, from you under
this License will not have their licenses terminated so long as such
parties remain in full compliance.

  5. You are not required to accept this License, since you have not
signed it.  However, nothing else grants you permission to modify or
distribute the Program or its derivative works.  These actions are
prohibited by law if you do not accept this License.  Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

  6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions.  You may not impose any further
restrictions on the recipients' exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties to
this License.

  7. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License.  If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Program at all.  For example, if a patent
license would not permit royalty-free redistribution of the Program by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is
implemented by public license practices.  Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

  8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License
may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among
countries not thus excluded.  In such case, this License incorporates
the limitation as if written in the body of this License.

  9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time.  Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number.  If the Program
```

specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year>  <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.
```

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this

```
when it starts in an interactive mode:

    Gnomovision version 69, Copyright (C) year name of author
    Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
    This is free software, and you are welcome to redistribute it
    under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License.  Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary.  Here is a sample; alter the names:

  Yoyodyne, Inc., hereby disclaims all copyright interest in the program
  `Gnomovision' (which makes passes at compilers) written by James Hacker.

  <signature of Ty Coon>, 1 April 1989
  Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs.  If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library.  If this is what you want to do, use the GNU Lesser General
Public License instead of this License.
```

Apache 2.0

```
Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

  "License" shall mean the terms and conditions for use, reproduction,
  and distribution as defined by Sections 1 through 9 of this document.

  "Licensor" shall mean the copyright owner or entity authorized by
  the copyright owner that is granting the License.

  "Legal Entity" shall mean the union of the acting entity and all
  other entities that control, are controlled by, or are under common
  control with that entity. For the purposes of this definition,
  "control" means (i) the power, direct or indirect, to cause the
  direction or management of such entity, whether by contract or
  otherwise, or (ii) ownership of fifty percent (50%) or more of the
  outstanding shares, or (iii) beneficial ownership of such entity.

  "You" (or "Your") shall mean an individual or Legal Entity
  exercising permissions granted by this License.

  "Source" form shall mean the preferred form for making modifications,
  including but not limited to software source code, documentation
  source, and configuration files.

  "Object" form shall mean any form resulting from mechanical
  transformation or translation of a Source form, including but
  not limited to compiled object code, generated documentation,
  and conversions to other media types.

  "Work" shall mean the work of authorship, whether in Source or
  Object form, made available under the License, as indicated by a
  copyright notice that is included in or attached to the work
  (an example is provided in the Appendix below).
```

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not

```
      pertain to any part of the Derivative Works, in at least one
      of the following places: within a NOTICE text file distributed
      as part of the Derivative Works; within the Source form or
      documentation, if provided along with the Derivative Works; or,
      within a display generated by the Derivative Works, if and
      wherever such third-party notices normally appear. The contents
      of the NOTICE file are for informational purposes only and
      do not modify the License. You may add Your own attribution
      notices within Derivative Works that You distribute, alongside
      or as an addendum to the NOTICE text from the Work, provided
      that such additional attribution notices cannot be construed
      as modifying the License.

  You may add Your own copyright statement to Your modifications and
  may provide additional or different license terms and conditions
  for use, reproduction, or distribution of Your modifications, or
  for any such Derivative Works as a whole, provided Your use,
  reproduction, and distribution of the Work otherwise complies with
  the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
  any Contribution intentionally submitted for inclusion in the Work
  by You to the Licensor shall be under the terms and conditions of
  this License, without any additional terms or conditions.
  Notwithstanding the above, nothing herein shall supersede or modify
  the terms of any separate license agreement you may have executed
  with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
  names, trademarks, service marks, or product names of the Licensor,
  except as required for reasonable and customary use in describing the
  origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
  agreed to in writing, Licensor provides the Work (and each
  Contributor provides its Contributions) on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
  implied, including, without limitation, any warranties or conditions
  of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
  PARTICULAR PURPOSE. You are solely responsible for determining the
  appropriateness of using or redistributing the Work and assume any
  risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
  whether in tort (including negligence), contract, or otherwise,
  unless required by applicable law (such as deliberate and grossly
  negligent acts) or agreed to in writing, shall any Contributor be
  liable to You for damages, including any direct, indirect, special,
  incidental, or consequential damages of any character arising as a
  result of this License or out of the use or inability to use the
  Work (including but not limited to damages for loss of goodwill,
  work stoppage, computer failure or malfunction, or any and all
  other commercial damages or losses), even if such Contributor
  has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
  the Work or Derivative Works thereof, You may choose to offer,
  and charge a fee for, acceptance of support, warranty, indemnity,
  or other liability obligations and/or rights consistent with this
  License. However, in accepting such obligations, You may act only
  on Your own behalf and on Your sole responsibility, not on behalf
  of any other Contributor, and only if You agree to indemnify,
  defend, and hold each Contributor harmless for any liability
  incurred by, or claims asserted against, such Contributor by reason
  of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.
```

```
  To apply the Apache License to your work, attach the following
  boilerplate notice, with the fields enclosed by brackets "[]"
  replaced with your own identifying information. (Don't include
  the brackets!)  The text should be enclosed in the appropriate
  comment syntax for the file format. We also recommend that a
  file or class name and description of purpose be included on the
  same "printed page" as the copyright notice for easier
  identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

   http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
```

BSL 1.0

```
Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization
obtaining a copy of the software and accompanying documentation covered by
this license (the "Software") to use, reproduce, display, distribute,
execute, and transmit the Software, and to prepare derivative works of the
Software, and to permit third-parties to whom the Software is furnished to
do so, all subject to the following:

The copyright notices in the Software and this entire statement, including
the above license grant, this restriction and the following disclaimer,
must be included in all copies of the Software, in whole or in part, and
all derivative works of the Software, unless such copies or derivative
works are solely in the form of machine-executable object code generated by
a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT
SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE
FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.
```

BSD 2.0

```
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
   * Redistributions of source code must retain the above copyright
     notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above copyright
     notice, this list of conditions and the following disclaimer in the
     documentation and/or other materials provided with the distribution.
   * Neither the name of the <organization> nor the
     names of its contributors may be used to endorse or promote products
     derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
```

MIT

FTL

```
  We  specifically  permit  and  encourage  the  inclusion  of  this
  software, with  or without modifications,  in commercial products.
  We  disclaim  all warranties  covering  The  FreeType Project  and
  assume no liability related to The FreeType Project.


  Finally,  many  people  asked  us  for  a  preferred  form  for  a
  credit/disclaimer to use in compliance with this license.  We thus
  encourage you to use the following text:

   """
    Portions of this software are copyright © <year> The FreeType
    Project (www.freetype.org).  All rights reserved.
   """

  Please replace <year> with the value from the FreeType version you
  actually use.


Legal Terms
===========

0. Definitions
--------------

  Throughout this license,  the terms `package', `FreeType Project',
  and  `FreeType  archive' refer  to  the  set  of files  originally
  distributed  by the  authors  (David Turner,  Robert Wilhelm,  and
  Werner Lemberg) as the `FreeType Project', be they named as alpha,
  beta or final release.

  `You' refers to  the licensee, or person using  the project, where
  `using' is a generic term including compiling the project's source
  code as  well as linking it  to form a  `program' or `executable'.
  This  program is  referred to  as  `a program  using the  FreeType
  engine'.

  This  license applies  to all  files distributed  in  the original
  FreeType  Project,   including  all  source   code,  binaries  and
  documentation,  unless  otherwise  stated   in  the  file  in  its
  original, unmodified form as  distributed in the original archive.
  If you are  unsure whether or not a particular  file is covered by
  this license, you must contact us to verify this.

  The FreeType  Project is copyright (C) 1996-2000  by David Turner,
  Robert Wilhelm, and Werner Lemberg.  All rights reserved except as
  specified below.

1. No Warranty
--------------

  THE FREETYPE PROJECT  IS PROVIDED `AS IS' WITHOUT  WARRANTY OF ANY
  KIND, EITHER  EXPRESS OR IMPLIED,  INCLUDING, BUT NOT  LIMITED TO,
  WARRANTIES  OF  MERCHANTABILITY   AND  FITNESS  FOR  A  PARTICULAR
  PURPOSE.  IN NO EVENT WILL ANY OF THE AUTHORS OR COPYRIGHT HOLDERS
  BE LIABLE  FOR ANY DAMAGES CAUSED  BY THE USE OR  THE INABILITY TO
  USE, OF THE FREETYPE PROJECT.

2. Redistribution
-----------------

  This  license  grants  a  worldwide, royalty-free,  perpetual  and
  irrevocable right  and license to use,  execute, perform, compile,
  display,  copy,   create  derivative  works   of,  distribute  and
  sublicense the  FreeType Project (in  both source and  object code
  forms)  and  derivative works  thereof  for  any  purpose; and  to
  authorize others  to exercise  some or all  of the  rights granted
  herein, subject to the following conditions:

    o Redistribution of  source code  must retain this  license file
```

```
    (`FTL.TXT') unaltered; any  additions, deletions or changes to
    the original  files must be clearly  indicated in accompanying
    documentation.   The  copyright   notices  of  the  unaltered,
    original  files must  be  preserved in  all  copies of  source
    files.

  o Redistribution in binary form must provide a  disclaimer  that
    states  that  the software is based in part of the work of the
    FreeType Team,  in  the  distribution  documentation.  We also
    encourage you to put an URL to the FreeType web page  in  your
    documentation, though this isn't mandatory.

These conditions  apply to any  software derived from or  based on
the FreeType Project,  not just the unmodified files.   If you use
our work, you  must acknowledge us.  However, no  fee need be paid
to us.

3. Advertising
--------------

Neither the  FreeType authors and  contributors nor you  shall use
the name of the  other for commercial, advertising, or promotional
purposes without specific prior written permission.

We suggest,  but do not require, that  you use one or  more of the
following phrases to refer  to this software in your documentation
or advertising  materials: `FreeType Project',  `FreeType Engine',
`FreeType library', or `FreeType Distribution'.

As  you have  not signed  this license,  you are  not  required to
accept  it.   However,  as  the FreeType  Project  is  copyrighted
material, only  this license, or  another one contracted  with the
authors, grants you  the right to use, distribute,  and modify it.
Therefore,  by  using,  distributing,  or modifying  the  FreeType
Project, you indicate that you understand and accept all the terms
of this license.

4. Contacts
-----------

There are two mailing lists related to FreeType:

  o freetype@nongnu.org

    Discusses general use and applications of FreeType, as well as
    future and  wanted additions to the  library and distribution.
    If  you are looking  for support,  start in  this list  if you
    haven't found anything to help you in the documentation.

  o freetype-devel@nongnu.org

    Discusses bugs,  as well  as engine internals,  design issues,
    specific licenses, porting, etc.

Our home page can be found at

  https://www.freetype.org
```