

I S S N 0 3 8 6 - 5 8 7 8

土木研究所資料 第4415号

土木研究所資料

土石流・掃流状集合流動・掃流砂流の氾濫に関する
並列計算オープンソースプログラムの開発

Development of an Open Source Program for Parallel Computation
of Inundation by Stony debris flow, Immature stony debris flow and bed load

令和4年3月

国立研究開発法人土木研究所
土砂管理研究グループ 火山・土石流チーム

Copyright © (2022) by P.W.R.I.

All rights reserved. No part of this book may be reproduced by any means, nor transmitted, nor translated into a machine language without the written permission of the Chief Executive of P.W.R.I. except the source codes.

この報告書は、国立研究開発法人土木研究所理事長の承認を得て刊行したものである。したがって、本報告書の全部又は一部の転載、複製は、国立研究開発法人土木研究所理事長の文書による承認を得ずしてこれを行ってはならない。ただし、**fortran** ソースコードのライセンスはクリエイティブ・コモンズ表示-継承 4.0 国際 パブリック・ライセンス (CC BY-SA 4.0) である。

土石流・掃流状集合流動・掃流砂流の氾濫に関する 並列計算オープンソースプログラムの開発

(国研) 土木研究所 火山・土石流チーム
研究員 清水 武志
元交流研究員 高橋 佑弥
元主任研究員 藤村 直樹
元上席研究員 石井 靖雄

要 旨：

第 4 期中長期計画の研究課題「土砂移動の監視を踏まえた被害予測技術に関する研究」において、土石流・掃流状集合流動・掃流砂流による氾濫計算プログラムのデータ構造やアルゴリズムを検討し、計算時間を短縮化した共有メモリ型並列計算プログラムを開発した。本資料は、同プログラムを説明するものである。

キーワード：土石流、掃流状集合流動、掃流砂流、氾濫計算、オープンソース

第1部

ソースコードの解説

はじめに

2011年度に国土交通省は、土石流などの水と土砂の混合物の流れによる氾濫範囲を推定する数値計算プログラム（以下、土石流氾濫計算プログラムという。）を開発した。同プログラムは、土砂災害警戒区域等における土砂災害防止対策の推進に関する法律に基づき、重大な土砂災害の急迫している状況において土砂災害が想定される土地の区域の設定に活用するため、一つ又は隣接する数か所の流域の末端から流出した降雨により溪流の土石等が水と一体となった流れが到達し得る氾濫範囲を速やかに推定することを目的として開発されたものである¹⁾。2012年度から2014年度にかけて、土木研究所が国土交通省の土石流氾濫計算プログラムの安定化等を行なった成果は、同法で利用するために、2014年度（2015年3月31日）に国土交通省から国土交通省各地方整備局、北海道開発局、内閣府沖縄総合事務局（以下、各地方整備局等）に配布された土石流氾濫計算プログラムを含むアプリケーション「QUick Analysis system for Debris flow (QUAD)」(version 1.1)に搭載された²⁾。2016年度から2017年度にかけて、土木研究所は、第4期中長期計画（2016年度～2021年度）における「土砂移動の監視を踏まえた被害予測技術に関する研究」の一部として、計算速度向上のため並列化した土石流氾濫計算プログラムを作成した。その成果は、2018年5月に国土交通省各地方整備局等に配布されたQUAD (version1.2)に搭載された。

本資料は、国土交通省と土木研究所が開発した土石流氾濫計算プログラムのFortranソースコードを、その基礎方程式、差分方程式、高速化のための工夫、計算事例、入力データフォーマット等の解説書とともに取りまとめたものである。第1部に土石流氾濫計算プログラムのソースコード解説、第2部にFortranソースコードを示す。第1部の解説書は第2部のソースコードを忠実に解説するよう心がけた。また、実務者（行政官、技術者等）や学生に役立つように、本資料の範囲において土石流等の理論に関する基礎的な事項を丁寧に記述するよう努めた。本文の補足や相互参照を記すための脚注を多く設けるとともに、付録には原著論文を引用しながら本資料で用いた式の導出や関連文献に表れる類似式との関係を整理した。引用する文献や内容の解釈が正確ではない記述があるかもしれない。その点はご容赦願いたい。

本資料第1部の構成は次の通りである。本資料の概略を短時間で掴めるように、第1章と第2章に要点と計算事例を示す。第3章と第4章に、第1章に示す基礎方程式、差分方程式の詳細を解説する。第5章に、ソースコードの構成やサブルーチンの機能、高速化に関するデータ構造などを示し、第6章に記号表を示す。付録Aに第3章における式導出、付録Bに侵食・堆積速度式等の関係式を整理し、末尾に引用文献を年代順に並べた。

第2部におけるFortranソースコードの著作権者は国土交通省と土木研究所である。ライセンスは、「政府標準利用規約（第2.0版）」³⁾を参考に、クリエイティブ・コモンズ表示-継承4.0国際パブリック・ライセンス（以下、CC BY-SA 4.0）⁴⁾とする。したがって、利用者は、ソースコードの利用にあたり、次の2点を承諾し、遵守しなければならない。

- 1.著作権表示
- 2.継承（2次著作物を作った場合、CC BY-SA 4.0のライセンスを付与）

また、利用者は、いかなる利用に対しても以下の点を承諾したものとする。

- 1.利用者は商用利用や改変も含めて自由に利用できること
- 2.利用者の利用に対して、著作権者及び関係者は一切その責任を負わないこと
- 3.著作権者及び関係者は一切のサポートを行わないこと

1) 技術的検討の一部は、緊急調査時における数値計算の考え方・手法（田村ほか、2010；内田ほか、2011）、QUADの設計とQUAD-Lの開発（清水ほか、2012a,b）、QUAD-Vの開発（Yamakoshi *et al.*, 2013）として発表されている。

2) 国土交通省の事務連絡参照（全国治水砂防協会、2016、p.341）

3) 政府標準利用規約（第2.0版）https://www.kantei.go.jp/jp/singi/it2/densi/kettei/gl2_betten_1.pdf

4) クリエイティブ・コモンズ表示-継承4.0国際パブリック・ライセンス <https://creativecommons.org/licenses/by-sa/4.0/deed.ja>

目次

第1章	概要	1
1.1	プログラムの開発目的	1
1.2	対象土砂移動現象と計算条件	1
1.3	計算速度向上の工夫	2
1.4	基礎方程式	3
1.4.1	運動方程式（運動量保存則）	3
1.4.2	連続式（質量保存則）	4
1.5	差分方程式	5
1.5.1	運動方程式	5
1.5.2	連続式	6
第2章	計算事例	8
2.1	事例1：雲仙普賢岳において1993年4月28日に発生した土石流氾濫	8
2.1.1	土石流氾濫による土砂堆積範囲	8
2.1.2	パラメータ	9
2.1.3	地形データ	9
2.1.4	ハイドログラフ	10
2.1.5	結果：ケースa	12
2.1.6	結果：ケースb	12
2.2	事例2：熊本県一の宮町において1990年7月に発生した土石流氾濫	13
2.2.1	土石流氾濫による土砂堆積範囲	13
2.2.2	パラメータ	14
2.2.3	地形データ	14
2.2.4	ハイドログラフ	15
2.2.5	結果	17
第3章	基礎方程式	18
3.1	対象土砂移動現象と用語	18
3.1.1	土石流，掃流状集合流動，掃流砂流	18
3.1.2	本資料で用いる用語	19
3.2	各流砂形態の発生条件	20
3.2.1	土石流・掃流状集合流動の災害調査実績	20
3.2.2	河床勾配による土石流の発生・流下・堆積区分	22
3.2.3	土砂濃度による流砂形態の遷移	23
3.2.4	河床勾配と相対水深による発生領域の分類	23
3.2.5	河床勾配と平衡土砂濃度による流砂形態の分類	24
3.3	現象のモデル化	25
3.3.1	連続体のモデル化の一般論	25
3.3.2	本資料における基礎方程式	26
3.3.3	境界条件	28
3.3.4	本章における基礎方程式の配列	29
3.4	運動方程式	29
3.4.1	X方向	29
3.4.2	Y方向	30
3.5	流れの連続式	30
3.5.1	流れ全体の連続式	30
3.5.2	粗粒分砂礫の連続式	30

3.5.3	細粒分砂礫の連続式	31
3.6	流砂形態に応じた流れの特性	31
3.6.1	河床せん断応力	31
3.6.2	侵食・堆積速度式	32
3.6.3	平衡土砂濃度	33
3.7	河床位方程式（河床堆積層の連続式）	35
3.7.1	河床位方程式	35
3.7.2	河床堆積層の粗粒分砂礫の連続式	35
3.7.3	河床堆積層の細粒分砂礫の連続式	35
3.8	境界条件	36
3.8.1	流出境界	36
3.8.2	流入境界	36
3.8.3	水際境界	36
3.8.4	壁境界	36
第4章	差分方程式	37
4.1	離散化と計算条件	37
4.1.1	基礎方程式の離散化に関する一般的事項	37
4.1.2	計算条件	37
4.2	本資料における離散化の方法	38
4.3	リーブフログ法の特徴	38
4.4	スタガード格子と格子点・格子辺, 評価点	39
4.4.1	格子の変数配置と格子点・格子辺	39
4.4.2	基礎方程式の評価点	40
4.4.3	ステンシルにおける評価点と格子点の名称	41
4.4.4	添字のルール	41
4.5	演算子	41
4.5.1	差分演算子	41
4.5.2	平均演算子	42
4.6	運動方程式	42
4.6.1	X成分	42
4.6.2	Y成分	43
4.7	連続式	45
4.7.1	流れ全体の連続式	45
4.7.2	粗粒分砂礫の連続式	47
4.7.3	細粒分砂礫の連続式	47
4.7.4	河床位方程式	48
4.8	境界条件	48
4.8.1	流出境界	48
4.8.2	流入境界	49
4.8.3	水際境界	50
4.8.4	壁境界	50
4.9	物理的に不整合な計算結果の対処法	50
4.9.1	流動深と流量の修正	50
4.9.2	土砂濃度の修正	50
4.9.3	侵食・堆積速度式の修正	50
第5章	プログラムへの実装	51
5.1	未知量と出力	51
5.2	Fortran と OpenMP の概説	51
5.2.1	Fortran の実行と構成	51
5.2.2	OpenMP の概説	52

5.3	プログラム全体フロー	52
5.4	サブルーチン解説	53
5.4.1	前処理	54
5.4.2	運動方程式の処理	56
5.4.3	流れ全体の連続式の処理	56
5.4.4	流れの粗粒・細粒分砂礫の連続式と河床位方程式の処理	57
5.4.5	後処理1	58
5.4.6	後処理2	60
5.5	データ構造	62
5.5.1	従来の附番記法	62
5.5.2	本資料の附番記法	62
5.5.3	格子に関する用語	63
5.5.4	格子番号の取得方法	63
5.6	地形データ変換と隣接リストの作成	66
5.6.1	地形変換プログラムの入出力データ	66
5.6.2	格子番号の附番方法	66
5.6.3	スカラー評価点の接続リスト構築	67
5.6.4	ベクトル評価点 (X 方向) の接続リスト構築	69
5.6.5	ベクトル評価点 (Y 方向) の隣接リスト構築	70
5.6.6	流出境界の設定	71
5.7	入力データの書式	72
5.7.1	入出力ファイル名指定ファイル filename.dat の書式	73
5.7.2	地形データ (dem.dat) の書式	73
5.7.3	境界データ (boundary.dat) の書式	74
5.7.4	総時間・時間ステップ入力データ (time.dat) の書式	74
5.7.5	ハイドログラフ入力データ (hydr.dat) の書式	75
5.7.6	パラメータ入力ファイル (param.dat) の書式	77
5.7.7	入力ファイル間の整合	79
5.8	コンパイル環境	80
5.9	処理例	80
5.9.1	処理の仕方	80
5.9.2	実行環境	80
5.9.3	準備	81
5.9.4	入力データの地理座標系などメタデータの確認	81
5.9.5	地理座標系の変換	81
5.9.6	地形データの切り出し	82
5.9.7	JGD2011 longlat に座標系を付け替えて, AAIGrid 形式へ変換 (省略可)	82
5.9.8	作成した地形データのメタデータ確認	82
5.9.9	地形データの汎用解析入力形式への変換	82
5.9.10	ハイドログラフ入力格子点番号の取得	83
5.9.11	filename.dat の作成	85
5.9.12	計算実行	85
5.9.13	結果の可視化	85
第 6 章	記号表	86
6.1	基礎方程式・差分方程式に関する記号表	86
6.2	ソースコードの変数に関する記号表	88
付録 A	第 3 章の式導出	93
A.1	準備	93
A.1.1	単一粒径と混合粒径	93
A.1.2	記号の整理	93

A.1.3	状況設定	94
A.2	土石流	95
A.2.1	発生条件と移動堆積層厚	95
A.2.2	平衡土砂濃度と侵食・堆積過程	97
A.2.3	侵食速度式	100
A.2.4	堆積速度式	101
A.2.5	河床せん断応力	102
A.3	掃流状集合流動	105
A.3.1	平衡土砂濃度	105
A.3.2	河床せん断応力	106
A.4	掃流砂流	107
A.4.1	平衡土砂濃度	107
A.4.2	河床せん断応力	109
A.5	基礎方程式	109
A.5.1	3次元の連続式, 運動方程式 (ナビエ・ストークス方程式)	109
A.5.2	浅水流方程式の導出	111
付録 B	関連する類似式の整理	117
B.1	土石流発生下限勾配等	117
B.2	土石流の侵食速度式	117
B.2.1	飽和河床堆積層における侵食速度式	117
B.2.2	不飽和河床堆積層における侵食速度式	119
B.2.3	両者を総合した侵食速度式	121
B.2.4	乱流型土石流の不飽和河床堆積層における侵食速度式	121
B.2.5	侵食速度係数	121
B.3	土石流の堆積速度式	122
B.3.1	単一粒径の堆積速度式	122
B.3.2	混合粒径の堆積速度式	122
B.3.3	他の流砂形態の堆積速度式	123
B.3.4	堆積速度係数	124
B.4	掃流状集合流動の平衡土砂濃度	124
B.4.1	平衡と応力のつり合い, 水深の関係式	124
B.4.2	抵抗係数と流砂量との関係式	125
B.4.3	平衡土砂濃度の導出	125
B.4.4	平衡土砂濃度の他表現	126
B.4.5	無次元流砂量式などの類似表現や関連式	127

第1章 概要

本章では、本資料におけるプログラムの開発目的、対象土砂移動現象と適用できる計算条件、計算速度向上の工夫、基礎方程式と差分方程式の概略などを述べる。基礎方程式の詳細は第3章及び付録A、差分方程式の詳細は第4章で述べる。

1.1 プログラムの開発目的

本資料における土石流氾濫計算プログラムは、一つ又は隣接する数か所の流域の末端（谷出口¹⁾ともいう）から流出した降雨により溪流の土石等が水と一体となった流れが到達し得る氾濫範囲を速やかに推定することを目的として開発された（図1.1）。

そのため、実務で想定する対象事象発生直後に運用できるように、単純な地形条件などの下で単純なモデルにより安定して結果を出力できる計算を行うことを想定して作成されたものである。流砂形態に応じた土砂移動や河床変動計算もできるが汎用的な利用などを意図していないことに留意されたい。

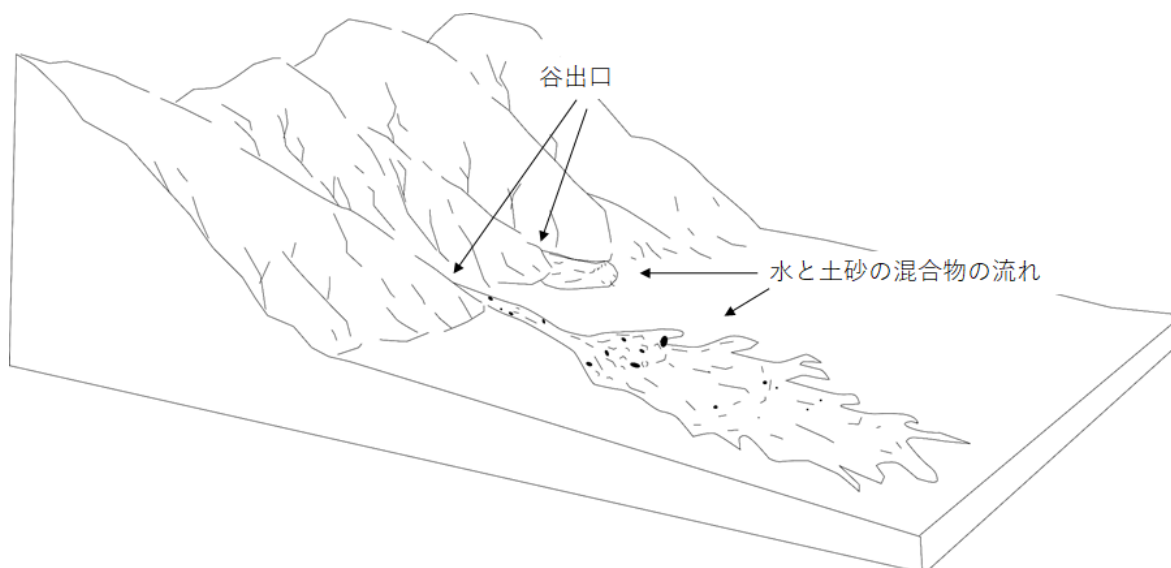


図1.1 水と土砂の混合物の流れの氾濫模式図（Varnes, 1978 を修正した Osanai *et al.*, 2010 に追記）

1.2 対象土砂移動現象と計算条件

本資料における土石流氾濫計算プログラムは、高橋保及び共同研究者が提案・発展させた土石流理論（例えば、高橋, 2004）に基づく基礎方程式を用いた。条件設定により降雨流出解析による氾濫範囲推定と、表面流によって河床堆積層と物質交換をしながら流下する土石流や掃流状集合流動、掃流砂流の氾濫範囲を推定を切り替えることができる。なお、土石流として非粘着性の粗い粒径の砂礫と水（細粒分砂礫も含む）で構成される石礫型土石流を想定する。

開発目的や様々な制約を踏まえた本資料における土石流氾濫計算プログラムの計算条件は以下の通りである。

- 流砂形態は、土砂濃度や勾配に応じて、土石流、掃流状集合流動、掃流砂流に遷移する。
- 流れによる河床の侵食・堆積過程（河床変動）を表現する。河床は計算領域の全域で移動床とする。固定床（岩床河川や舗装、道路など）は設定できない。

1) 日本地形学連合（編）（2017）では「谷口」と呼び、その定義は、河川が河谷から平坦地（低地・段丘面）に流下する地点または地区、である。谷出口はハイドログラフの流入境界条件の設定位置になるため、流入境界設定点あるいはハイドログラフ流入点と呼ぶことにする。

- 土地利用に応じた流れへの抵抗 (粗度係数など) は設定できず計算領域の全域で一定とする。
- 流域末端などの点において時系列流量 (ハイドログラフ) を流入境界として設定する。ハイドログラフは複数点に与えられる。
- ハイドログラフ以外の方法で計算領域内に水を与えられない (降雨や河川の常時流水などを与えられない)。
- 地形データは矩形格子で表現し、任意の位置座標に対して標高値を1つ有する形式とする。
- 標高は初期条件として地形データを与え、構造物や家屋は地形データとして反映させる。
- 暗渠内の流れ等、地形データとして表現できない流れは計算できない。
- 地盤への浸透流は表現されない。
- ハイドログラフに含まれる粗粒・細粒分砂礫の土砂濃度を設定でき、表面流による河床堆積物侵食型の土砂移動を扱うこともできる。

1.3 計算速度向上の工夫

本資料のソースコードは、OpenMP による共有メモリ型の並列計算を実行できるように、国土交通省が2011年度に開発したソースコードを改良したものである (高橋ほか, 2018)。OpenMP は単一メモリ・複数コアにおける計算環境 (パーソナルコンピュータなどマルチコア CPU とメモリでの実行環境) を対象として標準化された並列計算の手法のことである。OpenMP ではソースコードにコメントとして並列指示 (ディレクティブ) を記載するため、同一ソースコードを非並列・並列の両環境で利用できる利点がある。

図 1.2 に計算速度向上の結果を示す。ここで、比較対象プログラムは、2011年度プログラムを指し、従来型のデータ構造により次のように挙動する。

- 運動方程式と連続式の計算において、格子点を1点ずつ走査し流動深を基準として水域か陸域かを判定
- 矩形計算領域を構造格子で分割し、行・列方向に (i,j) の格子番号を付して格子点を管理するデータ構造

これらの点に対して次の改良を行った (図 1.3)。

- 時間ステップごとに計算対象の格子点を1度だけ走査して水域と陸域の判定を行った後、水域でのみ運動方程式と連続式を計算
- 格子間の接続を隣接リストによるグラフ関係で表現し、多角形による計算領域を設定できる並列計算可能なデータ構造

その結果、比較対象プログラムと比べて、1CPU で5倍以上の、8CPU では10倍以上の計算速度の向上を実現した。

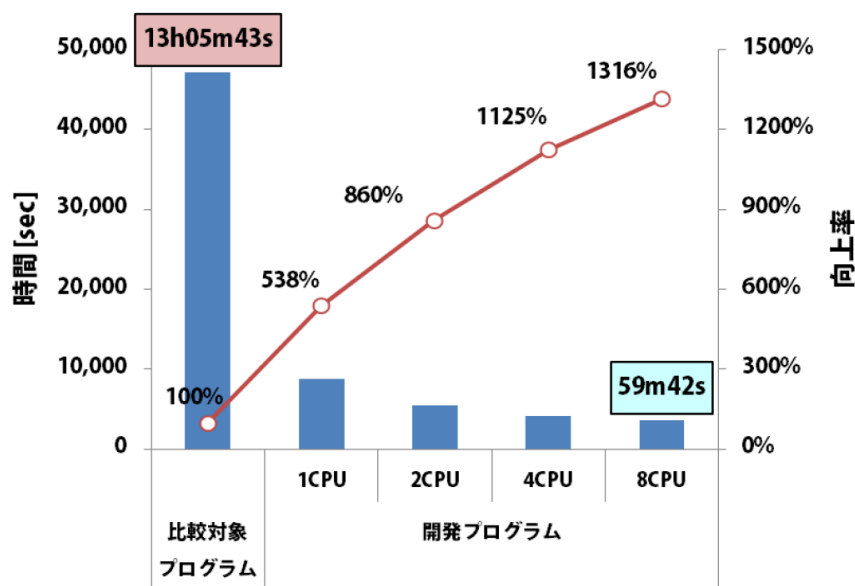


図 1.2 計算効率の向上

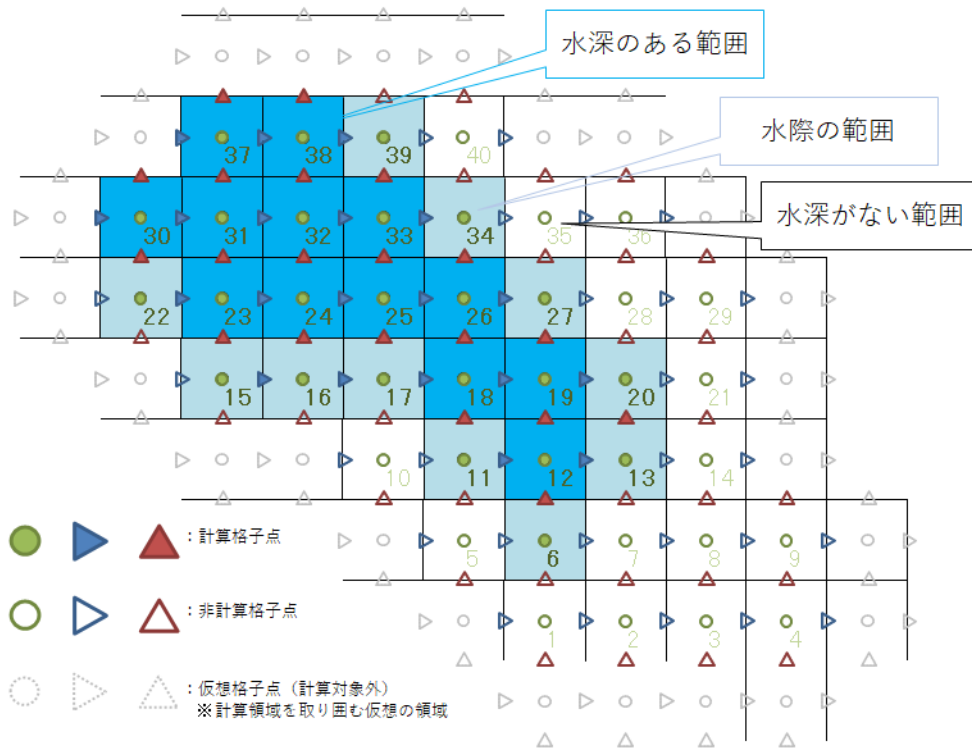


図 1.3 計算領域における水域・陸域格子と計算格子点・非計算格子点・仮想格子点の関係

1.4 基礎方程式

本節では、プログラムに実装される主な式を示す。式番号は、対照しやすいように、詳細を述べる第 3 章の式番号を用いる。基礎方程式は中川ほか (2002) とほぼ同様であり、侵食速度式は高橋 (2004) と川池ほか (2000) を組合わせた式、掃流砂流の平衡土砂濃度式は高橋ほか (1999) を用いた。水と土砂の混合物の流れの運動方程式は一流体の連続体として扱い、流量を計算する。運動方程式から得た流れ場を用いて、水域における流れ全体および各相（水、粗粒分砂礫、細粒分砂礫）それぞれにおいて連続式を解いて土砂や水の輸送を計算する。土砂濃度（固相の体積分率）で表現される各相の空間の和が検査体積となることから、4 本の連続式のうち独立な 3 本の連続式、つまり全体・粗粒分砂礫・細粒分砂礫の連続式を用いる。

以下、本章では基礎方程式を次のように配置する。

- 運動方程式（検査体積は、水と土砂の混合物の流れ）
 - 運動方程式右辺第 2 項の河床せん断応力
- 連続式一式（検査体積は、水と土砂の混合物の流れおよび河床堆積層）
 - 連続式右辺の侵食・堆積速度式
 - 侵食・堆積速度式を計算するための平衡土砂濃度

なお、各方程式で用いられる記号は、第 3 章や第 6 章の記号表を参照されたい。

1.4.1 運動方程式（運動量保存則）

水と土砂の混合物の流れ全体の運動方程式は流量 hu の変化を表現する式であり、ベクトル形式で示すと、

$$\frac{\partial [hu]}{\partial t} + \beta \nabla \cdot (huu) = -gh\nabla(z_b + h) - \frac{\tau_b}{\rho T} \quad (3.1)$$

である²⁾（Y 方向は式 3.2）。流砂形態に応じた流れの特性は右辺第 2 項の河床せん断応力に反映される。

2) ここに $\mathbf{u} = (u, v) = u_i$ ($i = x, y$) である。 $\nabla = (\partial/\partial x, \partial/\partial y) = \partial/\partial x_j = \partial_j$ ($j = x, y$) である。移流項のいくつかの表現を示す。 \mathbf{uu} はディアド積（ベクトルとベクトルからテンソル（行列）を作るテンソル積 \otimes ）の簡略表記であり、明示すると $\mathbf{u} \otimes \mathbf{u}$ である。行列形式で書くと \mathbf{uu}^T である。テンソル表記であれば $\partial/\partial x_j([hu]u_j)$ である。テンソル表記では、同じ添字で総和規約を適用し X 方向あるいは Y 方向をそれぞれ自由指標 i で記すと $\partial_j(hu_i u_j) = \partial_x(hu_i u_x) + \partial_y(hu_i u_y)$ となる。

河床せん断応力は、流砂形態によらず統一的に $\tau_b/\rho_T = f_r\|\mathbf{u}\|u$ と表現できるとすれば、流れの特性は流砂形態に応じた流動抵抗係数 f_r に表れる。粗粒分砂礫の土砂濃度によって流砂形態が変化する。

- 土石流³⁾: $0.4 C_{*L} < C_L$

$$\frac{\tau_b}{\rho_T} = \frac{1}{8} \left[\frac{\rho_T}{\sigma} \right]^{-1} \left[\left(\frac{C_{*DL}}{C_L} \right)^{1/3} - 1 \right]^{-2} \left(\frac{h}{d_L} \right)^{-2} \|\mathbf{u}\|u \quad (3.6)$$

- 掃流状集合流動: $0.02 \leq C_L \leq 0.4 C_{*L}$

$$\frac{\tau_b}{\rho_T} = \frac{1}{0.49} \left(\frac{h}{d_L} \right)^{-2} \|\mathbf{u}\|u \quad (3.7)$$

- 掃流砂流: $C_L < 0.02$

$$\frac{\tau_b}{\rho_T} = \frac{gn^2}{h^{1/3}} \|\mathbf{u}\|u \quad (3.8)$$

1.4.2 連続式 (質量保存則)

水と土砂の混合物の流れと河床堆積層に設定した検査体積において、全体および粗粒分砂礫、細粒分砂礫の連続式を立てる。移動床における侵食・堆積過程は、流れと河床の相互作用項 (湧き出し項) である侵食・堆積速度式で評価する。なお、侵食・堆積速度式 i は単位時間・単位面積あたりのみかけの土砂量 (間隙流体を含む) である。粗粒分・細粒分砂礫の連続式では、流れや河床堆積層の土砂濃度を乗じて実質の土砂量 (固体成分のみ) に変換している。 $i > 0$ は侵食過程, $i \leq 0$ は堆積過程を表す。

1.4.2.1 流れ全体の連続式

流動深 h の時間変化を表現する式である。

$$\frac{\partial h}{\partial t} + \nabla \cdot (h\mathbf{u}) = i \quad (3.3)$$

1.4.2.2 流れにおける粗粒分砂礫の連続式

検査体積における粗粒分砂礫の体積 $C_L h$ の変化を表現する式である。粗粒分砂礫の土砂濃度 C_L を求められる。

$$\frac{\partial C_L h}{\partial t} + \nabla \cdot (C_L h \mathbf{u}) = \begin{cases} i C_{*L} & (i \geq 0) \\ i C_{*DL} & (i < 0) \end{cases} \quad (3.4)$$

1.4.2.3 流れにおける細粒分砂礫の連続式

検査体積における細粒分砂礫の体積 $(1 - C_L) C_F h$ の変化を表現する式である。細粒分砂礫の土砂濃度 C_F を求められる。

$$\frac{\partial (1 - C_L) C_F h}{\partial t} + \nabla \cdot ((1 - C_L) C_F h \mathbf{u}) = \begin{cases} i(1 - C_{*L}) C_{*F} & (i \geq 0) \\ i(1 - C_{*DL}) C_F & (i < 0) \end{cases} \quad (3.5)$$

1.4.2.4 河床位方程式 (河床堆積層における質量保存則)

河床位 z の変化を表現する式である。河床勾配や河床堆積層の厚さを求められる。

$$\frac{\partial z}{\partial t} = -i \quad (3.20)$$

侵食・堆積速度式 侵食過程と堆積過程で異なる式を用いる。 C_∞ は流砂形態に応じて異なる式を用いる。

3) $[\rho_T/\sigma]^{-1}$ は $[C_L + (1 - C_L)\rho_m/\sigma]^{-1}$ である。後述の差分方程式ではこの関係式を用いて表記する。

- 侵食： $i \geq 0; C_L \leq C_\infty$

$$i = \delta_e \frac{C_\infty - C_L}{C_* - C_\infty} \|\mathbf{u}\| \quad (3.10)$$

- 堆積： $i < 0; C_L > C_\infty$

$$i = \delta_d \frac{C_\infty - C_L}{C_{*DL}} \|\mathbf{u}\| \quad (3.11)$$

平衡土砂濃度

- 土石流： $0.138 < \tan \theta$

$$C_\infty = \frac{\rho_m \tan \theta}{(\sigma - \rho_m)(\tan \phi - \tan \theta)} \quad (3.12)$$

- 掃流状集合流動： $0.03 < \tan \theta < 0.138$

$$C_\infty = 6.7 \left[\frac{\rho_m \tan \theta}{(\sigma - \rho_m)(\tan \phi - \tan \theta)} \right]^2 \quad (3.13)$$

- 掃流砂流： $\tan \theta < 0.03$

$$C_\infty = \frac{(1 + 5 \tan \theta) \rho_m \tan \theta}{\sigma - \rho_m} \left(1.0 - \bar{\alpha}^2 \frac{\tau_{*c}}{\tau_*} \right) \left(1.0 - \bar{\alpha} \sqrt{\frac{\tau_{*c}}{\tau_*}} \right) \quad (3.14)$$

1.5 差分方程式

本節では、プログラムに実装される主な差分方程式を示す。式番号は、対照しやすいよう、詳細を述べた第4章の式・図番号を用いる。離散化の方法として、高橋ほか(1986)に示される差分法を用いた。差分方程式は梶島(2014)を参考に差分演算子と平均演算子を導入して簡潔に表現する。各演算子の定義は第4章に述べる。

1.5.1 運動方程式

1.5.1.1 X方向

ベクトル評価点(X方向)で評価する。流量 $M_{(P)}^{n+2}$ を求められる。ステンシルは図4.6を用いる。移流項の差分式は、 $XDX = \partial[hu]u/\partial x, XDY = \partial[hu]v/\partial y$ と略記する。

$$\frac{M_{(P)}^{n+2} - M_{(P)}^n}{2\Delta t} + \beta \cdot XDX + \beta \cdot XDY = -g \overline{h_{(P)}}^x \delta_x [(h+z)_{(P)}] - f_r \hat{u}_{(P)} \|U_{(P)}\| \quad (4.10)$$

河床せん断応力 流動抵抗係数 f_r は、河床せん断応力 $\tau_b/\rho_T = f_r \|\mathbf{u}\| \mathbf{u}$ に由来し、それぞれの流砂形態に応じて次のように示される。演算子の位置と向きに注意。

- 土石流： $0.4 C_{*L} < \overline{C_{L(P)}}^x$

$$f_r = \frac{1}{8} \left[\overline{C_{L(P)}}^x + (1 - \overline{C_{L(P)}}^x) \frac{\overline{\rho_{m(P)}}^x}{\sigma} \right]^{-1} \left[\left(\frac{C_{*DL}}{\overline{C_{L(P)}}^x} \right)^{1/3} - 1 \right]^{-2} \left(\frac{\overline{h_{(P)}}^x}{d_L} \right)^{-2} \quad (4.14)$$

- 掃流状集合流動： $0.02 \leq \overline{C_{L(P)}}^x \leq 0.4 C_{*L}$

$$f_r = \frac{1}{0.49} \left(\frac{\overline{h_{(P)}}^x}{d_L} \right)^{-2} \quad (4.15)$$

- 掃流砂流： $\overline{C_{L(P)}}^x < 0.02$

$$f_r = \frac{gn^2}{(\overline{h_{(P)}})^{1/3}} \quad (4.16)$$

1.5.1.2 Y方向

ベクトル評価点 (Y方向) で評価する。流量 $N_{(P)}^{n+2}$ を求められる。ステンシルは図 4.7 を用いる。移流項の差分式は、 $YDX = \partial[hv]u/\partial x$, $YDY = \partial[hv]v/\partial y$ と略記する。

$$\frac{N_{(P)}^{n+2} - N_{(P)}^n}{2\Delta t} + \beta \cdot YDX + \beta \cdot YDY = -g\overline{h_{(P)}}^y \delta_y[(h+z)_{(P)}] - f_r \hat{v}_{(P)} \|V_{(P)}\| \quad (4.19)$$

河床せん断応力 流動抵抗係数 f_r は、河床せん断応力 $\tau_b/\rho_T = f_r \|u\|u$ に由来し、それぞれの流砂形態に応じて次に示される。演算子の位置と向きに注意。

- 土石流: $0.4 C_{*L} < \overline{C_{L(P)}}^y$

$$f_r = \frac{1}{8} \left[\overline{C_{L(P)}}^y + (1 - \overline{C_{L(P)}}^y) \frac{\rho_{m(P)}^y}{\sigma} \right]^{-1} \left[\left(\frac{C_{*L}}{\overline{C_{L(P)}}^y} \right)^{1/3} - 1 \right]^{-2} \left(\frac{\overline{h_{(P)}}^y}{d_L} \right)^{-2} \quad (4.23)$$

- 掃流状集合流動: $0.02 \leq \overline{C_{L(P)}}^y \leq 0.4 C_{*L}$

$$f_r = \frac{1}{0.49} \left(\frac{\overline{h_{(P)}}^y}{d_L} \right)^{-2} \quad (4.24)$$

- 掃流砂流: $\overline{C_{L(P)}}^y < 0.02$

$$f_r = \frac{gn^2}{(\overline{h_{(P)}}^y)^{1/3}} \quad (4.25)$$

1.5.2 連続式

スカラー評価点で評価する。ステンシルは図 4.5 を用いる。

1.5.2.1 流れ全体の連続式

流動深 $h_{(P)}^{n+3}$ を求められる。

$$\frac{h_{(P)}^{n+3} - h_{(P)}^{n+1}}{2\Delta t} + \delta_x[M_{(P)}^{n+2}] + \delta_y[N_{(P)}^{n+2}] = i_{(P)}^{n+1} \quad (4.28)$$

1.5.2.2 流れにおける粗粒分砂礫の連続式

流れの粗粒分砂礫量は $V_{L(P)} = C_{L(P)} h_{(P)}$ であり、 $C_{L(P)}^{n+3}$ を求められる。

- 侵食: $i \geq 0$

$$V_{L(P)}^{n+3} = V_{L(P)}^{n+1} + 2\Delta t \{ -\delta_x[q_{bLx(P)}^{n+2}] - \delta_y[q_{bLy(P)}^{n+2}] + i_{(P)} C_{*L(P)} \} \quad (4.43)$$

- 堆積: $i < 0$

$$V_{L(P)}^{n+3} = V_{L(P)}^{n+1} + 2\Delta t \{ -\delta_x[q_{bLx(P)}^{n+2}] - \delta_y[q_{bLy(P)}^{n+2}] + i_{(P)} C_{*DL(P)} \} \quad (4.44)$$

1.5.2.3 流れにおける細粒分砂礫の連続式

流れの細粒分砂礫量は $V_{F(P)} = (1 - C_{L(P)}) C_{F(P)} h_{(P)}$ であり、 $C_{F(P)}^{n+3}$ を求められる。

- 侵食: $i \geq 0$

$$V_{F(P)}^{n+3} = V_{F(P)}^{n+1} + 2\Delta t \{ -\delta_x[q_{bFx(P)}^{n+2}] - \delta_y[q_{bFy(P)}^{n+2}] + i_{(P)} (1 - C_{*L(P)}) C_{*F(P)} \} \quad (4.49)$$

- 堆積: $i < 0$

$$V_{F(P)}^{n+3} = V_{F(P)}^{n+1} + 2\Delta t \{ -\delta_x[q_{bFx(P)}^{n+2}] - \delta_y[q_{bFy(P)}^{n+2}] + i_{(P)} (1 - C_{*DL(P)}) C_{F(P)} \} \quad (4.50)$$

1.5.2.4 河床位方程式 (河床堆積層における連続式)

河床の標高 (河床位) $z_{(P)}^{n+3}$ を求め、河床堆積層厚 $z_{vL(P)}$ や $z_{vF(P)}$ など求められる。

$$z_{(P)}^{n+3} = z_{(P)}^{n+1} - 2\Delta t \cdot i_{(P)} \quad (4.56)$$

侵食・堆積速度式

- 侵食: $i \geq 0; C_{L(P)} \leq C_\infty$

$$i_{(P)} = \delta_e \frac{C_{\infty(P)} - C_{L(P)}}{C_* - C_{\infty(P)}} U \quad (4.40)$$

- 堆積: $i < 0; C_{L(P)} > C_\infty$

$$i_{(P)} = \delta_d \frac{C_{\infty(P)} - C_{L(P)}}{C_{*DL}} U \quad (4.41)$$

平衡土砂濃度 勾配は、 $\tan \theta = S$ と表現し直すと、水面勾配を用いることにすれば

$$S = \left(\delta_X(z_{(P)} + h_{(P)}) + \delta_Y(z_{(P)} + h_{(P)}) \right)^{1/2} \quad (4.30)$$

である。水中重量 $\gamma_s = \sigma / \rho_{m(P)} - 1$ (式 4.32) とすれば、流砂形態に応じた平衡土砂濃度は次のように示される。

- 土石流: $0.138 < S_{(P)}$

$$C_{\infty(P)} = \frac{S_{(P)}}{\gamma_s(\tan \phi - S_{(P)})} \quad (4.33)$$

- 掃流状集合流動: $0.03 < S_{(P)} < 0.138$

$$C_{\infty(P)} = 6.7 \left[\frac{S_{(P)}}{\gamma_s(\tan \phi - S_{(P)})} \right]^2 \quad (4.34)$$

- 掃流砂流: $S_{(P)} < 0.03$

$$C_{\infty(P)} = \frac{(1 + 5S_{(P)})S_{(P)}}{\gamma_s} \left(1.0 - \bar{\alpha}^2 \frac{\tau_{*c}}{\tau_*} \right) \left(1.0 - \bar{\alpha} \sqrt{\frac{\tau_{*c}}{\tau_*}} \right) \quad (4.35)$$

第2章 計算事例

本章では、次の災害を例として、本資料のプログラムによる計算結果を示す。

1. 雲仙普賢岳において1993年4月28日に発生した土石流氾濫
 - a. 1箇所の谷出口でハイドログラフを与える事例
 - b. 2箇所の谷出口でハイドログラフを与える事例
2. 熊本県一の宮町において1990年7月に発生した土石流氾濫

事例1は、火山噴火によって生産された土砂が降雨によって土石流として流下した事例である。ケース a. は1箇所の谷出口にハイドログラフを設定した事例、ケース b. は2箇所の谷出口にハイドログラフを設定した事例である。地形データは、当時の撮影された密着焼き空中写真をデジタル化した画像を用いて SfM-MVS で作成した。

事例2は、火山地域において降雨により土石流などが発生した事例である。地形データは、国土地理院の基盤地図情報数値標高データを用いた。

2.1 事例1：雲仙普賢岳において1993年4月28日に発生した土石流氾濫

1990年11月に噴火した雲仙普賢岳の山麓では、噴火後に土石流が頻発した。1993年4月28日から29日未明の大雨により赤松谷川およびおしが谷で発生した土石流を対象に計算を実施した。

2.1.1 土石流氾濫による土砂堆積範囲

土砂堆積範囲の実績を、計算結果と比較するために作成する。土砂堆積範囲は、対象土石流発生後の1993年6月17日に撮影された密着焼き空中写真を用いて SfM-MVS により作成した地形データとオルソ画像(清水ほか, 2020)を3次元的に表示し、オルソ画像の色調を目視で確認しながら、土砂が堆積したと考えられる範囲の境界をトレースして作成した(図2.1)。ただし、火砕流や土石流が複数回発生したことに起因する新旧堆積物の境界、火砕流の熱風部と考えられる森林域における微妙な色調変化による境界、道路や田畑の区画など細かく凸凹した境界などでは、境界線のトレースは曖昧である。なお、1993年4月28日の降雨による対象土石流の発生から同年6月17日の空中写真撮影までの間に、5月2日未明と6月13日に1回ずつ土石流が発生し、また各種工事などが実施された(例えば、太田2019, p.177-188)。



図 2.1 土砂堆積範囲の抽出(黒線)

2.1.2 パラメータ

計算に使用したパラメータを次表に示す。地形データの格子幅を 50 [m] とし、時間刻み Δt を 0.1 [s] とした¹⁾。

表 2.1 計算に使用したパラメータ

パラメータ	設定値	根拠文献
代表粒径 d_L [m]	0.01	平野ほか (1994)
砂礫密度 σ [kg/m^3]	2600	川池ほか (2000)
水密度 ρ [kg/m^3]	1000	川池ほか (2000)
マンニング粗度係数 n [-]	0.04	岩佐ほか (1980) ²⁾
内部摩擦角 $\tan \phi$ [-]	0.7	川池ほか (2000)
河床堆積層の土砂濃度 C_* [-]	0.6	土木学会水工学委員会 (2019, p.256)
河床堆積層の細粒分砂礫の土砂濃度 C_{*F} [-]	0.2	中川ほか (1996)
掃流状集合流動と掃流砂流の境界土砂濃度 [-]	0.02	中川ほか (1996)
限界掃流力 τ_{*c} [-]	0.05	岩垣 (1956)
侵食速度係数 δ_e [-]	0.0007	川池ほか (2000)
堆積速度係数 δ_d [-]	0.001	川池ほか (2000)
限界判定水深 ε [m]	0.001	岩佐ほか (1980)

2.1.3 地形データ

地形データは、図 2.2 に示す対象土石流発生前の 1991 年 6 月 16 日に撮影された空中写真から SfM-MVS により作成したものをを用いた (清水ほか, 2020)。この地形データの誤差は、4 点の地上基準点による平均自乗誤差で計ると、水平方向に約 8 [m]、垂直方向に約 4 [m] である。この精度は、土石流による堆積深などを検討するには必ずしも十分ではないが、現在のように精度の高い航空レーザ測量データなどが存在しない当時の地形データとして、氾濫範囲の概要を推定するためには次の 3 つの理由から有用と考えた：1) 対象土石流 (1993 年 4 月 26 日) 発生以前の当時の地形が表現されている、2) 世界測地系の平面直角座標系 (2 系) の地理情報を持つ、3) 計算格子の幅を約 50 m に設定したため水平方向に 1 格子以上ずれる可能性は低い。

計算領域の切り出しには地理院地図³⁾で作成した KML ファイル (座標系は世界測地系の緯度経度) を用いた (図 2.3 における赤色の範囲)。

- 1) この値は、CFL (Courant-Friedrichs-Lewy) 条件を考慮したものである。CFL 条件とは、格子幅を Δx 、時間ステップを Δt とし、物理現象の波の伝搬速度を u 、計算上の情報伝搬速度を $\Delta x/\Delta t$ としたとき、陽解法の時間積分法を用いて離散化した際、後者が前者を上回る制限、つまり、 $u(\Delta t)/\Delta x < 1$ を満足しなければ、移流を伴うその物理現象を計算できない制限のこと、をいう。本資料では、第 4 章に示すように、 $2\Delta t$ で基礎方程式一式を解くため、この説明における $\Delta t = 2\Delta t$ とし、表面流速を考慮して $C = U + \sqrt{gh}$ とする。時間刻みは大きいほど計算速度があがるが、発散の原因になるため注意が必要である。なお、CFL 条件は差分の収束に関する必要条件であり、安定を保証する条件ではない (例えば、安部ほか, 2014)。以上は、直観的な説明であるため、詳細は差分法に関する数値計算の教科書を参照されたい。
- 2) 岩佐ほか (1980) では洪水の氾濫解析の実施にあたり、マンニング粗度係数を、水田・田畑で 0.025、宅地では 0.04、山地河川では 0.06 に設定している。川池ほか (2000) は、中川ほか (1996) と同様に氾濫域では 0.04 を一律に与えた。本資料でもこれを参考とした。
- 3) 国土交通省国土地理院ウェブサイトにおける地理院地図による標準地図を転載した。本資料の他の箇所において「地理院地図」と記載した場合も同様である。URL: <https://maps.gsi.go.jp>

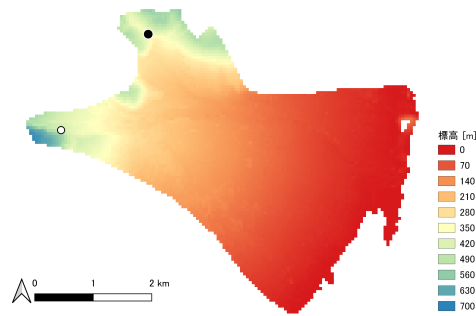


図 2.2 切り出した地形（計算の入力データ）

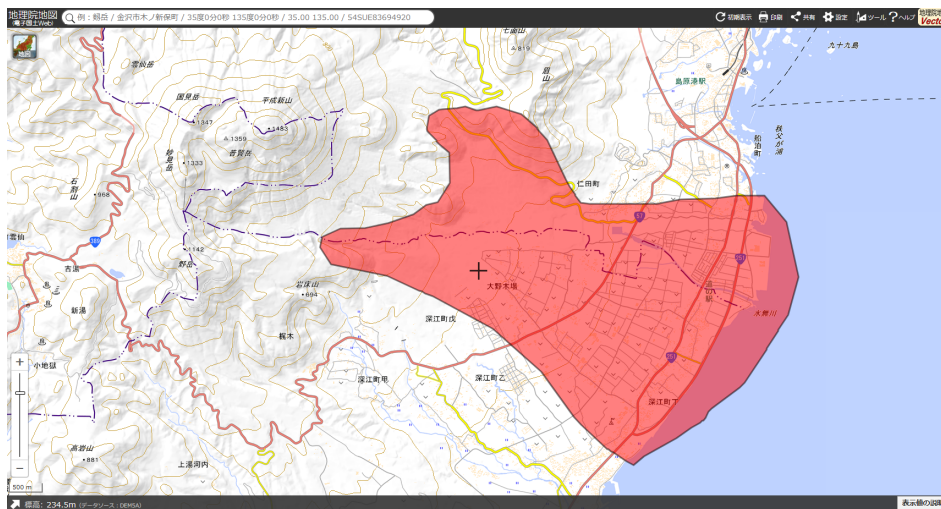


図 2.3 計算領域の切り出し（地理院地図を使用）

2.1.4 ハイドログラフ

赤松谷川およびおしが谷の谷出口にハイドログラフ流入点（本プログラムによる下流域の氾濫計算における流入境界）を設定した（それぞれ図 2.2 における白丸と黒丸）。その点より上流域で降雨流出解析を行い、氾濫解析の入力データとなるハイドログラフを求めた。

上流域の雨量データは気象庁雲仙測候所で観測された 60 分間雨量強度の時系列データである⁴⁾（図 2.4）。ここでは、入手できたデータが 1 時間雨量であったため 60 分雨量を用いる措置を採ったが、既往研究によれば 10 分雨量で計算することが適切である⁵⁾。図 2.5 に赤松谷川の流域界を示す。kinematic wave 法（特性曲線法）（水理委員会, 2002, 1-8）を用いた降雨流出解析によって、図 2.6 のハイドログラフを得た⁶⁾⁷⁾。これをハイドログラフ流入点に与え土石流氾濫計算の境界条件とする。

- 4) ハイドログラフは、気象庁雲仙測候所による当時の雨量データを用いた。当時 10 分雨量が観測されていなかったことから 1 時間雨量を用いた。
- 5) 奥田・土石流研究グループ（1973）や奥田ほか（1976）によれば、焼岳の土石流観測結果では毎 10 分間降雨量のピーク時刻と土石流発生時刻とが非常によく一致し、他の観測地でも同様であることが知られている。近年の災害を事例として計算する場合には 10 分雨量強度を用いることが望ましい。次章 3.2 節参照。
- 6) ここでは粗粒分砂礫と細粒分砂礫の土砂濃度は計算されない。
- 7) kinematic wave 法に inputs する地形条件は、赤松谷川では、上流の流域面積 4.8 [km²]、河道長 2,980 [m]、斜面勾配 0.44、平均河床勾配 0.20、河道幅 20 [m]、粗度係数 0.05 とし、降灰条件は浸透能 0 [mm/hr]、斜面の等価粗度 0.1 とした。おしが谷流域も地形計測値は異なるが他の係数は同様である。また、計算されたハイドログラフに土砂は含まれない。

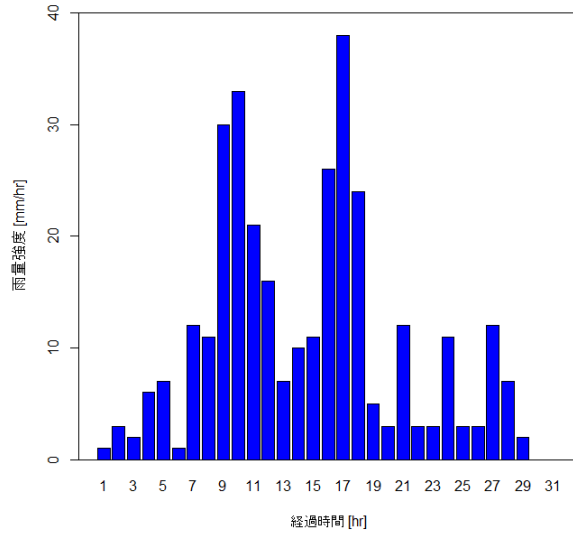


図 2.4 ハイドログラフを計算するための雨量条件

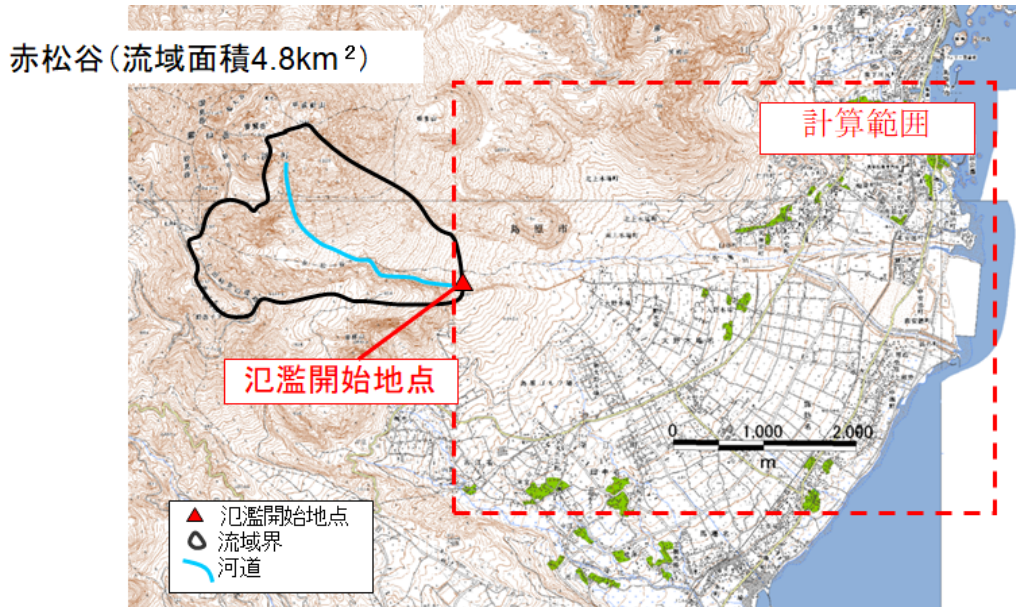


図 2.5 ハイドログラフを計算するための地形条件

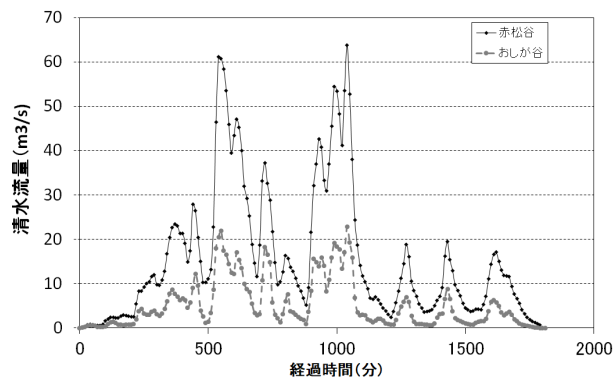


図 2.6 キネマティックウェーブ法で求めたハイドログラフ

2.1.5 結果：ケース a

ケース a は 1 箇所の谷出口でハイドログラフを与える事例である。

石礫型土石流は先頭部に巨石が集中する特徴があるが、実態として巨石を有する先頭部が流れる継続時間は 10 秒オーダーと短く、その後、数 10 分間は掃流状集合流動や後続流と呼ばれる細粒分砂礫からなる泥水が流れて、河床における大量の土砂が下流に輸送される (水山, 2003, 2020)。また、土石流で輸送された土砂が一旦堆積しても、掃流砂流の後続流など、低い土砂濃度の流れで再侵食される現象が生じる。流下後にみられる土砂堆積範囲と、水と土砂の混合物の流れによって被害が及ぶ恐れのある範囲は必ずしも一致しない。

水と土砂の混合物の流れの計算結果をどのように表現するのが適当かは、表現したい対象により様々である。ここでは、下流域 (比較的緩勾配地域) において土砂が到達し得る範囲を得たいとし、境界線を内田ほか (2011) が提案した方法で定めることにする。この方法は、計算時間を通じて、各位置の最大河床せん断応力が、対象粒径の (移動) 限界掃流力 (土砂が動き得る掃流力) を上回る位置を特定し、さらに、ハイドログラフ流入点からの連続する部分を連ねることによって、土砂が到達し得る範囲を推定する方法である⁸⁾。ある粒径の限界掃流力以上の掃流力が作用する範囲には、少なくとも掃流砂流が存在することになる。以上より、計算結果として、1 [cm] 以上の粒径の土砂が到達し得ると考えられる範囲を図 2.7 に水色で示す⁹⁾。

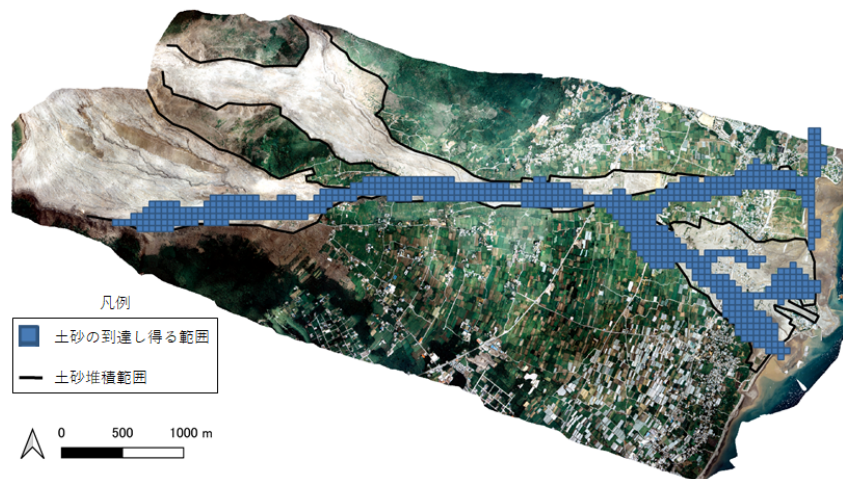


図 2.7 ひとつの流入境界による計算結果 (背景図は 1993 年 6 月 17 日のオルソ画像)

2.1.6 結果：ケース b

ケース b は、上述のケース a にハイドログラフの流入点を追加した設定であり、2 箇所の谷出口でハイドログラフを与える事例である。

-
- 8) 流速が遅くなると砂礫が停止しはじめる掃流力は「停止」限界掃流力である。本来、停止限界掃流力と掃流力との比で評価すべきと考えられるが、ここでは「移動」限界掃流力 (静止した河床の土砂に作用する掃流力を少しずつ増加させたときに、土砂が動き始める限界の掃流力である。通常これを限界掃流力と呼ぶ。) で評価している。2つの限界掃流力の関係は式 A.81 に示すように、 α^2 と移動限界掃流力の積が停止限界掃流力と等しい (芦田ほか, 1978)。芦田ほか (1978, 表 1) に示される α^2 の値を見ると、停止限界掃流力は、勾配が $\tan \theta = 0 \sim 0.1$ の範囲で、移動限界掃流力の 8 割から 6 割程度の大きさだと分かる。一方、第 4 章に示すサブルーチン τ_{*c} で用いる限界掃流力は、岩垣の式による「移動」限界掃流力 $\tau_{*c} = 0.05$ で一定値である。緩勾配での α^2 の値を考えると、停止限界掃流力は移動限界掃流力の 8 割程度、つまり約 0.04 となる。したがって、砂礫の停止限界の範囲を移動限界掃流力で評価する本手法は、停止限界掃流力を用いた場合に比べわずかに大きくなる可能性がある簡便法であるが、河床勾配による限界掃流力の変化などの計算を省略できる利点がある。
- 9) 本資料のプログラムで計算する場合、CFL 条件などを満たしても発散する可能性があるため、計算が終わったら CFL の大きさの確認、または、流量の大きさ等を見て計算が発散していないか、現実的にあり得ない数値が記録されていないかを確認するとよい。特に流砂も計算する場合、CFL 条件だけでなく侵食・堆積速度式などにより時間刻みを小さくしなければいけない場合があるため注意すること (太田ほか, 2018)。

ケース a と同様に、内田ほか (2011) の方法により 1 [cm] 以上の粒径の土砂が到達し得ると考えられる範囲を図 2.8 に水色で示す。

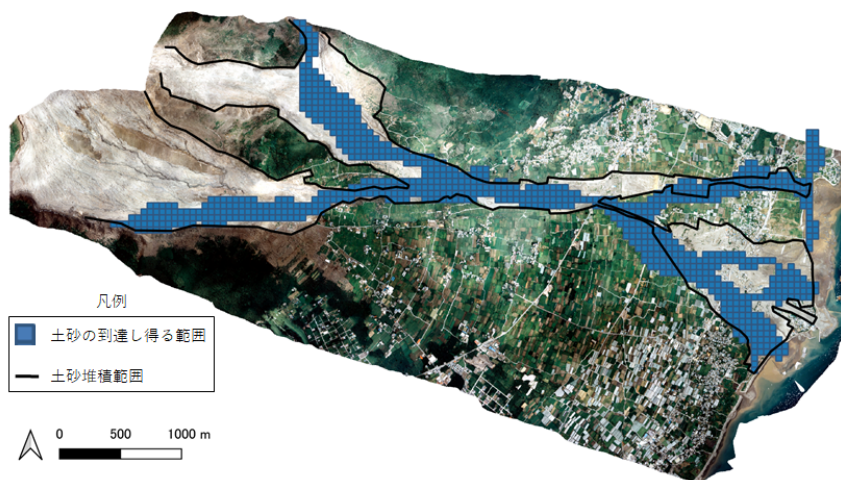


図 2.8 2つの流入境界を設けた計算結果（背景図は1993年6月17日のオルソ画像）

2.2 事例2：熊本県一の宮町において1990年7月に発生した土石流氾濫

1990年7月九州中北部を襲った豪雨により、阿蘇カルデラ内の古恵川において7月2日に土石流が発生し、熊本県一の宮町の百戸以上の家屋を倒壊させ8名の命を奪う災害が発生した。

流木が坂梨地区の河道を閉塞したことを契機に土石流が氾濫した（橋本ほか，1991）。本計算プログラムは流木による河道の閉塞は考慮できないが、地形データの解像度が低くもともと橋が表現されないため、大局的な地形により氾濫範囲を計算できると考えた。

橋本ほか（1991）に示された当時の降雨記録を用いて上流域における降雨流出解析を行い、そのハイドログラフを流入境界条件として下流域における水と土砂の混合物の流れの氾濫解析を実施した。

2.2.1 土石流氾濫による土砂堆積範囲

橋本ほか（1991，図1）による土砂堆積範囲を実績図として用いた（図2.9）。計算結果と重ね合わせられるように、GISを用いてトレースして地理情報を付すジオレファレンスを行った。

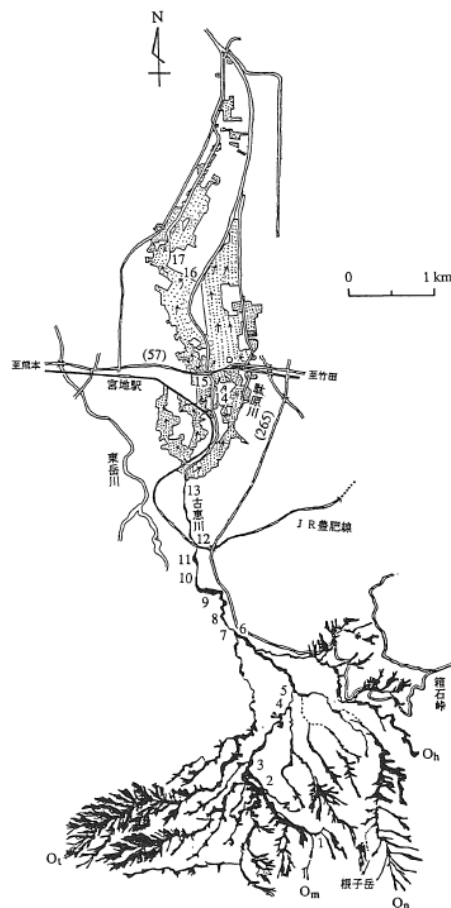


図 2.9 土砂堆積実績図 (橋本ほか, 1991)

2.2.2 パラメータ

事例 1 と同じ値を用いた (表 2.1)。

2.2.3 地形データ

地形データは事例 1 とは異なり、本資料執筆時に国土地理院の基盤地図情報数値地形標高データ 10m メッシュ (地形図の等高線から生成したメッシュ標高データ) をダウンロードして用いた (座標系は世界測地系の緯度経度)。この理由は、雲仙普賢岳の事例と異なり噴火や頻発する土石流によって地形が経時的に変化する訳ではないこと、当時の地形を復元するためのデータを入手できなかったこと、である。

計算領域の切り出しには地理院地図で作成した KML ファイル (座標系は世界測地系の緯度経度) を用いた (図 2.10 の赤色の範囲)。

格子幅をメートル単位で表現できるよう、両方のデータを古恵川が位置する平面直角座標系 2 系に変換した後に切り出し、図 2.11 に示す計算で使用する地形データを得た。

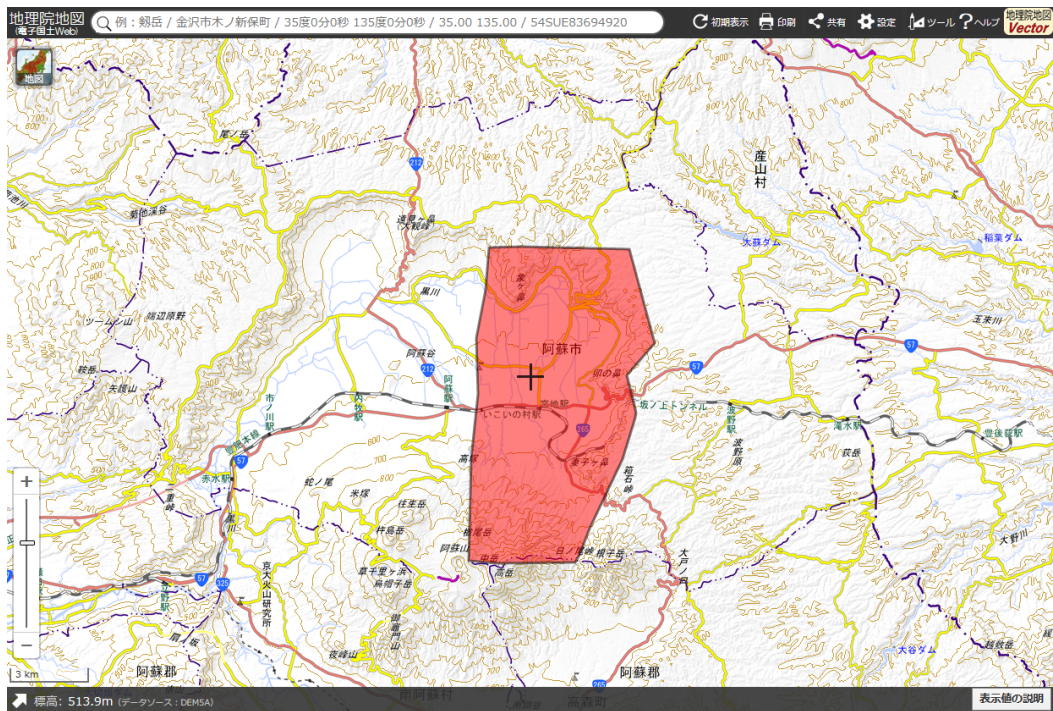


図 2.10 計算領域の切り出し（画面は地理院地図を利用）

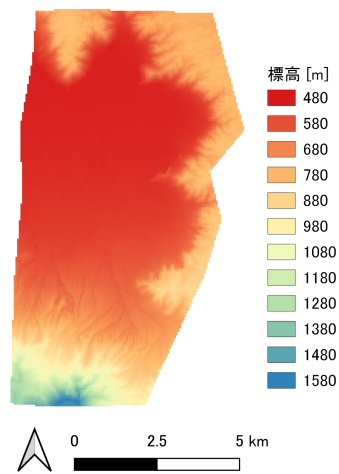


図 2.11 切り出した地形データ（計算の入力データ）

2.2.4 ハイドログラフ

図 2.12 に示す熊本県一の宮土木事務所観測による降雨記録（橋本ほか，1991，図 2）を用いて，事例 1 と同様に kinematic wave 法による降雨流出解析を行い谷出口（図では「氾濫開始地点」と表示）におけるハイドロ

グラフを算出した¹⁰⁾ (図 2.13)。前の事例と同様であるが、本来土石流発生の議論をするには 10 分間雨量強度を用いることが望ましいが(次章 3.2 節参照), 入手可能であった 1 時間雨量強度を用いた。

計算して得たハイドログラフを図 2.14 に示す。

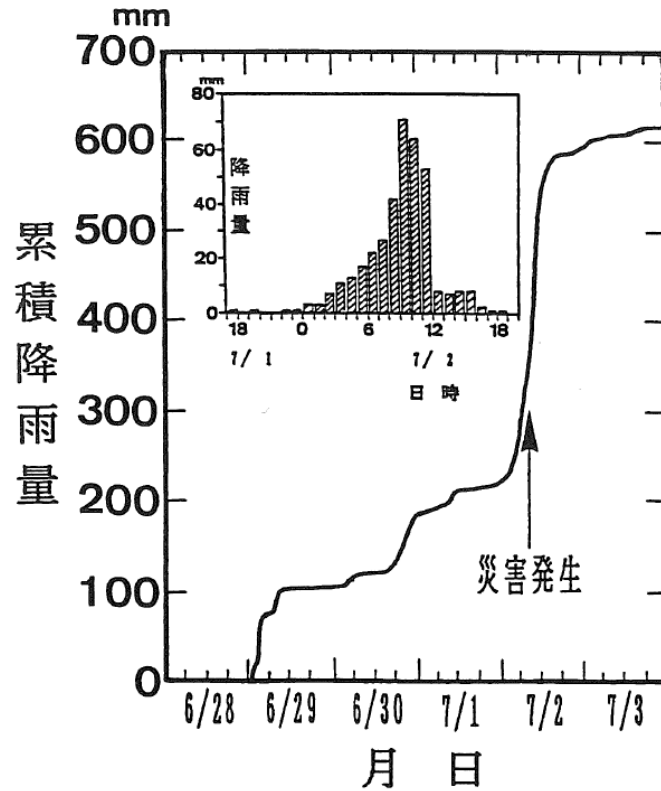


図 2.12 熊本県一の宮土木事務所による降雨記録(橋本ほか, 1991)

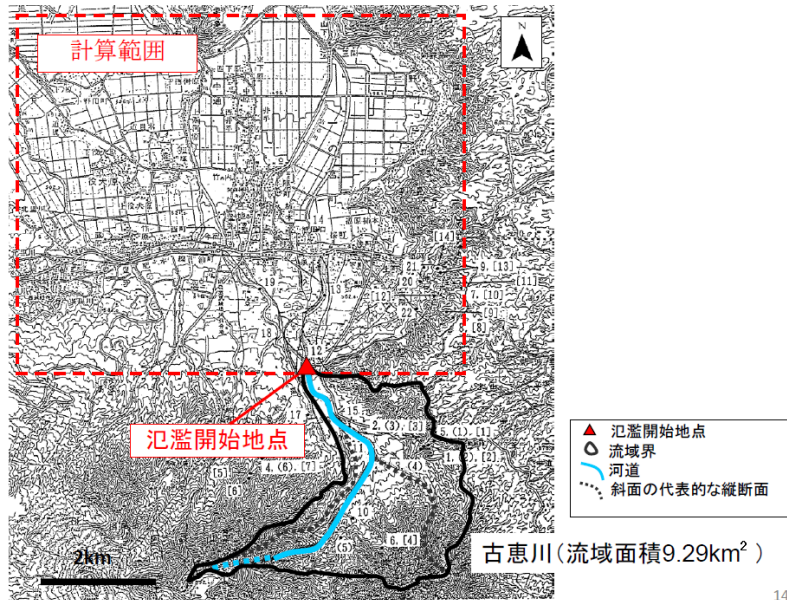


図 2.13 降雨流出解析と土石流等氾濫解析の区域

10) 地形条件は, 上流の流域面積 9.21 km^2 , 河道長 $4,960 \text{ [m]}$, 斜面勾配 0.32 (右岸), 0.15 (左岸), 平均河床勾配 0.05 , 河道幅 14.6 [m] , 粗度係数 0.04 , とし, 降灰条件として流域全体に降灰があるものとして浸透能 0 [mm/hr] , 斜面の等価粗度 0.1 とした。また, 計算されたハイドログラフに土砂は含まれない。

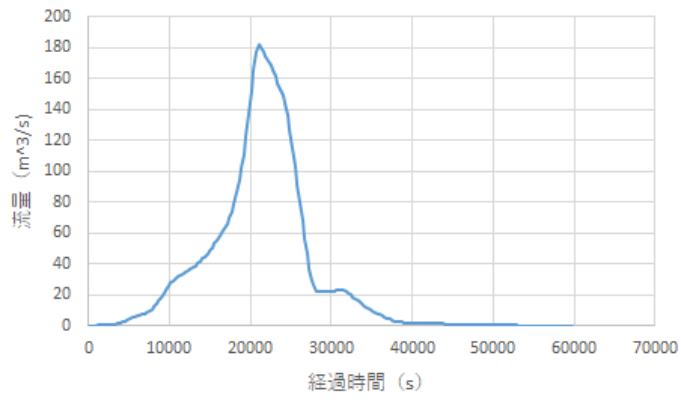


図 2.14 降雨流出解析結果（下流域の計算の流入境界の入力データ）

2.2.5 結果

本資料のプログラムによる氾濫解析の結果を図 2.15 に示す。

着色した範囲は、事例 1 と同様に 1 [cm] 以上の粒径の土砂が到達し得ると考えられる範囲である。

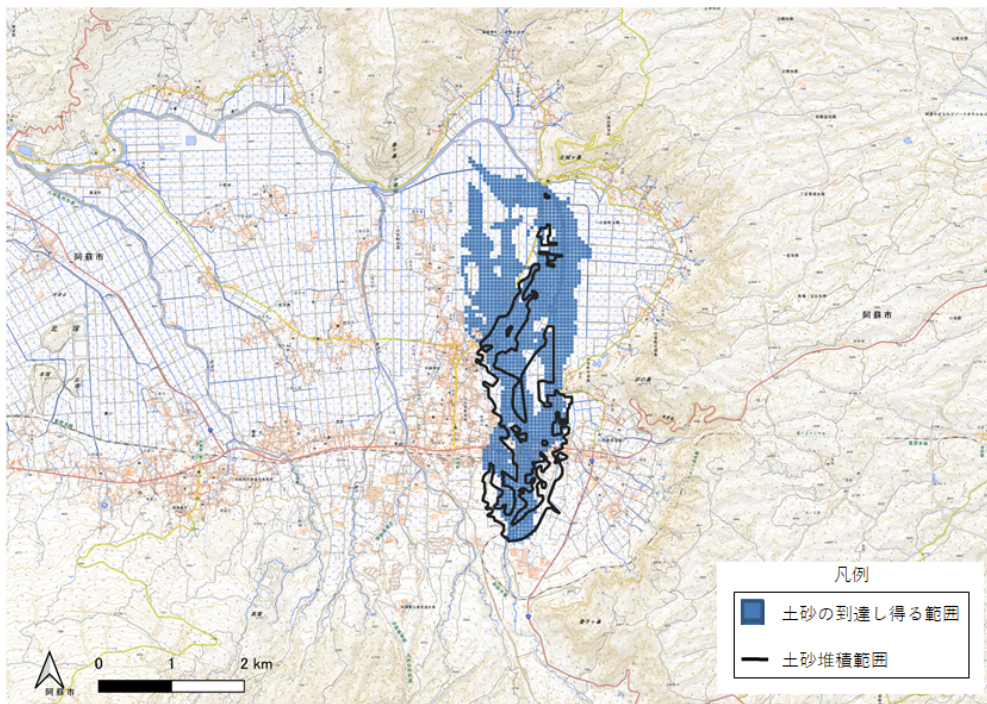


図 2.15 計算結果と実績の比較

第3章 基礎方程式

本章では、対象土砂移動現象と用語、各流砂形態の発生条件、現象のモデル化、基礎方程式及び境界条件を述べる。脚注には、式の導出あるいは式に現れる定数の根拠、他の文献との比較などの参考情報を示す。各式の導出の詳細は付録 A に示す。

3.1 対象土砂移動現象と用語

3.1.1 土石流、掃流状集合流動、掃流砂流

本資料で扱う土砂移動現象は、土石流、掃流状集合流動、掃流砂流である。それぞれ次のように説明される(図 3.1)。

- 土石流¹⁾とは、力学的には水と土砂礫との混合物の水の作用に基づく一種の連続体であるかのような流動である(高橋, 1983)。急勾配の流路を砂礫と水が混じり合って流れ下る現象である(土木学会水工学委員会, 2019, p.255)。
- 掃流状集合流動²⁾は、水山(1977)が初めて見出した流砂の移動形態³⁾で、流動の上層部は水流であり下層部は土石流と同じ流動メカニズムをもつ砂礫移動層の集合流動である⁴⁾。土石流と掃流砂流の遷移的な流れである(芦田, 1992)。
- 掃流砂流⁵⁾とは、個々の粒子(砂礫)が水との相対速度に基づく流体力をうけて輸送される各個運搬状態の流れである(例えば, 高橋, 1983, p.64)。砂の移動層の厚さは粒径の1~2倍程度である(芦田, 1992)。

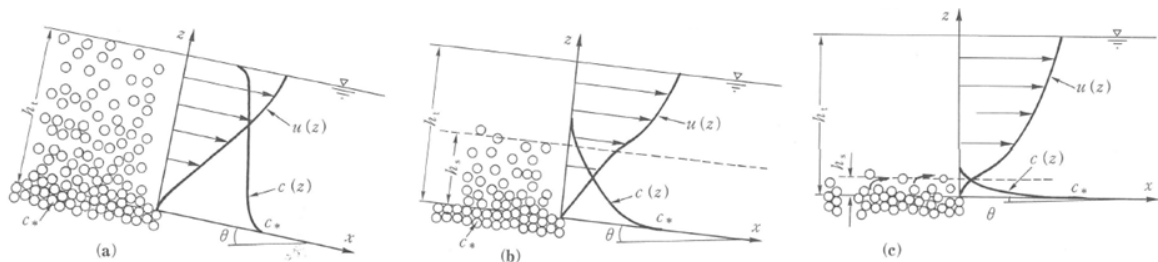


図 3.1 流砂形態の模式図(江頭・宮本, 2003 を編集): a) 土石流, b) 掃流状集合流動, c) 掃流砂流

- 1) 粒子衝突応力, 乱流混合による応力, 粘性流体の変形による応力のいずれが卓越するかによって, 石礫型土石流, 乱流(泥流)型土石流, 粘性型土石流あるいはその中間型に分類される(高橋, 2004, p.36-39; 高橋, 1992; 高橋ほか, 1996)。固体粒子が十分に接している時には準静的な摩擦による応力および粒子と流体の間に働く応力が存在するため, その力学的機構も考慮する分類もある。石礫型土石流と乱流型土石流を総括して慣性土石流と呼ぶ(高橋・里深, 1999; 土木学会水工学委員会, 2019, p.256)。本資料では, 日本における典型的な土石流のひとつである石礫型土石流を単に土石流と表現する。なお, マスメディア等の用法も含め「土石流」という用語の変遷は西本(2008)に詳しい。
- 2) 土石流, 不完全土石流(高橋, 1982)も同義語である(高橋, 2004, p.154-155)。「土砂流」は建設省で使われていた土石流の災害調査の中で, 石が多いと土石流, 砂が多いと土砂流とした材料の区分で, 流れの区分ではなかった(水山・江頭, 2001; 池谷・水山, 1982)。また, 海外では高濃度流れ(hyperconcentrated flow)ともいうが, 特に中国の黄河にみられるような河床付近の非常に高濃度な土砂が流れる状態を意識して使われる(水山・江頭, 2001)。日本では高濃度な細粒分砂礫の流れは泥流に分類することが多い。
- 3) 「掃流状集合流動」という用語の学術文献上の初出は水山(1980)であろう。芦田(1992)は芦田ほか(1978)を引用して水山ほか(1977)が名付けたと述べている。水山(1977)には水山(1980)と同様の内容が示される。以上の理由から, ここでは水山(1977)が見出したと記した。掃流状集合流動という名称は, 水路実験を行った際に水路を上から見ると掃流砂が流れているように見え, 側方から見ると砂礫流動層による集合流動が存在する流れであるため, とのことである。なお, 高橋(1977)ではこの流砂形態を Type I と表現し, 高橋(1980)では「掃流状集合流動」が用いられる。
- 4) 「掃流状集合流動」(水山, 1980)に関して, 水山・江頭(2001)により, 土石流と掃流状集合流動という専門用語の再定義案が提案されている。参考に引用する。「流れ全体に砂礫が分散しているものを土石流, 下層は集合流動で上層に水の層がある流れを掃流状集合流動または土砂流と呼んできましたが, この狭義の土石流は勾配約 14 度以上の限られた区間にしか存在しません。谷を流下して土石流災害を引き起こすものは, その先頭部を除けば流れとしてはほとんどの場合, 掃流状集合流動です。(中略)『土石流とは, 下層は集合流動で, 上層に水の層を有する流れで, 集合流動の厚さは勾配が急になるほど増加し, 勾配が約 14 度以上になると流れ全体に砂礫が分散するようになる。』と定義しなおすのが適当になったと考えてはどうでしょうか。」
- 5) 例えば, 芦田ほか(2008)に示される用語である。掃流あるいは掃流砂という専門用語の方が使用頻度が高い(例えば, 国土交通省, 河川・砂防技術基準)。ここでは, 掃流砂が砂の移動の形態を表し, その流れ, つまり掃流砂を伴う流れという意味で, 掃流砂流という用語を使用する。

3.1.2 本資料で用いる用語

土砂移動の現象は漸移的であるため、一つの用語である現象を明瞭に定義できないこともあって、ものの方や文脈に応じて用語が異なる場合がある。ここでは、用語の共通認識を得るために、本資料で使用するいくつかの用語を説明する。学術的な厳密な定義ではないので注意してほしい。なお、これらの説明は、芦田ほか(2008)、高橋(2004)、砂防学会編(2004)、江頭・宮本(2003)、吉川(1985)、松島ほか(2020)などの図書における定義や文脈上の使用方法を調べ、本資料の記述に適切と考えられるものを選択もしくは著者が解釈して示したものである。

- 土砂：岩塊、石礫、砂から微細粒子までいろいろな粒径のものを含んでいるがそれらを総称した名称（芦田ほか, 2008, p.27, 松島ほか, 2020, p.39）
- 流砂：流れによって運ばれる土砂（江頭・宮本, 2003, p.44）
- 流砂形態：流砂の運動形態
- 流砂現象：様々な流砂形態による運動や流れ
- 水と土砂の混合物の流れ：土石流、掃流状集合流動、掃流砂流、浮遊砂流など流砂形態を限定せずに流砂現象を指す総称。固液混相流を表す英語表現のひとつである particle-fluid mixture flow の訳語と考えられる。本資料では、土石流、掃流状集合流動、掃流砂流の流れを区別せずに総称する際に用いる。同義語として、水と土砂の流れ、流れ、流れ全体、固・液混相流（芦田ほか, 1973）、固体粒子と水の混合物の流れ（伊藤, 2001）、混合体の流れ（芦田ほか, 2008）、（広義の）土石流（砂防学会編, 2004）などもあり、本資料では適宜これらの用語も用いる。
- 土砂移動現象：流砂現象の他に、地すべり、崩壊など土砂の移動全般を含む用語である。関連して芦田(1985)を引用すると「流域内に存在している土砂礫が外力の作用によって移動したり、あるいは移動しやすくなる状態になることを土砂生産という。また、生産された土砂礫が、流域内のある地点を通過するとき、その量をその地点における流出土砂量と呼び、その現象を土砂流出という。しかし、実際はどこまでが土砂生産でどこからが土砂流出であるか明瞭に区分することは難しい場合が多く、総称して土砂の生産・流出現象とよくいわれる。」
- 土砂濃度：水と土砂の混合物の流れにおいて、ある閉空間内における土砂の占める体積の割合（固体の体積分率）、あるいは、その空間の表面を通過する総流量に対する流出土砂量の割合をいう。前者を体積土砂濃度あるいは土砂容積濃度という。後者を輸送土砂濃度あるいは輸送濃度という。本資料において、単に土砂濃度という場合、体積土砂濃度を表す。
- 平衡土砂濃度：ある瞬間のある位置において、その流砂形態の流れが河床を侵食も堆積もせずに輸送できる土砂濃度のことをいう。流れの土砂濃度がこれより大きいと堆積（流れから土砂が離脱）し、土砂濃度がこれより小さいと侵食（流れが河床から土砂を取り込む）するため、流れが保持できる最大土砂濃度である。
- 流動深：水と土砂の混合物の流れの水深をいう。本資料では水深も適宜用いる。
- 相対水深：流動深を代表粒径で割った無次元量で、流れに対する相対的な粗度の大きさを表現する。流れが砂礫等から受ける抵抗の指標のひとつである。
- 代表粒径：本資料では高橋(2004)等々に示される水と土砂の混合物の流れにおける粗粒分砂礫の代表粒径をいう。高橋(2004, p.120)は、「代表粒径という概念は曖昧であるが一応河床堆積物の平均粒径を用いる」と述べている。中央粒径が用いられる場合もある（例えば、高橋, 1982）。
- 砂礫：本資料では流砂および河床を構成する非粘着性の土砂を「砂礫」と呼ぶことにする。本資料では非粘着性材料しか扱わないことや粒度分布を持つ不定形状の粒状体を想起させるためである。土質材料は粒径によって呼び方が異なり、地盤工学会の区分では、石（粒径 75 mm 以上）、礫（2 mm 以上）、砂（0.075 mm 以上）、シルト（0.005 mm 以上）、粘土（0.005 mm 未満）であるが、本資料の「砂礫」は代表粒径が「砂」や「礫」の粒径区分に該当するという意味ではない。「石礫型土石流」の「石礫」も類義語であり、2 mm ~ 75 mm 以上の土質材料からなる土石流という意味ではない。

3.2 各流砂形態の発生条件

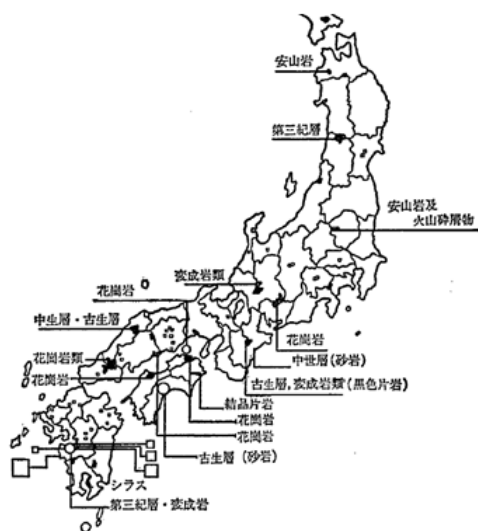
本節では、災害調査による土石流と掃流状集合流動（土砂流）の実態及び降雨と土石流発生実態について示した後、水と土砂の混合物の流れに関していくつかの分類を示す。

- 単一指標による分類
 - 河床勾配による土石流の発生・流下・堆積区分（実務で活用）
 - 土砂濃度による流砂形態の遷移
- 二指標による分類
 - 河床勾配と相対水深による発生領域の分類
 - 河床勾配と平衡土砂濃度による流砂形態の分類

これらの指標による流砂形態の分類は、後述する基礎方程式において、土砂濃度によって河床せん断応力が、河床勾配によって平衡土砂濃度が遷移する根拠となっている。

3.2.1 土石流・掃流状集合流動の災害調査実績

土石流および掃流状集合流動（土砂流）の災害調査の結果を図 3.2 及び図 3.3 に示す。これらの図は、池谷・水山(1982)が、建設省が調査した昭和 47 年から 52 年（1972 年から 1977 年）の 6 年間における土石流災害発生後の状況を整理したものである。図 3.2 に、発生した位置と地質（左図）、土石流発生件数、地質分類ごとの発生割合（右図）を示す。図 3.3 に、土石流発生後の調査資料を石礫型土石流（土石流）と掃流状集合流動（土砂流と表記：細粒土砂が流水により多量に流出したと判断される土砂の流れであり材料区分）に分けて、発生源と考えられる上流の崩壊地⁶⁾、流下部、堆積部の勾配を整理した図を示す。土石流は、平均的に 20 度以上で発生し、10 度以下で堆積することが分かる(水山, 2003)。実態のデータにはばらつきがある。様々な調査項目の最小値、最大値、最頻値を図 3.4 に示す。



岩 石	土石流発生件数A	土石流発生割合A	岩石の分布面積B	A/B
広域変成岩	36	9.3 %	3.6 %	2.54
花 崗 岩	180	46.3	13.3	3.48
塩基性および超塩基性貫入岩	—	—	1.6	—
第三紀および第四紀の火山岩	50	12.9	20.4	0.63
古生代堆積岩	38	9.8	12.2	0.80
中生代堆積岩	9	2.3	9.3	0.25
第三紀堆積岩	55	14.1	18.9	0.75
第四紀堆積岩	21	5.4	20.7	0.26
計	389	100	100	

図 3.2 1973 から 78 年に発生した土石流災害と地質別発生率 (池谷・水山, 1982)

6) 「土石流発生後の溪流調査で見られる崩壊が土石流になったかは明確ではない」(池谷, 1999, p.43)

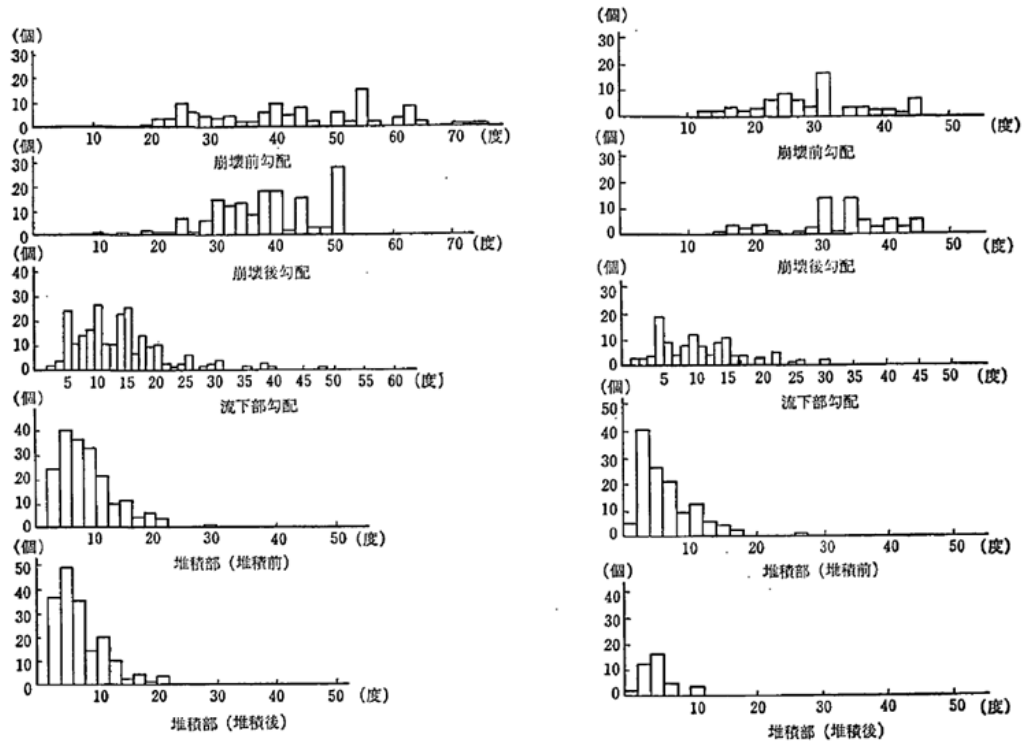


図 3.3 土石流（左図）と土砂流（右図）における崩壊地，流下部，堆積部の勾配 (池谷・水山, 1982)

項目	最小値	最大値	最頻値	備考
流域面積 (土石流)	0.2km ² 以下	10km ²	0.2~0.4km ²	
" (土砂流)	0.2km ² 以下	10km ²	0.2~0.4km ²	
崩壊前勾配 (土石流)	18°	76°	55°	
" (土砂流)	12°	46°	31°	
崩壊後勾配 (土石流)	10°	52°	52°	
" (土砂流)	14°	46°	31°	
流下部勾配 (土石流)	2.5°	49°	11°	
" (土砂流)	2.5°以下	31°	6°	
堆積部堆積前勾配 (土石流)	2°	22°	4~6°	
" (土砂流)	2°以下	18°	2~4°	
堆積部堆積後勾配 (土石流)	2°以下	22°	4~6°	
" (土砂流)	2°以下	12°	4~6°	
堆積土量 (土石流)	2,000m ³ 以下	100,000m ³	0~2,500m ³	ほとんど6,000m ³ 以下
" (土砂流)	2,000m ³ 以下	100,000m ³	5,000~7,500m ³	
堆積長 (土石流)	20m以下	1,000m	100~120m	
" (土砂流)	20m	1,000m	100~120m	
最大堆積幅 (土石流)	5m以下	450m	65m	ほとんど150m以下
" (土砂流)	5m以下	450m	95m	
堆積深 (土石流)	0.2m以下	5m	0.5m	ほとんど2m以下
" (土砂流)	0.2m	5m	0.5m	
分散角	5°以下	115°	15~20°	15°~45°が多い。
最大粒径 (土石流)	20cm	300cm	100cm	
" (土砂流)	20cm以下	300cm	20~40cm	ほとんど120cm以下
屈曲角 (土石流)	2°以下	110°	40°	
" (土砂流)	2°以下	110°	30°	
分散幅比 (土石流)	10以下	210	10~20	ほとんど80以下
" (土砂流)	10以下	210	—	

図 3.4 平均的な土石流の統計量 (池谷・水山, 1982)

次に、土石流発生時刻とその誘因のひとつである降雨との関係を述べる。奥田ほか (1972) によると、図 3.5 に示すように、土石流発生時刻と 10 分間雨量のピーク時刻がよく一致する。この図は、1970 年に国内で初めて土石流の科学的な総合的観測を実施した京都大学の奥田たちの土石流観測グループが、1971 年 9 月 6 日に焼岳上々堀沢左支の中腹左岸 Y 地点で計測したデータである (奥田ほか, 1972)。異なる地域でも土石流発生時刻と 10 分間雨量のピーク時刻との対応は同様に観測され、例えば、静岡県の大谷崩れ (Imaizumi *et al.*, 2005) や、三重県藤原岳西之貝戸川 (池田ほか, 2007) における報告がある。

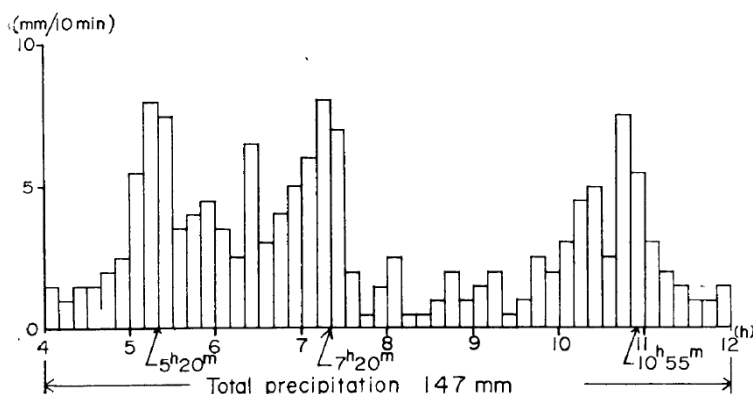


図 3.5 1971 年 9 月 6 日焼岳上堀沢の 10 分雨量と土石流発生時刻 (矢印) (奥田ほか, 1972)

3.2.2 河床勾配による土石流の発生・流下・堆積区分

砂防事業の実務で利用されている、勾配による土石流の流送区分を図 3.6 に示す (国土技術政策総合研究所, 2016)。前項に示した全国の土石流発生実績とその後の調査結果を踏まえて整理したものである。流れが遷緩点⁷⁾を通過すると、勾配が緩くなり平衡土砂濃度が小さくなるため、流れの中の余剰土砂が堆積する。流れの土砂濃度が減少すると流砂形態が変化する (次項参照)。

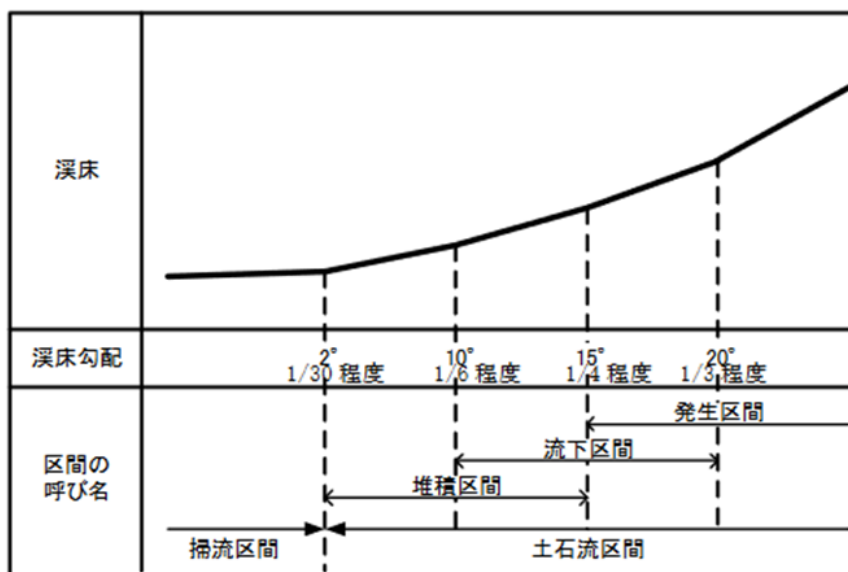


図 3.6 勾配による土石流の分類 (国土技術政策総合研究所, 2016)

7) 地表の縦断形が高所から低所に急激に緩傾斜になる地点 (日本地形学連合 (編), 2017)。急傾斜になる点は遷急点といい、総称を遷移点という。2次元の場合、傾斜変換線 (遷緩線と遷急線) という。傾斜変換線は細分すると傾斜方向変換線と傾斜角変換線の 2 種類に区別される。

3.2.3 土砂濃度による流砂形態の遷移

流砂形態の遷移は、土砂濃度を基準として、図 3.7 に示すように分類される⁸⁾ (高橋, 1992, 高橋, 2006, p.154)。斜面で発生した土石流が山地河川を流下する様子を想起できるように、図 3.7 における土砂濃度が高い方(右)から低い方(左)に向かって流砂形態との関係を、高橋 (2006, p.154) を参考に述べる。

最密充填土砂濃度 C_* は、斜面や河床の堆積土砂 (静止状態) の土砂濃度である。土砂濃度が C_1 になるとすべり面より上部が剛体的に滑動し、 C_2 になると砂礫が常に接触してクーロン摩擦力が作用しながら層状に流れはじめる。 C_3 になると砂礫の衝突により全層に砂礫が分散する流体である土石流となる。 C_4 になると上部の水流層と下部の砂礫移動層の 2 層に分離する掃流状集合流動 (図 3.7 では「不完全土石流」と表記) へ遷移する。掃流状集合流動における下層の砂礫移動層は、土石流と類似のメカニズム (衝突による粒子分散) の集合流動であるため、砂礫移動層に限ると土砂濃度は C_3 から C_4 の範囲に土砂濃度を保つが、上層の水流層も含めた全体としては C_4 から C_5 に土砂濃度が低下する。 C_5 になると砂礫粒径の 1~2 程度の厚さを持つ砂礫移動層厚において各個運搬で土砂が輸送される掃流砂流になる。

以上の土砂濃度の目安を高橋 (2006, p.154-155) に基づいて示す。括弧内に数値の根拠を示す⁹⁾。

- C_* : 0.644(Bagnold, 1966), C_1 : 0.604(Bagnold, 1966)
- C_2 : 0.555(Bagnold, 1966), C_3 : 0.51(Bagnold, 1966)
- C_4 : 約 0.2(高橋, 1982), C_5 : 0.02(高橋, 1982)

以上の知見は、運動方程式の右辺における河床せん断応力 τ_b について、土砂濃度を基準にして流砂形態が遷移すること、の根拠である。

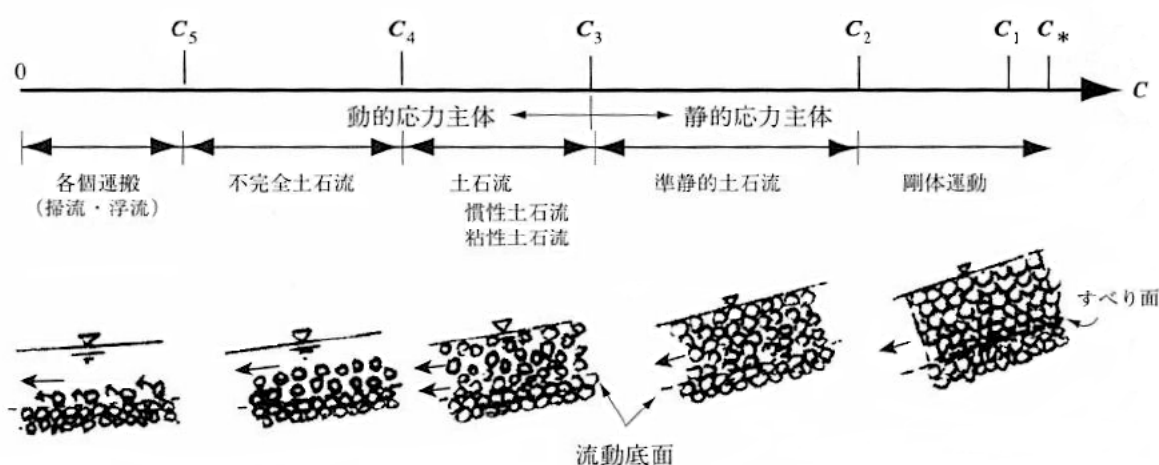


図 3.7 土砂濃度により変化する流砂形態 (高橋, 2006)

3.2.4 河床勾配と相対水深による発生領域の分類

高橋 (1980) は、図 3.8 に示すように、勾配と相対水深の平面上で様々な流砂形態の発生領域を分類した。掲載した図は、高橋 (1980) に示される詳細な情報を省き注記や勾配 (度) が加えられた芦田 (1988) から引用した。

掃流状集合流動発生下限勾配 (掃流砂との境界線) や土石流発生下限勾配線 (勾配約 14 度で一定の線: $\tan \theta_1$ と追記) を示す式は、高橋 (1977) 及び芦田ほか (1978) が提案した¹⁰⁾。土石流の発生下限勾配は物性値を与えると勾配だけの関数となり、実験などの物性値を使うと土石流発生域の下限勾配が約 14 度となる。こ

8) 高橋 (1992) において土砂濃度による流砂形態区分 (原著論文では流れのレジーム) が示され、高橋 (2006) はこれを分かりやすく図にまとめた、と考えられる。土砂濃度の添字 (星印を含む) は Bagnold (1966) による土砂濃度を基準とした状態分類と同一である。ただし、 C_4, C_5 は高橋 (1982) の実験結果を基に高橋 (2006) で追加された、と考えられる。

9) ここに示した土砂濃度の境界値は Bagnold (1966) における海岸砂による実験結果である。同論文では鉛球では $C_* = 0.74, C_2 = 0.63$, ワックス球では $C_* = 0.74, C_3 = 0.60$ という結果も示される。

10) 図 3.8 における、「土石流・崩壊」の注記の左の線の下限勾配 ($\tan \theta_2$)、「土石流」の注記の左の線 (土石流発生下限勾配 $\tan \theta_1$)、掃流状集合流動と掃流砂の境界線 (掃流状集合流動発生下限勾配) は高橋 (1977) が示した。詳細は付録 A に示したが、これらの式はそれぞれ式 A.6, 式 A.12, 式 A.10 である。また、「掃流」発生下限線は芦田ほか (1978) による掃流砂量式 (式 A.79) である。なお、「土石流・崩壊」の右の線は内部摩擦角と同じ傾斜で、 $\tan \theta = \tan \phi$ である。

れは、図 3.3 などに示す実績（特に左図の石礫型土石流の崩壊前勾配を見れば下限は約 14 度と分かる）や、実験から明らかになった土石流の発生勾配の下限値を説明した。

3.2.5 河床勾配と平衡土砂濃度による流砂形態の分類

平衡土砂濃度と勾配の関係を図 3.9 に示す。左図には、掃流砂流 (Bed load), 掃流状集合流動 (Immature stony debris flow), 石礫型土石流 (Stony debris flow), 乱流型土石流 (Turbulent muddy debris flow), 粘性型土石流 (Viscous debris flow) の平衡土砂濃度が示されている。右図は勾配約 14 度未満における掃流状集合流動と掃流砂流に注目した図である。勾配が大きくなるにつれて平衡土砂濃度が増加するが、水と土砂の流れが土砂で満たされるにつれて増加率が減少し、図 3.1 における C_3 と考えられる濃度に漸近する。水路実験において土石流先頭部の土砂濃度を計測した結果（高橋, 1977 など）によれば、その上限値を、高橋 (1980) は $0.8 \sim 0.9C_*$, 高橋・八木 (1983) は $0.9C_*$ と表現した¹¹⁾。

以上の知見は、連続式の右辺における侵食・堆積速度式 i について、勾配を基準にして流砂形態が遷移して平衡土砂濃度が決まること、および土石流では平衡土砂濃度に $0.9C_*$ の上限が存在すること、の根拠である。

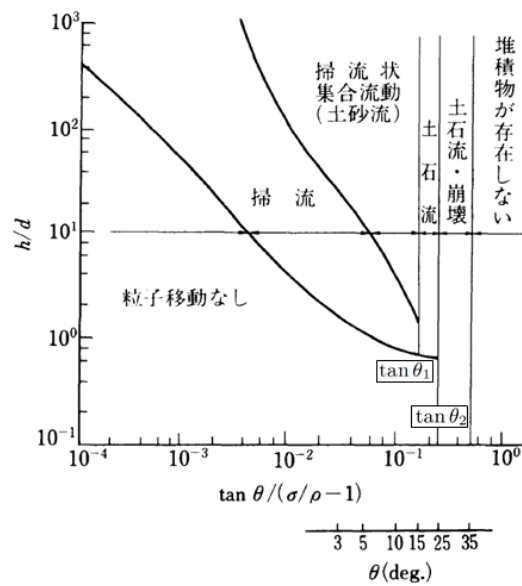


図 3.8 流砂形態の発生領域区分 (芦田, 1988 に追記)

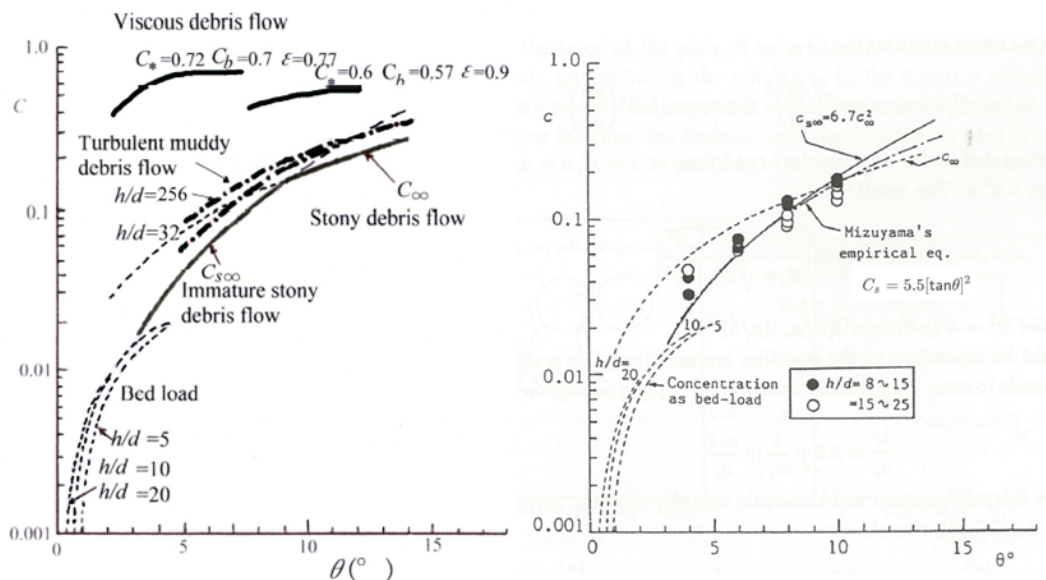


図 3.9 河床勾配と平衡土砂濃度, 流砂形態の関係。左図は様々な流砂形態 (Takahashi, 2009), 右図は勾配 14 度未満の掃流状集合流動のみを表示した図 (Takahashi, 1991 に数式を追記)

11) 数値例: $C_* = 0.6$ とすれば, 0.48 から 0.54 。であり, 概ね C_3 または C_2 の値を示す。

3.3 現象のモデル化

流砂現象を表現できる基礎方程式に、ある場所や条件における境界条件を与えると、原理的には解きたい問題の運動を推定できる。本節では次節以降の数式の背景にある考え方の理解に資するよう、まず連続体力学と混相流におけるモデル化の一般論を概説した後、本資料における基礎方程式の成り立ち、流砂形態の構成則（抵抗則）、河床との相互作用（侵食・堆積過程）、境界条件の種類を述べる。

3.3.1 連続体のモデル化の一般論

3.3.1.1 基礎方程式と連続体力学

基礎方程式とは、解きたい現象（本資料では水と土砂の混合体の流れ）を表す未知変数の数と同数の一揃いの方程式（多くの場合、複数の偏微分方程式）のことである¹²⁾。

有限の大きさをもつ変形する物体を連続体という¹³⁾。連続体力学（固体力学と流体力学の共通事項を整理した力学）では、連続体の中に閉空間を設定して（検査体積ともいう）、その空間内における質量および運動量、角運動量、エネルギーについて、内部の時間変化と表面を通過する収支のつり合い（保存則）による式を立てて運動を記述する¹⁴⁾。検査体積において、質量保存則を数式で表現した式を連続式、運動量保存則を数式で表現した式を運動方程式という。

連続式と運動方程式だけでは方程式が4つしかなく（3次元の場合）通常未知変数より少ない。また、流れの特性が表現されていない。そのため、未知数と方程式数が同数になるように、流れの特徴をモデル化した構成方程式（後述）や変数間の関係式が必要となる。

連続体力学の詳細は本資料を超えるため、詳しくは河床変動現象と基礎方程式が解説された宮本（1997）、土石流の基礎方程式から離散化まで解説した高濱（1998）、連続体力学や流体力学、固体力学等の教科書を参照されたい。

3.3.1.2 混相流

土石流は、粗粒分砂礫や細粒分砂礫、水が混合しながら流れる現象である。砂礫（固体）や水（液体）をそれぞれ「相」と呼び、このような流体を混相流（multiphase flow）あるいは混合体の流れ（mixture flow）という。固液混相流においては、液体を連続相、固体を分散相という。固相が粗粒分砂礫と細粒分砂礫の二成分からなる本資料の土石流は、二成分二相流である¹⁵⁾。土石流では粒子衝突応力が小さければ連続相の下層に分散相が集まり固相は空間内に不均質な分布¹⁶⁾を示す。この分布を検査体積内で平均化すれば、各相が均質に分布するようモデル化される。このとき、各相の分布を体積の割合（体積分率という）で表現できる。

体積平均の取り方により、流れ全体を平均化してひとつの流体とみなしてモデル化すると一流体モデル、各相を別々の流体とみなしてモデル化すると二流体モデルなどという（多相であれば多流体モデルという）。多流体モデルでは各相間の相互作用を表す項を考える必要があるが、全ての各相を足し合わせる一流体モデルでは相互作用項が作用・反作用の法則により相殺して零になるため考えなくてよい。従って、一流体モデルは、混相流の全体としての特徴を表現しつつ流体内部の相互作用を考えなくてよいモデルといえる。

また、相と相の流れる速度が一定でなければ界面に速度差（すべり）が発生する。この界面のすべりを無視できるモデルを均質流モデルという。

混相流の詳細は本資料を越えるため、例えば Gidaspow（1994）や芦田ほか（1994）、芦田ほか（1995）、日本

12) 未知変数の数と方程式の数が同一であることを「閉じる」と表現する。基礎方程式のことを支配方程式、基礎式などともいう。基礎方程式や支配方程式のことを、連続式や運動方程式、構成方程式、状態方程式、初期条件や境界条件などそれぞれを指し、一揃いの連立方程式を支配方程式系のように表現することもある。用法は様々である。

13) 固体や流体の総称である。質量があり大きさが小さい対象を質点、質点の集まりを質点系、変形しない連続体あるいは距離が変わらない質点系が剛体である。

14) 本資料の範囲では、角運動量保存則は応力テンソルが対称であることに反映される。エネルギー保存則は熱の流れなどがないため考えない。従って、質量保存則と運動量保存則のみを考える。ただし、エネルギー保存則は土石流の構成則を導く際に考える場合があるが本資料の範囲外である。

15) 相（phase）は固体、液体、気体を指す。水と油の混合体は液液二相流などという。また、化学工学などにおいて元素種別の輸送のように沢山の「種」や「成分」がある場合、x成分Y相流などと呼ぶこともある。本資料における混合粒径の土石流は、粗粒分砂礫の固相、細粒分砂礫の液相、水の液相からなる固液液三相混相流あるいは二成分の固相と液相から成る二成分二相流、と考えられる。

16) 均質あるいは一様（homogeneity）とは平行移動をしても性質が同一であること（場所によらないこと）をいう。ここでは全体としてムラがない、という日常用語の意味で使っている。なお、等方（isotropy）は回転しても性質が同一であること（方向によらないこと）をいう。

流体力学学会 (1991), 後藤 (2022) を参照されたい。

3.3.2 本資料における基礎方程式

本資料の基礎方程式は、中川ほか (2002) による基礎方程式を基にして、連続式における侵食速度式に高橋 (2004) と川池ほか (2000) を組合わせた式、掃流砂流の平衡土砂濃度に高橋ほか (1999) を用いたものであるが、同論文に示される式と完全に一致するわけではない。高橋ほか (1988b), 高橋ほか (1990), 高橋・中川 (1991), 中川ほか (1996), 中川ほか (1998), 高橋ほか (1999), 川池ほか (2000), 中川ほか (2001), 中川ほか (2002) に示される降雨の設定, 粗粒分や細粒分砂礫の扱い, 1次元や2次元計算方法などを参照して, 本資料における開発目的にあわせた平面2次元の混合粒径の基礎方程式としている。

以下, 本資料における基礎方程式の理解に資する浅水流化や混合粒径の扱い方, 流砂形態の特性の表現方法(土石流理論)を述べ, 基礎方程式の構成を示す。

3.3.2.1 浅水流方程式

本資料では水と土砂の混合物の流れが低地において氾濫する現象を表現する。このような氾濫や河川の流れを考えると, 水深方向に比べて水平方向のスケールが大きいので, 水深方向の詳細な分布は不要である。そこで, 3次元の連続式と運動方程式を河床から水面まで積分し, さらに流速分布などを平均化して得られる平面2次元の流れの方程式によって連続式と運動方程式を表現する。以上のように3次元の方程式から平面2次元の方程式を得る操作を浅水流化といい, 導かれた式を浅水流方程式という。導出の詳細を付録Aに示した。基礎方程式が浅水流化されることから, 本資料において, 例えば, 流速 u, v は断面平均流速であるように, 種々の水理量は水深方向の分布を持たない水深平均値 (あるいは断面平均値) である¹⁷⁾。

3.3.2.2 混合粒径の固液混相流における体積分率による表現

本資料では, 運動方程式には土石流等の流砂形態の流れの特性を表す構成則 (後述) を与えて, 流れ内部の相互作用と相間相対速度による応力を無視して全体の運動を記述する一流体・均質流モデルを用いる。一方, 連続式は, 運動方程式によって得られる流れ場において, 混相流の各相の輸送を推定するために多流体モデル (粗粒分砂礫, 細粒分砂礫, 水) とする。

本資料で扱う水と土砂の混合物の流れの混合粒径の扱いは高橋・匡 (1986) に基づくものである (図 3.10)。固相は粗粒分砂礫と細粒分砂礫の二成分からなる¹⁸⁾。粗粒分砂礫は粒子間摩擦や粒子衝突応力, 粒子骨格応力を担う。一方, 細粒分砂礫は, 沈降速度と乱れ速度の比によって決まる成分であり, 間隙流体中を浮遊してあたかも間隙流体の密度が大きくなったかのような役割を担う¹⁹⁾ (高橋ほか, 1990)。このことから, 粒径 d_L は粗粒分砂礫でのみ考える。

前述のとおり, 固体と液体が十分に混合していれば各相を体積分率で表現できる。従って, 固液混相流全体を考えると, 固相 (粗粒分砂礫) と液相 (細粒分砂礫が「溶けた」間隙流体) は, 検査体積における粗粒分砂礫の土砂濃度 (固相の体積分率) C_L によって, それぞれ $C_L, 1 - C_L$ と表される。間隙流体の体積 $1 - C_L$ における液相は, 細粒分砂礫の土砂濃度²⁰⁾ C_F によって, 細粒分砂礫と水の体積にそれぞれ $(1 - C_L)C_F, (1 - C_L)(1 - C_F)$ と表される。本資料では飽和を仮定し, 空気のみからなる空間 (空隙) は考えない。

河床堆積層における混合粒径の扱い方も基本的に同様である (図 3.10)。ただし, 細粒分砂礫の土砂濃度は静止するため固相と考える。また, 土石流の堆積過程において, 新たに堆積する河床堆積層の土砂濃度は, 流れとも元々の河床堆積物とも土砂濃度が異なると考えられることから C_{*DL} と定義される。

17) 浅水流方程式における流速などの水理量は断面平均量である。水深方向に平均されていない場合, 流速分布や土砂濃度分布という用語を使う。

18) 「砂礫」については, 粗粒分「土砂」や細粒分「粒子」などの表現もある。高橋・匡 (1986) を始め多くの文献では, 細粒分砂礫を「微細粒子」や「微細土砂」などといい, 浮遊して液体として振舞うことを想起する用語を用いる。ただし, 本資料では固体成分のことを不定形非粘着性材料を想起させる「砂礫」という用語で基本的に表現し, 粗粒・細粒として対比した用語を用いる。

19) この考え方によれば, 粗粒・細粒とは例えば土質力学や地盤工学における粒径による分類を意味せず, 流れ場の状態に依存したものであることに注意が必要である。液相と固相の遷移に関する実験的検討は Hotta and Miyamoto (2008) にレビューされ, 実験 (Sakai and Hotta, 2019) や数値計算 (Uchida *et al.* 2013; Sakai and Hotta 2019) の研究もある。間隙流体の密度を細粒分砂礫と水の密度の平均値 $\rho_m = C_F \sigma + (1 - C_F) \rho$ とすると, 細粒分砂礫と水の二相を一相に置き換えられる。このような操作を擬均質という場合がある (日本混相流学会, 2021, p.77)。

20) 細粒分砂礫の土砂濃度は, 粗粒分砂礫の体積を除いた検査体積における細粒分砂礫の体積の割合として定義する。ただし, 河床堆積層の土砂濃度については, 例えば, 高橋ほか (1991) や Uchida *et al.* (2013), 西口 (2015), 中谷ほか (2018) では, 本資料とは異なり, 粗粒分砂礫と細粒分砂礫の土砂濃度が共に, 検査体積に対する体積分率で定義される。文献によって定義が異なる場合があるため注意が必要である。

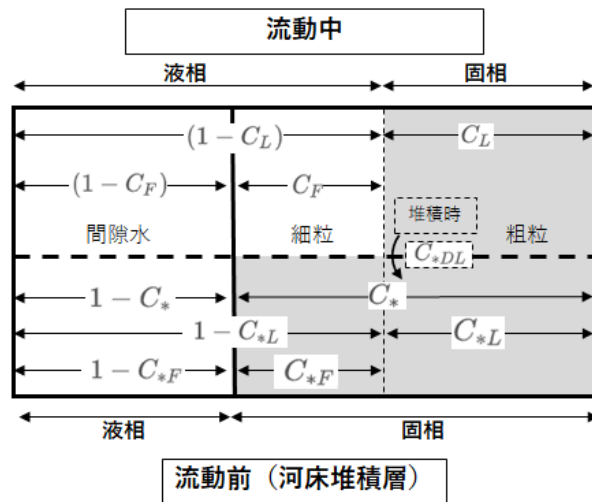


図 3.10 流動に関する土砂濃度と河床堆積層に関する土砂濃度の定義 (宮本・山下, 2003 を基に作成)

3.3.2.3 土石流の構成則

流砂形態による流れの特性は流れ内部に発生する応力とそれに対応する変形の指標であるひずみ速度 (速度勾配ともいう。) の関係で表される。この関係式を構成則や構成方程式という²¹⁾。本資料において構成則は、浅水流化した一流体モデルの運動方程式において、河床せん断応力を通じて流砂形態に応じた流れの特性を付与する役割を担うもので、抵抗則と呼ぶ²²⁾²³⁾。

土石流理論 (掃流状集合流動も含む) は流れの特性である構成則や様々な関連する現象の特性を体系化したものと考えられ、研究者による力学的な解釈の顕著な違いによって複数の理論が存在する²⁴⁾。国内外の土石流モデル (構成則) は例えば Hutter *et al.* (1994) や Iverson (1997) にレビューされている。国内で提案された土石流モデルは、高橋 (1977) や Takahashi (1978), 椿ほか (1982), 宮本 (1985), 江頭ほか (1989) などがあり、構成則の比較研究は例えば伊藤・江頭 (1998) や高濱ほか (2000b) がある。

伊藤 (2001) によれば、土石流の構成則に関する研究は、構成則を導く方法によって、1) 粒子間の衝突面あるいは接触面に作用する力を詳細に検討する方法、2) 応力モデルを規定しパラメータを決定する方法、3) 応力テンソルを保存力と散逸力 (せん断応力) に分離してエネルギー保存則で評価する方法、の3つに分類される²⁵⁾。上に示した高橋 (1977) や Takahashi (1978), 椿ほか (1982) は1) に、宮本 (1985) や江頭ほか (1989) は3) に、国外での多くの研究は2) に分類される。

本資料では、京都大学防災研究所における高橋保と共同研究者が主として提案・発展させた土石流力学の理論を用いる。本人によってまとめられた文献として、例えば高橋 (1980) や高橋 (1983), 高橋 (1984), 芦田 (1985), Takahashi (1991), 高橋 (2004), 高橋 (2006), Takahashi (2009), Takahashi (2014) がある。本資料ではこの理論を高橋土石流理論と呼ぶことにする。

本資料の範囲では、(石礫型) 土石流に対しては水深方向に一樣な濃度を持つ (層状に並んだ砂礫が隣接して浮かんだ状態の) 粒子が衝突して流れの応力を受け持っているモデル (高橋, 1977) を用い、掃流状集合流動

21) 構成則は、変形する物体 (連続体) における物体の経験的材料特性を一般的にいう用語であり、固体における応力と変形 (ひずみなど) や流体における応力と流動 (ひずみ速度など) の間に成り立つ関係式をいう。例として、弾性体のフックの法則、ニュートン流体の粘性法則などがある。本資料における土石流の構成則 (高橋, 1977) は非ニュートン流体のひとつダイラタント流体モデルである。

22) 浅水流化した開水路の流れにおいては構成則を抵抗則や流動測ということが多いようである。

23) 抵抗則は、流れを取り巻く固体壁面との間に作用する摩擦力によって流れがどれだけ抵抗を受けるかを表す式のことである (関根, 2005, p.23)。評価指標は様々であるが、本資料では断面平均流速と摩擦速度 (河床せん断応力を速度の次元で表示した指標) との比 (つまり流速係数 $\phi = u/u_*$) で表現する。例えば、水理学における流速分布に関する滑面や粗面の対数則 (本資料では掃流砂流がこれに相当)、土石流や掃流状集合流動の流速係数と相対水深が比例する式が抵抗則にあたる。異なる抵抗則の指標として、流動抵抗係数 f_r や Manning 粗度係数なども該当し、Darcy-Weisbach 則から $f' = 2gh \sin \theta / u^2$ ($\tau_b = \rho gh \sin \theta$) などと表現する文献もある (例えば、鈴木ほか 2003, Hotta and Miyamoto 2008)。本資料の範囲では、河床面粗度は掃流砂流以外では影響を考えていない (土石流の河床粗度の研究には、例えば、鈴木ほか 2003 がある。)

24) 伊藤・江頭 (1998) が示すように研究者によって力学的な解釈に顕著な違いが見られる。江頭・宮本 (2003) によれば、物理的・数学的考察を実証することが困難であることや、それぞれの研究者の現象のとらえ方や理解の仕方が異なることに起因するようである。

25) Miyamoto *et al.* (2019) による応力テンソルを中心にした分類も参考に示す (括弧内の具体的文献は本文参照)。1) テンソルの数学的な要請を満たすように定義された応力テンソル (多くの外国人の研究)、2) 衝突による運動量の交換を基にして定義された応力テンソル (バグノルド, 高橋や椿ほかの研究)、3) 連続体の物理に従った質量, 運動量, エネルギー保存の連立方程式を解くことで定義された応力テンソル (宮本や江頭ほかの研究)

に対しては、下層の砂礫流動層は土石流と同じ流れ、上層の水流層は乱流モデルを考慮した移動床河川と同じ流れ（掃流砂流）とするモデル（高橋, 1982）を用いる²⁶⁾。

3.3.2.4 侵食・堆積速度

土石流が河床と相互作用することで削剥過程（侵食・堆積過程の総称）が生じる。その現象を表現するためには流れの構成則だけでは不十分である。流れの河床せん断応力を考慮しつつ、流れの内部に流入出する土砂量やその反対に河床から流入出する土砂量の時間変化を評価することが必要である。その評価式を侵食・堆積速度式という。これらの式は、流れと河床との間の単位時間・単位面積あたりの質量の時間変化（質量速度）を表し、流れの連続式および河床高の変動を表す連続式に導入される²⁷⁾。

3.3.2.5 検査体積の設定と基礎方程式

以上のような基本的な考えを踏まえて、高橋ほか (1986) や高橋ほか (1999), 中川ほか (2002) を参考として本資料の基礎方程式が出来上がっている。

検査体積は、水と土砂の混合物の流れと河床に設定する（図 3.11）。それぞれの空間で成立する式を示すと、基礎方程式の構成は次の通りである²⁸⁾（第 1 章はこの順番で数式を配列）。

- 流れの検査体積
 - 運動方程式（浅水流化した一流体モデル）
 - * 流砂形態に応じて変化する抵抗則に応じた河床せん断応力（流れの特性）
 - 流れ全体の連続式，粗粒分砂礫，細粒分砂礫の連続式（浅水流化した多流体モデル）
 - * 流れと河床との相互作用を表現する侵食・堆積速度式
- 河床あるいは河床堆積層の検査体積
 - 河床堆積層における連続式である河床位方程式，粗粒分砂礫の河床位方程式，細粒分砂礫の河床位方程式（多流体モデル）
 - * 流れと河床との相互作用を表現する侵食・堆積速度式

なお、河床堆積層の方程式は、流れの水底面の境界条件として与える考え方もあるが（例えば、岩佐・松尾, 1995）、本資料では河床堆積層の連続式も概念上基礎方程式に含めた²⁹⁾。

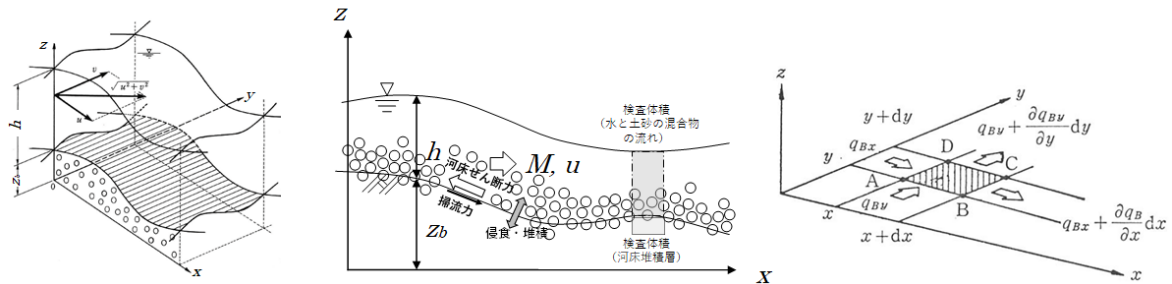


図 3.11 三次元の座標系（宮本・山下, 2003）と流下方向の流れイメージ，浅水流化された座標系（中川・辻本, 1986）

3.3.3 境界条件

連立偏微分方程式である基礎方程式は、原理的には境界条件と初期条件を与えると解けるが、解析的には解けないため次章に示すように数値的に解く。数値的に解く場合も解析的に解く場合と同様に、流量や地形など

26) 詳細は、付録 A の土石流の河床せん断応力の項に示す。

27) 侵食・堆積速度は、検査体積における湧き出し項として導入される。詳細は付録 A 参照。

28) 付録 A における「浅水流方程式の導出」の項に示すように、連続式の導出において、検査体積として河床堆積物から水面までを取った上で河床面で積分範囲を分割することで、流れと河床堆積層それぞれの連続式が求められる。従って、検査体積を流れと河床に設定するという表現は導出時には必ずしも適する訳ではないが、求められた結果はこのように解釈した方がわかりやすいと考えた。

29) 本資料では、検査体積を流体と河床堆積層に設定し、それぞれの検査体積の界面（河床）において侵食・堆積速度式（相互作用項）を設定することで、両者が結び付いていると解釈した。岩佐 (1995) によれば、数値水理学の数値モデルを作成する際には、水表面と河床はそれぞれ検査体積における自由表面と水底境界面であり、幾何学的条件、運動学的条件、力学的条件、物質の交換輸送条件を満たす境界条件として設定される。この考え方に従うと、河床は、流れの水底境界面であり物質の交換輸送条件を満たす境界条件と解釈される。

解析したい地域の情報である初期条件や境界条件を与える。

初期条件は地形などある。一方、境界条件は、本資料においては4種類あり、ハイドログラフを与える流入境界、計算領域端部における流出境界と壁境界、氾濫による移動境界の端部を決める水際境界である。

3.3.4 本章における基礎方程式の配列

本章の数式の配列は、第1章とは異なるが、保存則に対する比較的一般的な表現である方程式と、研究者により異なる土石流モデルに該当する式（河床せん断応力など）を認識しやすくするために次のように示す。次節以降を見るときに念頭におくと理解しやすいと考えられる。

- 流れの検査体積
 - 水と土砂の混合物の流れにおける運動方程式と連続式（比較的一般的な浅水流方程式）
 - 流れの特性に関する式（運動方程式右辺の河床せん断応力、連続式右辺の侵食・堆積速度式など、提案する研究者により表現が大きく異なる式）
- 河床位あるいは河床堆積層の検査体積
 - 河床堆積層における連続式（河床位方程式）
- 境界条件

3.4 運動方程式

運動方程式は、図 3.11 に示すように、水と土砂の混合物の流れに設定した検査体積（浅水流化したため、2次元であるが「体積」という用語を用いる）において、運動量とそれに換算できる物理量の収支が保存されることを表す式である³⁰⁾。つまり、X、Y方向それぞれの運動方程式は、検査体積内部の運動量の時間変化（左辺第1項）が、検査体積表面から流入する物体に付随する運動量の流れ（左辺第2項、第3項）および圧力（右辺第1項）と河床せん断力（右辺第2項）により供給される運動量の流れによりつり合うことを表す（詳細は付録A参照）。非圧縮流体を仮定して質量密度を一定とすれば単位幅流量³¹⁾が未知量であるから、本資料では単位幅流量の時間変化を表現する式と言える。

3.4.1 X方向

$$\frac{\partial q_x}{\partial t} + \beta \frac{\partial q_x u}{\partial x} + \beta \frac{\partial q_x v}{\partial y} = -gh \frac{\partial (z_b + h)}{\partial x} - \frac{\tau_{bx}}{\rho_T} \quad (3.1)$$

ここに、 x と y は空間座標、 t は時間、 q_x はX方向の単位幅流量（ $q_x = hu$ ）、 u, v はそれぞれX、Y方向の水深平均流速、 β は浅水流化に伴う運動量補正係数³²⁾、 g は重力加速度、 h は流動深（水と土砂の流れの水深）、 z_b は河床位、 τ_{bx} は河床せん断応力のX方向成分（詳細は後述）、 ρ_T は流れ全体の質量密度（水と土砂の混合物の流れの単位体積あたりの質量）である。流れ全体の質量密度は $\rho_T = C_L \sigma + (1 - C_L) \rho_m$ と表現でき、図 3.10 に示すように、 C_L は粗粒分砂礫の土砂濃度（水と土砂の混合物の流れにおける単位体積当たりの粗粒分砂礫の体積）、 σ は砂礫の質量密度、 ρ_m は間隙流体の質量密度（微細砂である細粒分砂礫を浮かべて重くなった間隙流体の単位体積あたりの質量）である。 $\rho_m = C_F \sigma + (1 - C_F) \rho$ であり、 C_F は細粒分砂礫の土砂濃度（粗粒分を除いた間隙部分の体積に対する細粒分砂礫の体積）、 ρ は水の質量密度である。土砂濃度における下付き添字 L は粗粒分、 F は細粒分を表す。

30) 運動方程式には応力（表面力）や体積力の項が存在する。今井(2003)によれば、連続体に「力が作用する」とは「運動量を受け取ること」である。外部から流れ（の検査体積）に表面力が作用すると、検査体積の境界（表面）を通じて運動量を受け取る。すなわち、応力は単位時間・単位面積あたりの運動量、つまり、運動量の流れである。また、体積力（重力など）は検査体積に単位時間・単位体積あたりに運動量が生成（湧き出しともいう；負なら消滅や吸い込み）する運動量源と言える。運動方程式は、運動量および運動量と等価に換算できる量の保存を表現した式である。ここに示す式は緩勾配のため重力項が落ちている（詳細は付録A参照）。

31) 一般に質量密度は位置と時間の関数 $\rho(\mathbf{x}, t)$ であるため、定数でないかぎり微分演算子の外に単純には出せない。本節に示す運動方程式の導出は付録Aに述べる。圧縮性流体として密度変化を考慮した基礎方程式の導出は、例えば伊藤・宮本(2002)、和田(2016, p.10-11)参照。

32) 運動量補正係数 β は、X方向では、移流項の各成分に対してそれぞれ β_{xx}, β_{xy} 、Y方向も同様に各成分に対してそれぞれ β_{yx}, β_{yy} と表記する（例えば高橋ら、1986）が、本資料では区別しない。 β は、土石流では1.25（例えば、高橋、1980）、掃流状集合流動及び掃流砂流では1.0（例えば、川池ほか、2000）を用いることが多い。土石流の値は、新井・水山(2011)に示されるように、Bagnoldの分散応力を土石流に適用した高橋によるダイラタント流体モデルでは理論的に導出される。掃流砂流のような乱流では、普通 $\beta = 1.0$ として扱われる（日野、1983, p.172）。また、高橋・藤井(1988)は、掃流状集合流動・掃流砂流の場合、便宜的に、 $\beta = 1.05 + 0.74C_L (C_L < 0.3)$ とする式を提示している。

3.4.2 Y 方向

$$\frac{\partial q_y}{\partial t} + \beta \frac{\partial q_y u}{\partial x} + \beta \frac{\partial q_y v}{\partial y} = -gh \frac{\partial(z_b + h)}{\partial y} - \frac{\tau_{by}}{\rho_T} \quad (3.2)$$

ここに、 q_y は Y 方向の単位幅流量 ($q_y = hv$)³³⁾、 τ_{by} は河床せん断応力の Y 方向成分である。

流量、流速、河床せん断応力は、後述する河床せん断力の項では q 、 u 、 τ_b のように、ベクトル表記を用いる。

3.5 流れの連続式

連続式は、水と土砂の混合物の流れの中に設定した検査体積において、質量の収支を表す式である(図 3.11)。粗粒・細粒分砂礫の土砂濃度(固相の体積分率)に応じて流れ全体の連続式を空間分割したものが、それぞれ粗粒分砂礫、細粒分砂礫の連続式である。

粗粒分砂礫及び細粒分砂礫の検査体積の表面を単位面積・単位時間あたりに通過する流砂量(流砂量フラックスあるいは流砂量流束という)は、流れ全体の流量($q_x = hu$, $q_y = hv$)に粗粒分・細粒分それぞれの土砂濃度を乗じて算出できるものと仮定する³⁴⁾。連続式右辺の侵食・堆積速度式(単位時間・単位体積あたりに河床から流れの検査体積に湧き出す質量)は、移動床において侵食・堆積過程(河床変動)が生じることを表す。河床が侵食されるときは、流れに土砂がとりこまれ検査体積内の質量が増加するため、流れの連続式右辺の侵食・堆積速度は正となる。 i は河床と流れで交換する全ての物体(固相と液相)の単位時間・単位面積あたりの交換量(つまり、みかけの土砂の速度)を表すため、粗粒分砂礫、細粒分砂礫の連続式において実質の固体の量に変換するために土砂濃度を乗じる。

水、粗粒分砂礫、細粒分砂礫の連続式の和をとると、流れ全体の連続式となる。4本のうち3本が独立であるため、水の連続式以外の3つの連続式を用いる。

3.5.1 流れ全体の連続式

流れ全体の連続式は流動深 h の時間変化を表現する式である。

$$\frac{\partial h}{\partial t} + \frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y} = i \quad (3.3)$$

ここに、 i は侵食・堆積速度である(詳細は後述)。

3.5.2 粗粒分砂礫の連続式

粗粒分砂礫の連続式は検査体積における粗粒分砂礫の体積 $C_L h$ の変化を表現する式であり、粗粒分砂礫の土砂濃度 C_L を求めるために用いる。右辺の侵食・堆積速度式において、侵食過程では($i \geq 0$)河床堆積層から離脱した土砂量が流れに取り込まれるため、河床堆積層の土砂濃度 C_{*L} をみかけの侵食速度 i に乗じて粗粒分砂礫の実質土砂量とする。一方、堆積過程では($i < 0$)、同様の理由によって、流れから土砂が離脱した時の河床堆積層の土砂濃度 C_{*DL} を i に乗じる。

$$\frac{\partial C_L h}{\partial t} + \frac{\partial C_L h u}{\partial x} + \frac{\partial C_L h v}{\partial y} = \begin{cases} i C_{*L} & (i \geq 0) \\ i C_{*DL} & (i < 0) \end{cases} \quad (3.4)$$

ここに、 C_{*DL} は土石流が新たに堆積する河床堆積層における粗粒分砂礫の充填土砂濃度、 C_{*L} は河床堆積層における粗粒分砂礫の充填土砂濃度、土砂濃度における下添字 $*$ は河床堆積層に関連する量である。

33) q_x, q_y を M, N と記述する文献も多く、本資料における差分式やコードに関する記述では X 方向を M 、Y 方向を N で表すこともある。

34) 図 3.1 に示すように、流速と輸送土砂濃度は水深方向に分布がある。そのため、水深平均した場合、連続式の左辺第 2・3 項の土砂輸送のフラックス項 $\nabla \cdot Ch\mathbf{u}$ に土砂輸送補正係数 $\gamma = (1/A) \int_A (CU)/(\bar{C}\bar{U})dA$ を導入して、 $\nabla \cdot \gamma Ch\mathbf{u}$ と表現する必要がある(芦田ほか, 2008, p.190; 伊藤・江頭, 2004 は粒子濃度の形状係数という)。江頭らの土石流力学理論を用いた輸送土砂濃度分布の検討結果によると、勾配に対して増加する傾向であり、河床勾配 5 度で 0.3 程度、10 度で 0.7 程度の値を示すため、 $\gamma = 1$ に設定すると流出土砂量の誤差が発生することになる(芦田ほか, 2008, p.196-197)。本資料では、中川ほか(2002)の表現にならない土砂輸送補正係数を考慮しない(実質的に 1 と設定)。他を例示すると、江頭・伊藤(2004)や竹林・藤田(2018)では明示されておらず(実質的に 1 と設定)、山崎・江頭(2018)は明示的に 1 と設定している。

3.5.3 細粒分砂礫の連続式

細粒分砂礫の連続式は検査体積における細粒分砂礫の体積 $(1 - C_L)C_F h$ の変化を表現する式であり、細粒分砂礫の土砂濃度 C_F を求めるために用いる。右辺の侵食・堆積速度式は細粒分砂礫の実質土砂量に変換するため、みかけの侵食・堆積速度 i に土砂濃度を乗じる。侵食時には、河床堆積層の土砂が流れに取り込まれるから、河床堆積層における間隙流体を除いた空間 $(1 - C_{*L})$ における河床の細粒分砂礫の充填土砂濃度 C_{*F} を乗じる。一方、堆積過程 ($i < 0$) では、流れから河床に離脱する土砂を扱うため、離脱後の河床堆積層における粗粒分砂礫の土砂濃度を用いた間隙体積 $(1 - C_{*DL})$ に流れ中の細粒分砂礫が拘束されるとした土砂濃度 C_F を乗じる³⁵⁾。

$$\frac{\partial(1 - C_L)C_F h}{\partial t} + \frac{\partial[(1 - C_L)C_F h u]}{\partial x} + \frac{\partial[(1 - C_L)C_F h v]}{\partial v} = \begin{cases} i(1 - C_{*L})C_{*F} & (i \geq 0) \\ i(1 - C_{*DL})C_F & (i < 0) \end{cases} \quad (3.5)$$

ここに、 C_{*F} は河床堆積層における細粒分砂礫の充填土砂濃度³⁶⁾、 $C_{*DL} = (C_* - C_{*F}) / (1 - C_{*F})$ である(中川ほか, 1996)。また、 C_* は河床堆積層における全砂礫の充填土砂濃度である。 C_{*L} は C_{*DL} を初期値として与える³⁷⁾。

3.6 流砂形態に応じた流れの特性

3.6.1 河床せん断応力

河床せん断応力³⁸⁾は、図 3.1 および図 3.7 に示したように、粗粒分砂礫の土砂濃度 C_L により遷移する流砂形態に応じた式を用いる(高橋, 1992)。本項に示すように数値計算におけるアルゴリズムとしてまとめたのは高橋ほか(1999)である。

土石流と掃流状集合流動の土砂濃度の境界値は、土石流の平衡土砂濃度に土石流発生下限勾配を代入して求めた式(芦田ほか, 1978)を基に、 $0.4C_{*L}$ と表現したものである(Takahashi, 1991; 中川ほか, 2001)³⁹⁾。

掃流状集合流動と掃流砂流の土砂濃度の境界値は、図 3.9 の実験結果に示されるように、実験に基づき 0.02 である⁴⁰⁾(高橋ほか, 1996)。

以下、各流砂形態の河床せん断応力の式を示す⁴¹⁾。

- 土石流

35) この物理的な意味を山崎・江頭(2018)を参考に述べる。堆積過程において、粗粒分砂礫が堆積するときには、その間隙に存在する水と細粒分砂礫のうち、細粒分砂礫が間隙に拘束されたまま固相へと変化する。つまり、流れの固相の粗粒分砂礫と液相だった細粒分砂礫が河床堆積層の固相へと変化する(図 3.10)。流れから離脱する土砂の堆積過程に伴う粗粒分砂礫の土砂濃度は不明であるから、 C_{*DL} と表す。また、もともと河床に堆積している河床堆積層はこのような過程で形成されたことを考えれば、 C_{*DL} は粗粒分砂礫の土砂濃度 C_{*L} に近いと考えられる。そこで、 $C_{*DL} = C_{*L}$ と仮定する。なお、高橋(2006, p.180)は、 C_{*DL} の間隙に細粒分砂礫が閉じ込められて堆積するのは C_F が約 26% 以下の場合とし、それ以上では細粒分砂礫の堆積層に粗粒分砂礫が分散して存在する、と述べる。

36) 河床堆積層における細粒分砂礫の土砂濃度 C_{*F} の空間分割の定義は、本資料では中川ほか(2002)と同様に、粗粒分砂礫を除いた空間における体積分率とした。流れの細粒分砂礫の土砂濃度 C_F と同様である(図 3.10 参照)。河床堆積層全体に占める細粒分砂礫の体積分率を C'_{*F} とすると、 $C'_{*F} = (1 - C_{*L})C_{*F}$ の関係が成立する。

37) C_{*DL} の式は、初期河床堆積層厚における全砂礫の充填土砂濃度 $C_* = C_{*L} + (1 - C_{*L})C_{*F}$ を粗粒分砂礫濃度 C_{*L} について解き、これを改めて C_{*DL} と定義し直したものである(中川ほか, 1996)。よって、初期には $C_{*DL} = C_{*L}$ と仮定している。参考として中川ほか(1996)における河床堆積層の土砂濃度の数値例を示すと、 $C_* = 0.65$ 、 $C_{*F} = 0.2$ 、 $C_{*L} = C_{*DL} = 0.56$ である。

38) 付録 A の浅水流方程式の導出から分かるように、深さ方向に微分された応力テンソルを、積分して得られる河床におけるせん断応力である(参照式 A.136)。底面せん断応力、底面摩擦係数などという。

39) 芦田ほか(1978)は、土石流発生下限勾配(式 A.12)を土石流の平衡土砂濃度(式 A.17)に代入して、砂礫が全層に一樣に分散することができる土石流の最低土砂濃度として、 $C_{dc} = (C_L^{(d)} / \kappa) C_* \kappa / (1 + \kappa)$ を得た。ここで、 κ は付録 A に示すように移動堆積層厚と表面水深の比で集合流動として発生する条件を示す定数である。高橋(1982)は、同じ式を $C_* / (1 + \kappa^{-1})$ と表現し、 $\kappa = 0.7$ より $0.412C_*$ を得た。 $0.4C_*$ と表現されるようになったのは高橋・匡(1986)以降であると考えられる。なお、 $\kappa = 1.0$ の方が実際と矛盾しないとすれば(高橋, 2006, p.126)、土石流と掃流状集合流動の境界は $0.5C_*$ となる。表現そのものは、江頭・宮本(2003)に示される式と同じとなる(ただし、物理的な意味は体積土砂濃度と輸送土砂濃度で異なる)。ここまでの議論は細粒分砂礫を考慮せず、河床堆積層の土砂濃度(実質的に粗粒分砂礫のみ)を用いて $0.4C_*$ としたものである(高橋ほか, 1999; 川池ほか, 2000)。細粒分砂礫も扱う場合は河床堆積層における粗粒分砂礫の土砂濃度を用いて $0.4C_{*L}$ とする(中川ほか, 2001, 2002)。

40) 高橋ほか(1999)や中川ほか(2001)、中川ほか(2002)では 0.01、中川ほか(1996)や高橋・里深(1999)、川池ほか(2000)では 0.02 を与えている。

41) ここに示す河床せん断応力は流砂形態の遷移の境界における土砂濃度で不連続となる。これに対して、計算技術的な観点からが解消する手法が提案されている(鈴木ほか, 2013; 江頭ほか, 2014)。本資料では未適用であり本文の説明に従った手法を用いた。なお、江頭ほか(1997)は土石流から掃流砂流まで一貫して連続的に遷移するように理論が作られている。

0.4 $C_{*L} < C_L$ を満たすとき、次の河床せん断応力を用いる⁴²⁾。

$$\frac{\tau_b}{\rho_T} = \frac{1}{8} \left[\frac{\rho_T}{\sigma} \right]^{-1} \left[\left(\frac{C_{*DL}}{C_L} \right)^{1/3} - 1 \right]^{-2} \left(\frac{h}{d_L} \right)^{-2} \| \mathbf{u} \| \mathbf{u} \quad (3.6)$$

• 掃流状集合流動

0.02 $\leq C_L \leq 0.4 C_{*L}$ を満たすとき、次の河床せん断応力を用いる⁴³⁾。

$$\frac{\tau_b}{\rho_T} = \frac{1}{0.49} \left(\frac{h}{d_L} \right)^{-2} \| \mathbf{u} \| \mathbf{u} \quad (3.7)$$

• 掃流砂流

$C_L < 0.02$ を満たすとき、次の河床せん断応力を用いる。

$$\frac{\tau_b}{\rho_T} = \frac{gn^2}{h^{1/3}} \| \mathbf{u} \| \mathbf{u} \quad (3.8)$$

ここに、 n は Manning 粗度係数、 $\| \mathbf{u} \|$ はベクトルのノルム（大きさ）である。

3.6.1.1 表現の整理

次章以降の差分式やコードとのつながり分かるように、上に示した3つの河床せん断応力を、

$$\frac{\tau_b}{\rho_T} = f_r \| \mathbf{u} \| \mathbf{u} \quad (3.9)$$

とまとめて表現する。ここに、 f_r は本資料では流動抵抗係数と呼ぶことにする⁴⁴⁾⁴⁵⁾⁴⁶⁾。

3.6.2 侵食・堆積速度式

流れの土砂濃度が平衡土砂濃度より小さい場合には、河床を侵食して土砂を取り込み、逆に平衡土砂濃度より大きい場合には土砂を河床に堆積させて土砂を失う（付録 A 参照⁴⁷⁾）。

42) $(\rho_T/\sigma)^{-1}$ は $[C_L + (1 - C_L)\rho_m/\sigma]^{-1}$ である。Bagnold (1966) がダイレタンシーを説明する際に導入した線濃度 $\lambda = [(C_{*DL}/C_L)^{1/3} - 1]^{-1}$ (高橋, 1977; Takahashi, 1978) を用いば、この式は、質量密度、線濃度、相対水深、流速により簡潔に表現できる。

43) この式は、高橋ほか (1988a) 及び高橋ほか (1988b) によると、「従来の実験結果から一つの経験則を誘導すれば」得られる実験式である。中川 (1989, p.183) には、高橋 (1982) と橋本ほか (1986) の実験結果から誘導された経験則、と記載され、特段の導出が記述されずに導入される。

44) 水理学において河床せん断応力は $\tau_b = f' \rho u^2/2$ と表現し (Darcy-Weisbach 則)、 f' を摩擦損失係数と呼ぶ。江頭ほか (1989) の土石流理論における河床せん断応力は、 $\tau_b = \tau_y + \rho f_r \| \mathbf{u} \|^2$ と表現され、 τ_y はクーロン型の降伏応力、 f_r は流動抵抗係数である (芦田ほか, 2008, p.194)。 f_r の呼称について、芦田ほか (1985) は流体抵抗係数、高橋ほか (1988a) は流体摩擦係数、宮本・山下 (2003) は抵抗係数、芦田ほか (2008, p.111) は河床の抵抗係数という。

45) 各式に表れる数字は流砂形態に応じた抵抗係数の実験値に基づいている。導出は付録 A に示すが、要点を簡潔にまとめる。土石流と掃流状集合流動の抵抗則は $u/u_* = R(h/d_L)$ と表現でき、流動抵抗係数と $f_r = R^{-2}(h/d_L)^{-2}$ の関係がある。ここに R を抵抗係数と呼ぶことにする。土石流の場合、線濃度 $\lambda = [(C_{*DL}/C_L)^{1/3} - 1]^{-1}$ とすると、抵抗係数 $R = (2/5)[(\rho_T/\sigma)/(a_i \sin \alpha)]^{0.5} \lambda^{-1}$ となり、流動抵抗係数 $f_r = (25/4)a_i \sin \alpha (\rho_T/\sigma)^{-1} \lambda^2 (h/d_L)^{-2}$ (式 A.58) となる。ここで、 $a_i \sin \alpha$ は、高橋 (2004, p.61) に示されるように a_i を Bagnold (1966) による 0.042、衝突角 α を固定床の土石流実験 (高橋・吉田, 1979) から $\tan \alpha = 0.6$ ($\sin \alpha = 0.5145$) とすると、係数 $(25/4)a_i \sin \alpha = 1/7.4$ を得る。中川ほか (1998) 以降、丸めて、 $1/8$ で記述されていると考えられる。一方、掃流状集合流動の場合、実験による抵抗則 $u/u_* = (0.4 \sim 0.7)(h/d_L)$ (高橋, 2004, p.88) (式 A.68) より、抵抗係数 $R = 0.7$ として流動抵抗係数を計算すると係数 $1/0.49$ が得られる。ただし、発表の時系列を考えると、高橋・匡 (1986) において高橋 (1982) や橋本ほか (1986) による理論的研究や実験的検討を参考に簡略化した $R = 0.7h^{3/2}$ が示され (この R は高橋 (1982) に示される R' と次の脚注の方法で $m = 5/2$ として得た値であるが、 $R = 0.7$ と比較すると流動深の 1.5 乗のずれがある。)、高橋・藤井 (1988) の実験式 (式 A.66) から適当な物性値を用いると $R = 0.69$ が示される。また、掃流砂流のときは、Manning の平均流速公式と単位幅流量 $q = hu$ の関係から $q = (1/n)h^{2/3}I^{1/2}$ となり、これを摩擦速度 $u_* = \sqrt{ghI}$ に注意して $u/u_* = R(h/d_L)$ の形式に変形すると $u/u_* = h^{1/6}/(ng^{1/2}) = (d_L/h^{5/6}ng^{1/2})(h/d_L)$ となるため、 $R = d_L/h^{5/6}ng^{1/2}$ であり、流動抵抗係数を計算すると本文の式を得る。ただし、掃流砂流の抵抗則は土石流や掃流状集合流動とは異なり対数則となり相対水深に比例する訳ではないため、Manning 則を用いると形式上の表現を合せられるだけである。

46) 抵抗係数 R について、他の文献を読むための定義を整理する。高橋ほか (1990) や高橋 (2004, p.126)、高橋・中川 (1991)、中川ほか (1996) などでは、1次元のキネマティックウェーブ近似をした等流の流量 (運動方程式) として、 $q = R'h^m I^{1/2}$ (ただし、これらの文献では抵抗係数に C 、流動深に径深 R などの記号を使用) のように流動深の指数を m として一般的に表現する。このとき、抵抗を表す係数 R' と抵抗則の抵抗係数 R との関係を示す。等流の運動方程式を、 $q = u/h$ 、 $u_* = \sqrt{ghI}$ を踏まえて変形すると、 $u/u_* = R'h^{m-3/2}/\sqrt{g}$ である。 $m = 5/2$ であれば R との関係は、 $R = R'd_L/\sqrt{g}$ である。また、 $m = 5/3$ であれば、 $R = R'd_L/(\sqrt{g}h^{5/6})$ である。本来、土石流と掃流状集合流動は $m = 5/2$ で、掃流砂流は Manning 則より $m = 5/3$ が適当である。文献によっては、土石流と掃流状集合流動のときに $m = 5/3$ と示されているため、注意されたい。

47) 土石流の侵食・堆積が土砂濃度によることは高橋 (1977) で言及されるが、侵食速度式と堆積速度式の議論は高橋・匡 (1986) から始まる考えられる。本資料では、侵食速度式は高橋 (2004) と川池ほか (2000) を組合せた式、堆積速度式は中川ほか (2002) を用いた。

- 侵食

$i \geq 0$ は $C_L \leq C_\infty$ の場合に生じ、この時、侵食速度式を用いる⁴⁸⁾。

$$i = \delta_e \frac{C_\infty - C_L}{C_* - C_\infty} \|u\| \quad (3.10)$$

- 堆積

$i < 0$ は $C_\infty < C_L$ の場合に生じ、この時、堆積速度式を用いる⁴⁹⁾。

$$i = \delta_d \frac{C_\infty - C_L}{C_{*DL}} \|u\| \quad (3.11)$$

ここに、 δ_e は侵食速度係数、 δ_d は堆積速度係数である。

C_∞ は平衡土砂濃度であり、流砂形態に応じて次のように示される。

3.6.3 平衡土砂濃度

河床勾配で流砂形態が遷移し、それに伴って平衡土砂濃度は遷移する⁵⁰⁾ (図 3.9)。以下に示す本資料における土石流の式は高橋・匡 (1986)、掃流状集合流動の式は Takahashi (1991)、掃流砂流の式は高橋ほか (1999) によるものである⁵¹⁾。流砂形態の境界勾配の値は高橋ほか (1999) で示されたものであるが、その主たる考え方や導き方は中川ほか (1996) に示される⁵²⁾⁵³⁾。

- 土石流

$0.138 < \tan \theta$ の場合、次の平衡土砂濃度式を用いる。

-
- 48) 侵食速度式の基本的な考え方を示す (詳細は付録 A 参照)。飽和河床堆積層の場合、土石流あるいは掃流状集合流動発生条件となる河床堆積層における外力とせん断抵抗力がつりあう河床堆積層の深さ (集合流動の移動堆積層の厚さ) を a_L (水深の 1 次式 ah の形式に書ける) として、この深さの土砂が流れに取り込まれるまでの遅れ時間を次元解析的に考えて $T_e = (1/\delta_e)(d_L/u)$ (δ_e は侵食速度係数、 u は流下方向の流れの流速) と書けるものとすれば、単位時間あたりの侵食量である侵食速度式は $i = (a_L/T_e) = \delta_e \alpha(hu)/d_L = \delta_e \alpha(q/d_L)$ と書くことができる (高橋・匡, 1986)。これは流速に比例してその直交方向から土砂が取り込まれる連行の表現とみなせる。
- 49) 堆積速度式の基本的な考え方は、飽和時における侵食速度式と同様に、流れの土砂濃度が平衡土砂濃度より大きいときに、その余剰分を堆積量とみなし、それがあがる遅れ時間で堆積する、というものである (詳細は付録 A 参照)。堆積速度式は、遅れ時間の代表長さに粒径を使う場合と水深を使う場合がある。侵食速度式と堆積速度式の遅れ時間 T_e の評価は曖昧な点が残される。侵食は粒径が影響すると想定されるが、堆積は粒径より流動深の影響を考える方が適当という考えがある (高橋, 2004, p.129)。本資料の侵食速度式・堆積速度式は、水深を用いた川池ほか (2000) にならったものである。様々な表現方法や導出は付録 B に掲載した。
- 50) 土石流の数値シミュレーションは高橋・辻本 (1984) に始まり、土石流と掃流状集合流動の平衡土砂濃度を変化させる方法は水山・下東 (1985) に初めて示された、と考えられる。後者の方法は矢澤ほか (1986) や水山ほか (1987) にも示される。これらの文献では、図 3.8 に従い、土石流発生下限勾配 $\tan \theta_1$ で流砂形態を変化させた。中川ほか (1996) では平衡土砂濃度が一致する勾配で流砂形態を変化させるため、より緩勾配まで土石流の平衡土砂濃度で流下する。
- 51) 平衡土砂濃度の導出は付録 A に示す。ここでは概略を述べる。土石流の平衡土砂濃度は、河床堆積層において外力とクーロン摩擦力が一致する面を決定し、それより上が不安定な移動可能な砂礫移動層 (移動堆積層厚ともいう) a_L となり、 a_L が流れの全層に分散して流動する条件を考えると、単一粒径の平衡土砂濃度式 A.17 が導かれる (高橋, 1977)。本文の土石流の平衡土砂濃度 (式 3.12) は、同じ考え方を混合粒径の場合に拡張したもので、土石流の間隙には細粒分砂礫が存在するが、河床堆積層の間隙は細粒分砂礫を浮かべない水と仮定して (高橋, 2004, p.128)、外力と抵抗のつり合いより移動堆積層厚を求め侵食速度が 0 となる条件として導出される。(高橋・匡, 1986)。この土砂濃度は、 $C_\infty = (1/V) \int C dV$ (土砂の体積/その領域の全体積) で定義される”体積”土砂濃度 (土砂”容積”濃度ともいう) である。一方、掃流状集合流動の平衡土砂濃度 (式 3.13) と掃流砂流の平衡土砂濃度 (式 3.14) は、水と土砂の混合物の流れによる断面平均流量 q に対する流砂量式による断面平均流砂量 q_b の比、つまり、 $C_\infty = q_b/q = (\int C u dA)/(\int u dA)$ で定義される輸送土砂濃度である。このように、同じ平衡土砂濃度という名称であっても物理的には意味の異なるものである。実験において水路末端で観測するのは輸送土砂濃度であることから、江頭ほか (2014) は、土石流の研究において、土砂濃度と輸送土砂濃度の区別がなされないまま、不合理な結論に至っているものが少なくない状況を危惧するとともに、特に掃流状集合流動の領域では輸送土砂濃度は体積土砂濃度よりかなり小さくなる、と指摘する。江頭ほかの土石流理論 (江頭ほか, 1997) では、輸送濃度は土石流から掃流砂流まで連続的に変化することが示される。
- 52) 高橋ほか (1999)、川池ほか (2000)、中川ほか (2001, 2002) などにおいて、いずれも掃流状集合流動の $\tan \theta$ の上限値は 0.138、の下限値は 0.03 である。0.138 は、中川ほか (1996) の記述に基づいて計算すれば、次のように導出される。図 3.1 を見ればわかるように、常に式 3.13 < 式 3.12、つまり $6.7C_\infty^2 < C_\infty$ が成立する。 $\tan \theta$ について整理すると、 $\tan \theta < (\sigma - \rho) \tan \phi / (\sigma + 5.7\rho)$ を得る (例えば、鈴木ほか, 2013 や Uchida *et al.*, 2013 で明示)。中川ほか (1996) に従い $\sigma, \rho, \tan \phi$ をそれぞれ 2,650 [kg/m^3], 1,000 [kg/m^3], 0.75 とすると、 $\tan \theta < 0.138$ を得る。一方、掃流状集合流動の下限値 0.03 は実験値である。この土砂濃度を C_{sc} として試みに上と同じ手続によって、中川ほか (1996) に示される物性値を用いて、 $C_{sc} < 6.7C_\infty^2$ を計算してみると、 $\sqrt{C_{sc}/6.7(\sigma/\rho - 1) \tan \phi} / (1 + \sqrt{C_{sc}/6.7}) < \tan \theta$ となる。左辺は $C_{sc} = 0.02, 0.03$ においてそれぞれ約 0.064, 0.078 となる。 $\tan(0.138), \tan(0.06), \tan(0.08)$ は、それぞれ 7.86, 3.43, 4.57 である。一方、高橋 (2004, p.88) は、図 3.9 を示し、 C_∞ (式 3.13) の成立条件を掃流砂流と土石流の平衡土砂濃度 (式 3.12) の間の値と与える勾配の範囲に限られる、と述べ、その範囲を Takahashi (2014, p.90) は、 $4^\circ < \theta < 10^\circ$ と示す。この値は上に示した値と概ね一致する。
- 53) 流砂形態が変化する勾配で平衡土砂濃度が不連続になる。計算技術的な観点から水山ほか (1987) や水山ほか (1988) (掃流状集合流動の平衡土砂濃度は水山, 1980)、鈴木ほか (2013) は、連続にする方法を示した。中谷ほか (2018) は細粒分砂礫が液相化する効果を見込み鈴木ほか (2013) の方法を適用している。本資料ではこれらの方法は適用していない。

$$C_{\infty} = \frac{\rho_m \tan \theta}{(\sigma - \rho_m)(\tan \phi - \tan \theta)} \quad (3.12)$$

ただし、 C_{∞} は $0.9 C_*$ を上限とする⁵⁴⁾。

• 掃流状集合流動

$0.03 < \tan \theta < 0.138$ の場合、次の平衡土砂濃度式を用いる⁵⁵⁾。

$$C_{\infty} = 6.7 \left[\frac{\rho_m \tan \theta}{(\sigma - \rho_m)(\tan \phi - \tan \theta)} \right]^2 \quad (3.13)$$

• 掃流砂流

$\tan \theta < 0.03$ の場合、次の平衡土砂濃度式を用いる⁵⁶⁾。

$$C_{\infty} = \frac{(1 + 5 \tan \theta) \rho_m \tan \theta}{\sigma - \rho_m} \left(1.0 - \bar{\alpha}^2 \frac{\tau_{*c}}{\tau_*} \right) \left(1.0 - \bar{\alpha} \sqrt{\frac{\tau_{*c}}{\tau_*}} \right) \quad (3.14)$$

ここに、 ϕ は砂礫の内部摩擦角、 $\bar{\alpha}$ は係数⁵⁷⁾、 τ_* は無次元掃流力、 τ_{*c} は無次元限界掃流力⁵⁸⁾である。 α 、 τ_* 、 τ_{*c} はそれぞれ

$$\bar{\alpha}^2 = 2 \frac{0.425 - \frac{\sigma}{\rho_m} \frac{\tan \theta}{\sigma / \rho_m - 1}}{1 - \frac{\sigma}{\rho_m} \frac{\tan \theta}{\sigma / \rho_m - 1}} \quad (3.15)$$

$$\tau_* = \frac{u_*^2}{(\sigma / \rho - 1) g d_L} \quad (3.16)$$

$$\tau_{*c} = 0.05 \quad (3.17)$$

である。ただし、 u_* は摩擦速度で

$$u_* = \sqrt{gh \tan \theta} \quad (3.18)$$

と表される。 $\tan \theta$ は水面の最急勾配である水面勾配を用いることとし

$$\tan \theta = \sqrt{\left(\frac{\partial(z_b + h)}{\partial x} \right)^2 + \left(\frac{\partial(z_b + h)}{\partial y} \right)^2} \quad (3.19)$$

である⁵⁹⁾⁶⁰⁾。ここに z_b は河床位である。

54) 高橋 (1977) は土石流先頭部の土砂濃度を計測して土砂濃度の上限値が存在することを示唆し、実験を基に $0.81 C_*$ を示した。同じ実験結果から、高橋 (1980) は上限値を $0.8 \sim 0.9 C_*$ 、高橋・八木 (1983) は $0.9 C_*$ とした。水山ほか (1987) は数値計算に与える境界として $0.81 C_*$ を用いた。

55) ここに示す掃流状集合流動の平衡土砂濃度の式は、Takahashi (1991) あるいは高橋ほか (1991) が初出であると考えられる (中川ほか, 1996)。係数 6.7 は、高橋ほか (1991) には「従来の実験結果」によるとして導入されるが、高橋 (1982) で提案された式から導出されたものと考えられる。付録 A および B に詳細を述べる。

56) 掃流砂流の平衡土砂濃度は、Takahashi (1987) が提案した掃流砂量式 q_b (高橋ほか 1988b; 高橋, 2006, p.233) を Manning 則に基づく流量 q で除した輸送土砂濃度である (中川ほか, 1996)。本文に記した式 3.14 は高橋ほか (1999) による表現である。付録 A に導出の概要を述べる (式 A.70)。

57) この係数は芦田ほか (1978) による停止限界掃流力 (流れで輸送される土砂に作用する力が減少して堆積しはじめる限界の掃流力) と移動限界掃流力の検討で導かれる (詳細は付録 A の式 A.82)。この結果、急勾配河川で適用できる芦田・高橋・水山の掃流砂量式 (芦田ほか, 1978) が得られ、それを修正した Takahashi (1987) が導かれる。

58) 中川ほか (2002) における掃流砂流の平衡土砂濃度では、無次元限界掃流力に、芦田ほか (1978) による $\tau_{*c} = 0.04 \times 10^{1.72 \tan \theta}$ を用いる。この式は、急勾配で適用可能な無次元限界掃流力の芦田ほか (1973) による実験式 $\tau_{*c} = 0.034 \times 10^{0.33(d/h)}$ や、芦田ほか (1975) 及び水山 (1977, p.52) の実験式 $\tau_{*c} = U_{*c}^2 / [(s-1)gd] = 0.034 [\tan \phi \cdot \cos \theta - s / (s-1) \cdot \sin \theta] \times 10^{0.32(d/h)}$ に対して $s = \sigma / \rho = 2.65$ 、 $\tan \phi = 1.0$ を与えて得られる (芦田ほか, 1975)、とされる。ここで、 U_{*c} は限界摩擦速度である。本資料では緩勾配での氾濫に適用することを想定するため、同式の下限值 0.04 と岩垣式の砂礫粒径に対する限界掃流力 0.05 を考慮して一定値 0.05 を与えるものとした。

59) 勾配として、河床勾配を用いる場合 (例えば、高橋ほか, 1999, 中川ほか, 2002)、エネルギー勾配を用いる場合 (例えば、川池ほか, 2000)、水面勾配を用いる場合 (例えば、高橋・中川, 1991; 鈴木ほか, 2013) がある。水面勾配は流速方向の場合と最急勾配の場合がある。本資料では最急勾配の水面勾配と河床勾配を実装した。なお、高濱ほか (2000b) は、エネルギー勾配を河床勾配に変更すると計算が不安定になる場合があることを報告した。木村ほか (2017) における土石流の計算に関する章を分担執筆した中谷氏は、水面勾配を用いることが多く、その理由として変化のスピードがエネルギー勾配より遅いが河床勾配より速く、計算上安定して且つ現象を表現できるため、と述べる。また、水理委員会 (2002) の「例題 2-5 土石流ハイドログラフの計算」では、「等流状態であれば河床勾配で支障はないが、不等流の場合は水面勾配を用いた方が妥当と考えられ、プログラム中では水面勾配を用いている。また、逆勾配が生じた場合にはエネルギー勾配を援用している」。本資料では、エネルギー勾配を用いると不安定であったため、水面勾配を基本とし、河床勾配も選択できるようにした。

60) 本資料のプログラムではコメントを付け替えてコンパイルし直せば河床勾配でも計算できる。河床勾配は河床の最急勾配とす

3.7 河床位方程式（河床堆積層の連続式）

河床位方程式は、河床堆積層に設定した検査体積における質量保存則を記述するものであり、未知変数である標高（河床位）を求められる。粗粒分・細粒分砂礫の河床位方程式を解くことで、河床勾配や河床堆積層の土砂濃度の変化を計算できる式であるが、本資料においては実装していない。

3.7.1 河床位方程式

河床位方程式の右辺における侵食・堆積速度式については、流れの検査体積では河床から流れへ質量が移動する方向を i の正とするため、その等量が失われる河床堆積層の検査体積では $-i$ と表す。 i は間隙水も含む、単位時間・単位面積あたりみかけの土砂量である。

$$\frac{\partial z_b}{\partial t} = -i \quad (3.20)$$

ここに、 z_b は河床位である。

初期河床位を z_{elev} とすると、河床変動量（正なら河床堆積層厚）は、 $z_v = z_b - z_{elev}$ となる。以下の粗粒分および細粒分砂礫の河床位方程式は、河床変動量を未知量に考えることにする。

3.7.2 河床堆積層の粗粒分砂礫の連続式

本式は本資料では実装していないが、流れと河床の土砂移動の考え方や他文献の理解に資するため参考に示す。右辺の侵食・堆積速度式において、侵食過程では河床堆積層から土砂が離脱する量を表したいため、河床堆積層の土砂濃度 C_{*L} を i に乗じて、間隙水を除いた粗粒分砂礫の実質土砂量とする。一方、堆積過程では、同様の理由によって、流れから土砂が離脱して堆積するため C_{*DL} を i に乗じる。

$$\frac{\partial z_{vL}}{\partial t} = \begin{cases} -iC_{*L} & (i \geq 0) \\ -iC_{*DL} & (i < 0) \end{cases} \quad (3.21)$$

ここに、 z_{vL} は粗粒分砂礫の河床位である。

C_{*L} は、固定床標高を z_{dp} とし、標高を単位面積あたりの土砂量と考えると、

$$C_{*L} = \frac{z_{vL}}{z_b - z_{dp}} \quad (3.22)$$

と求められる。

3.7.3 河床堆積層の細粒分砂礫の連続式

本式は本資料では実装していないが、流れと河床の土砂移動の考え方や他文献の理解に資するため参考に示す。粗粒分砂礫の河床位方程式と同様に、右辺の侵食・堆積速度式は細粒分砂礫の実質土砂量に変換するため土砂濃度を乗じる。侵食のときは河床堆積層の土砂が移動するから粗粒分・細粒分砂礫濃度として河床堆積層の充填土砂濃度を用い、堆積のときは流れから河床に離脱する土砂を扱うため粗粒分・細粒分土砂濃度として流れの土砂濃度を用いる⁶¹⁾。

$$\frac{\partial z_{vF}}{\partial t} = \begin{cases} -i(1 - C_{*L})C_{*F} & (i \geq 0) \\ -i(1 - C_{*DL})C_F & (i < 0) \end{cases} \quad (3.23)$$

れば、

$$\tan \theta = \sqrt{\left(\frac{\partial z_b}{\partial x}\right)^2 + \left(\frac{\partial z_b}{\partial x}\right)^2}$$

である。なお、エネルギー勾配は

$$\tan \theta = \frac{||\tau_b||}{\rho_T g h}$$

と表され、流砂形態毎の河床せん断応力によって計算する。本資料では用いていない。

61) 河床位方程式の粗粒分・細粒分の連続式が明示される文献は必ずしも多くはないが、式 3.23 の右辺の特に堆積の場合にいくつかの表現がある。本式は高橋ほか (1991) に従ったもので、粗粒分砂礫の土砂濃度は土石流から離脱する土砂濃度 C_{*DL} とし、粗粒分砂礫の間隙に均質に浮遊する細粒分砂礫は流れと同様の土砂濃度 C_F のまま粗粒分砂礫間に固定されて堆積される、と物理現象を考えたものと推定される。Uchida *et al.* (2013) や西口 (2015) は、流れ及び河床堆積層の細粒分砂礫の土砂濃度の定義が本資料とは異なり、単位体積あたり（あるいは検査堆積あたり）の細粒分砂礫の占める体積の割合、で定義し、流れから細粒分土砂が離脱する土砂濃度 C_{*DF} を新に定め、細粒分砂礫の堆積速度を iC_{*DF} と表現した。

ここに、 z_{vF} は細粒分砂礫の河床堆積層厚である。

C_{*F} は、粗粒分砂礫と同様に、

$$C_{*L} = \frac{z_{vF}}{z_b - z_{dP}} \quad (3.24)$$

と求められる。

3.8 境界条件

3.8.1 流出境界

計算領域外への流出流量は、次に示す長頂堰流れの堰の公式で求める (土木学会水工学委員会, 2019, p.397)。

$$q = Q/B = Ch^{\frac{3}{2}} \quad (3.25)$$

ただし、 q は単位幅流量、 Q は全流量、 B は堰の幅、 h 流動深、 C は流量係数であり、

$$C = 1.642(h/L)^{0.022} \quad (0 < h/L \leq 0.1)$$

である。ここに、 L は縦断方向の堰長である。 $h/L = 0.1$ のとき $C = 1.56$ になるため、流量係数は本資料では 1.6 の一定値とする⁶²⁾。

3.8.2 流入境界

ハイドログラフをある位置 (谷出口など) に設定し、時々刻々と変化する単位幅流量とその地点の流入勾配から、その位置における等流水深を

$$h = [qnI^{\frac{1}{2}}]^{\frac{3}{5}} \quad (3.26)$$

として与える。ここに、 n はマンニング粗度係数、 I は河床勾配である。

3.8.3 水際境界

非常に小さな流動深 ε を閾値として、流動深と比較して、 $h > \varepsilon$ の領域を水際として設定する (岩佐ほか, 1980)。 $h < \varepsilon$ であれば流れが存在しないため、その位置からの流出流量を零とする⁶³⁾。

3.8.4 壁境界

壁境界はスリップ条件⁶⁴⁾とし、壁境界の法線方向の流速は零とする。ただし、本資料では壁境界条件を使う場面がないため実装していない。

62) 計算領域外への流出境界として透過境界を用いる方法もある。

63) この処理は、岩佐ほか (1980) を参考に先端部の扱いが複雑であるために導入したものである。岩佐ほか (1980) が述べるように、本質的に質量保存則を破るが、その誤差は小さければ計算上水量の連続性は保たれると考えてよい。他に発生する誤差について、鈴木 (2013) は土石流の数値シミュレーションで生じる誤差発生要因を分析し、保存則を破る誤差が発生しないように侵食・堆積速度式を制御する方法を提案した。本資料ではこの手法は未適用である。

64) 接する壁と流体の間に相対速度が生じる条件のこと。

第4章 差分方程式

本章では、基礎方程式の離散化および差分方程式について述べる。本章の脚注には式が示されるソースコード名を記す。ソースコード内の記述箇所が分かりにくい場合は行数も示すことにする。

4.1 離散化と計算条件

4.1.1 基礎方程式の離散化に関する一般的事項

基礎方程式は偏微分方程式で記述され、初期条件や境界条件、物性値などのパラメータを与えれば原理的には解ける。ほとんどの偏微分方程式は解析的に解けないため、基礎方程式を離散化し数値計算による近似解によって運動を推定することになる（図4.1）。

離散化とは、大雑把に言えば、基礎方程式における空間と時間に関する微分演算子を、計算機による四則演算可能な代数方程式に変形することである。本資料では離散化の手法として差分法を用いる。基礎方程式を離散化した式を差分方程式あるいは差分式と呼ぶことにする。

ある時刻における全ての位置の差分方程式を解くとは、ひとつひとつの格子を検査体積とみなして連続式や運動方程式を差分化して、全ての格子でその差分方程式に基づいて次の時刻の質量や運動量を求めることを意味している。この手続を繰り返せば、求めた質量や運動量を基にして、さらに次の時刻の質量や運動量を計算できる。このように逐次計算によって時間発展させて運動を推定する。

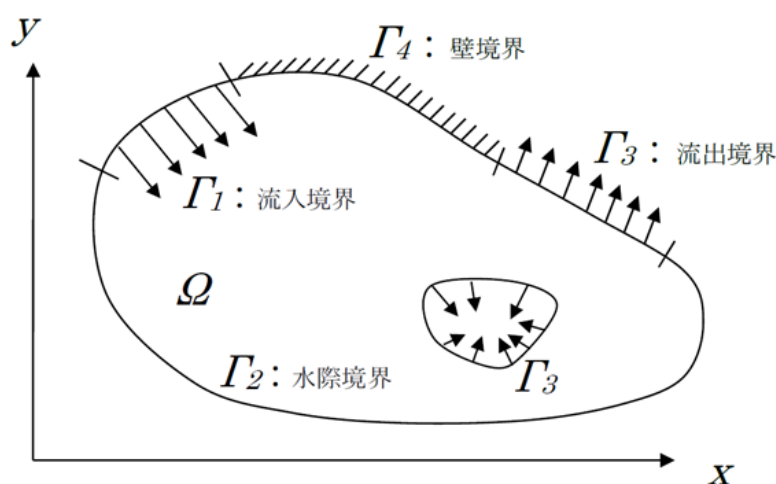


図4.1 計算領域と境界条件の概念図

4.1.2 計算条件

開発目的や様々な制約を踏まえた本資料における土石流氾濫計算プログラムの計算条件は以下の通りである。

- 流砂形態は、土砂濃度や勾配に応じて、土石流、掃流状集合流動、掃流砂流に遷移する。
- 流れによる河床の侵食・堆積過程（河床変動）を表現する。河床は計算領域の全域で移動床とする。固定床（岩床河川や舗装、道路など）は設定できない。
- 土地利用に応じた流れへの抵抗（粗度係数など）は設定できず計算領域の全域で一定とする。
- 流域末端などの点において時系列流量（ハイドログラフ）を流入境界として設定する。ハイドログラフは複数点に与えられる。
- ハイドログラフ以外の方法で計算領域内に水を与えられない（降雨や河川の常時流水などを与えられない）。

- 地形データは矩形格子で表現し、任意の位置座標に対して標高値を1つ有する形式とする。
- 標高は初期条件として地形データを与え、構造物や家屋は地形データとして反映させる。
- 暗渠内の流れ等、地形データとして表現できない流れは計算できない。
- 地盤への浸透流は表現されない。
- ハイドログラフに含まれる粗粒・細粒分砂礫の土砂濃度を設定でき、表面流による河床堆積物侵食型の土砂移動を扱うこともできる。

4.2 本資料における離散化の方法

本資料における離散化の方法は高橋ほか (1986) に示される差分法である。この方法は、

- スタガード格子
- リーフフロッグ法による差分
- 運動方程式の X 方向と Y 方向それぞれに存在する移流項には、計算の安定性と実験結果との整合を図るよう工夫された高橋ほか (1986) の上流差分を使用¹⁾
- 運動方程式における摩擦項 (河床せん断応力の項) には、岩佐ほか (1980) に示される Vasiliev 不安定²⁾を回避するため、流速に陰的解法による差分を使用

と要約される。本章では、基礎方程式とソースコードとのつながりの理解を助けるため、構成の上で次の点を工夫した。

- 式の配置は、基本的に計算進行に沿ってソースコードに出現する順番とした。
- 基礎方程式と差分方程式の類似点と相違点が視覚的に分かるよう差分演算子と平均演算子を導入した。
- ステンシル (ある評価点の計算に必要な周囲の格子点群) の格子点を指す記号は、評価点からの相対位置を表す方位とした。
- 差分式の記号・数式を概ねソースコードと合致させた。

次節から、リーフフロッグ法の特徴、差分・平均演算子の導入、運動方程式の差分式、連続式の差分式、境界条件の差分式を述べる。

4.3 リーフフロッグ法の特徴

岩佐・松尾 (1995) を参考にリーフフロッグ法の特徴を述べる。

リーフフロッグ (leap frog) 法は、時間進行に陽型解法 (explicit scheme)、時間差分、空間差分ともに中央差分を用いる 2 次精度の離散化の方法である。数値的散逸³⁾のない安定性のよい計算方法で計算時間も比較的短い⁴⁾が、その反面で数値分散⁴⁾が大きい欠点を持っている。このため、急激な変化を伴う現象に対しては平滑

- 1) 高橋ほか (1986) の移流項の上流化 (風上化ともいう) について、X 方向の運動方程式の場合、移流項、 $\partial h u u / \partial x (= X D X)$ と $\partial h u v / \partial y (= X D Y)$ のうち $\partial h u v / \partial y (= X D Y)$ については差分式の形が通常の上流化とは異なる (ここでいう「通常」は数値流体力学の教科書等で記される方法である。例えば梶島 (1994) 参照)。Y 方向では $\partial h v u / \partial x (= Y D X)$ が同様である。このことについて高橋ほか (1986) は、円形状であるべき氾濫が三角形になる方向依存の不自然さを取り除き安定した計算ができるスキームを検討して得た差分式と述べる一方、通常の上流化を逸脱した一貫性に欠ける差分スキームとも述べる。
- 2) Vasiliev 不安定とは、藤田・伊藤 (1971) によると、Vasiliev が 1963 年に河川不定流について指摘した差分法における不安定性の条件であり、河床せん断応力の項 (摩擦項)、つまり偏微分方程式の非斉次項の影響で Δt をある値未満に設定しないと増幅率が 1 より小さくできず発散することをいう。その内容を同書を参考に述べる。Chezy の摩擦項 (本資料における掃流砂流の河床せん断応力に相当) を例として、陽型解法の差分を行うと、 $\Delta t \leq R^{2/3} / (g \cdot n \sqrt{I})$ の時間刻みの制約条件が示される。河床せん断応力から与えられるこの条件は、空間格子幅 (Δx) と時間幅 (Δt) の比で与えられる条件 (CFL 条件) と比較して、 Δt に対してより厳しい制約となる場合がある。これを避ける方法として、速度の陰型解法を用いるとよいとし、次の 3 つの方法が紹介されている。1) $u^2 = u^n u^{n+1}$ (Vasiliev)、2) $u^2 = (u^n)^2 + 2u^n u^{n+1}$ (藤田・伊藤 (1971) の第 5 章)、3) $u^2 = \{(u^n)^2 + (u_{n+1})^2\} / 2$ 。3 番目の方法は、Vasiliev と同年に、伊藤ほか (1963) が、常微分方程式や高潮方程式のコリオリ力と海底摩擦力を含んだ式に対して同じ不安定性の指摘し、その解法として提案したものである。岩佐ほか (1980) やそれを基にした高橋ほか (1986) は、伊藤ほか (1963) の方法を適用しており、本資料でも同様である。Vasiliev 不安定に関する文献閲覧の容易さという点でいうと、藤田・伊藤 (1971) より伊藤ほか (1963) が確認しやすい。なお、本脚注の高橋ほか (1986) における方法の解説を逸脱するが、近年、浅水流方程式の湧き出し項による数値解法に伴って生じる実際の物理現象との相違を回避するために種々の新しい数値解法が適用されている (例えば、水野ほか、2016)。
- 3) 基礎方程式を離散化し差分式に書き直すと、2 階空間微分に相当する差分項があらわれる場合がある (移流項の上流化など)。この項によって生じる数値計算上の見かけの拡散のことをいう。数値的散逸が大きい場合、波が減衰しない基礎方程式であっても離散化して数値計算を行うと波が減衰する。
- 4) 周波数が波数と比例関係にない場合、その波は分散性を持つ。波 (情報) は様々な周波数の波の重ね合わせとみなすことができる

化の程度が顕著となるほか、物理現象に無関係な波を生じてしまい、計算精度上問題となることがある。陽型解法とは、 $t = (n + 1)\Delta t$ の解を求める際に、 $t = n\Delta t$ における未知量が明らかな形であるものをいう。直接には解かれていない場合、陰型解法 (implicit scheme) という。

本資料の方法では、図 4.2 に示すように、時間軸の増分が偶数の時 ($n-2, n, n+2$ など) に運動量方程式を解き、奇数の時 ($n-1, n+1, n+3$ など) に連続式を解く。運動方程式と連続式を Δt で食い違いに配置させると、 $2\Delta t$ の時間刻みをひと単位として、基礎方程式一式が解ける。つまり、運動方程式と連続式の時間微分項の差分 (時間積分ともいう) をそれぞれ $2\Delta t$ の間隔で差分する。一方、空間差分も中央差分であるが、スタガード格子の場合、次節に示すように隣合う格子点までが半格子幅となるため、数式としては 1 格子幅の差分 (前進差分) と同じになる。

図 4.2 を見ると、 $n+3$ の連続式を解く際に必要となる評価点に、多数の格子点から線が繋がっている。このように、ある格子点の差分式を評価する際に必要な格子点群をステンシルという。ただし本資料では、同じ時間ステップ⁵⁾における空間の評価点群のみをステンシルと呼ぶことにし、時間方向のステンシル内の格子点は時間平均演算子で示す。

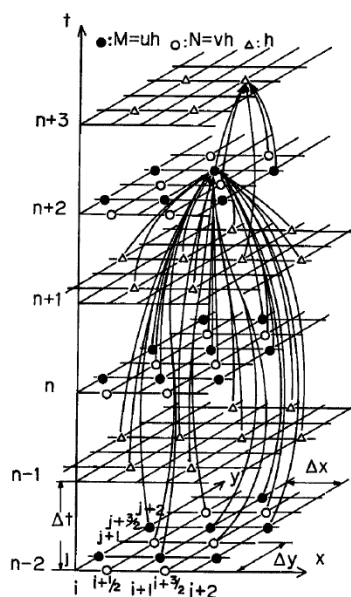


図 4.2 時間発展とステンシルの模式図 (高橋ほか, 1986)

4.4 スタガード格子と格子点・格子辺, 評価点

4.4.1 格子の変数配置と格子点・格子辺

スタガード格子を用いた差分法の場合、図 4.3 に示すように、格子の中央にスカラーを記憶するための点 (図の丸)、格子の辺にベクトルを記憶するための点 (図の X 方向と Y 方向の三角) を配置する。本資料では、それぞれを格子点、格子辺と呼ぶことにする。このようにスカラーとベクトルの各方向の離散点が異なって配置される格子をスタガード (staggered; 食い違い) 格子という。

1 つの格子を検査体積と考える (有限体積法の考え方を援用)。格子点は検査体積内部の変化量を表す点、格子辺は検査体積の表面 (辺) を通過する流束⁶⁾を表す点とみなすことができる⁷⁾。

計算する空間全体に格子を張れば、図 4.4 に示すように格子点と格子辺が配置される。

が、分散性があると周波数に応じて波の形が変化する。差分によって生じる見かけの分散を数値分散という。

5) 本資料では、 Δt や $2\Delta t$ を「時間刻み」、時刻を「時間ステップ」と呼ぶ。

6) フラックス; 単位時間・単位面積 (2次元なら単位幅) あたり通過する物理量のこと

7) 差分法では通常、格子点は数値が与えられる離散的な空間位置とみなし、格子間には何も値が存在しない。有限体積法では検査体積内には一定値が存在し、有限要素法では節点間は補間関数による値が存在するものとみなす。

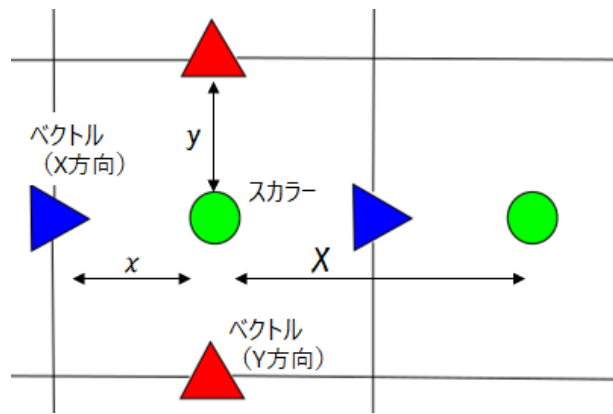


図 4.3 スタガード格子における格子点と格子辺の配置

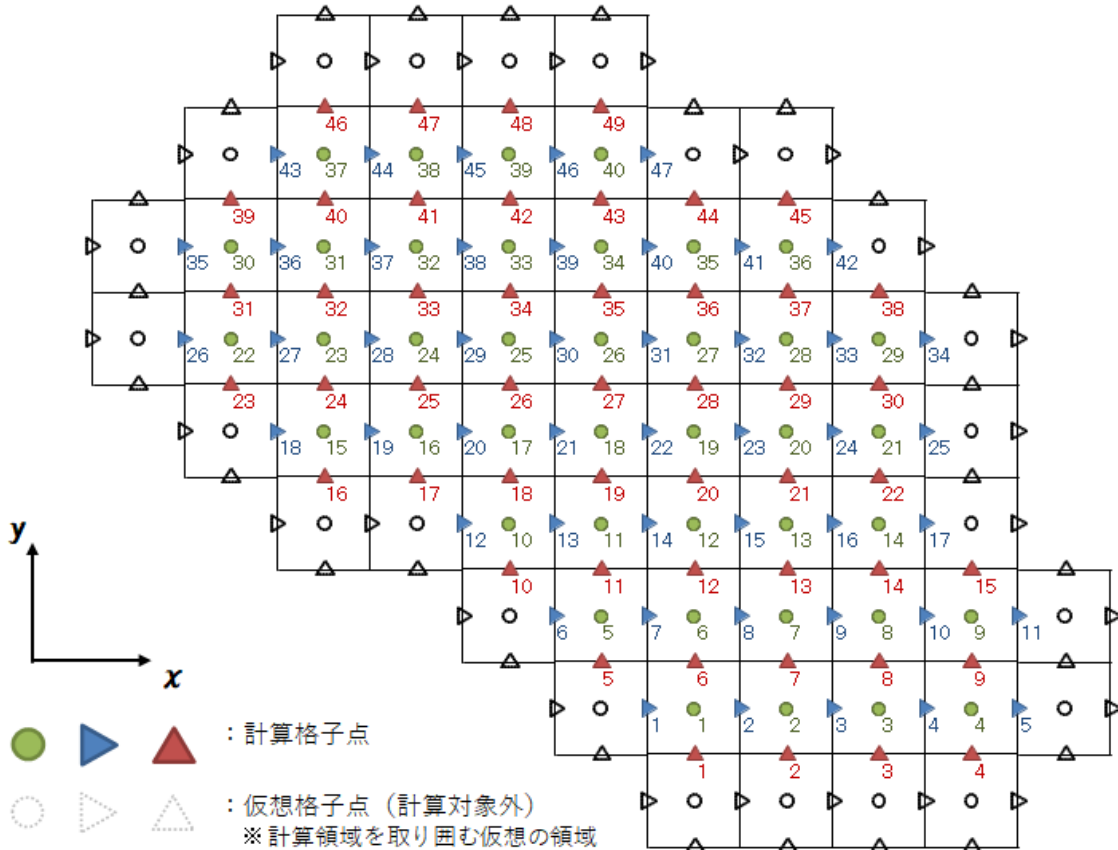


図 4.4 計算領域における格子点の分布イメージ

4.4.2 基礎方程式の評価点

運動方程式により流量 q_x, q_y を求められる。流量はベクトルであるため、格子辺に配置される。このため、運動方程式の差分方程式を解く際には、ある格子辺を中心としてその周辺の格子点及び格子辺（ステンシル）を用いて評価する。本資料では、運動方程式の差分方程式の評価の際に着目する一つの格子辺を、ベクトル評価点と呼ぶことにする。

一方、連続式により、 h, z 、などのスカラーを求められる。スカラーは格子点に配置される。このため、連続式の差分方程式を解く際には、ある格子点を中心としてその周辺の格子点及び格子辺（ステンシル）を用いて評価する。本資料では、連続式や河床位方程式の差分方程式の評価の際に着目する一つの格子点を、スカラー評価点と呼ぶことにする。

4.4.3 ステンシルにおける評価点と格子点の名称

本資料では、差分方程式を解く際に着目する評価点は、高橋ほか (2018) に従って、スカラーとベクトルに関わらず共通して P の記号を用いる。差分方程式を見る際に、スカラーとベクトル (X 成分または Y 成分) で評価点 P の位置が異なることに注意する必要がある。

ステンシルにおける格子点の位置は、 P に対する相対的な位置を方位を用いて表す。 P に対して、1 格子分離れた格子点を大文字 E, W, N, S 、半格子離れた格子点を小文字 e, w, n, s などで示す。この命名法は、有限体積法での慣習に従ったものである (例えば、パタンカー, 1985)。例として、図 4.5 にスカラー評価点を選択した場合を示す。

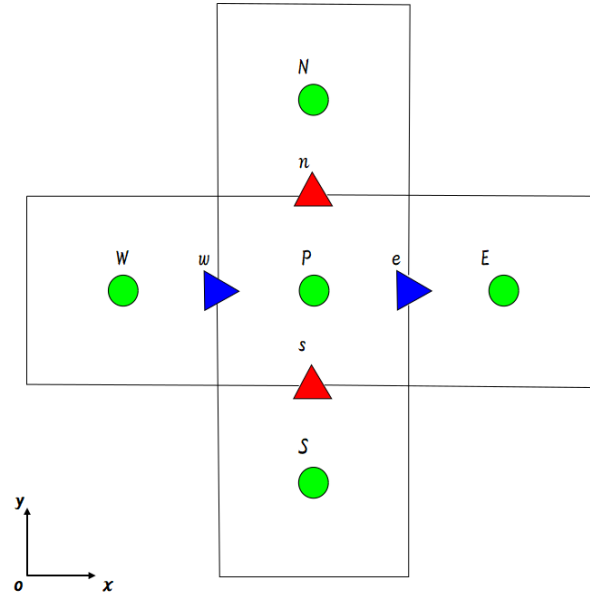


図 4.5 スカラー評価点のステンシル

4.4.4 添字のルール

物理量を表す記号において、空間位置は丸括弧で囲んだ下添字、時間ステップは上添字で示す。時間ステップの上添字は、混乱しない範囲で省略する。

4.5 演算子

本節では、梶島 (1994, 2014) を参考に差分演算子と平均演算子を示す。差分演算子と平均演算子を用いることで、差分方程式を簡潔に表現でき、基礎方程式やソースコードと対照しやすくなると考えている。

4.5.1 差分演算子

差分演算子の下添字は差分の方向を表す。小文字と大文字の下添字は、評価点 P に対して、それぞれ半格子幅の 2 点、1 格子幅の 2 点を表す⁸⁾。

$$\delta_x[A_{(P)}] = \frac{A_{(e)} - A_{(w)}}{\Delta x} \quad (4.1)$$

$$\delta_y[A_{(P)}] = \frac{A_{(n)} - A_{(s)}}{\Delta y} \quad (4.2)$$

$$\delta_X[A_{(P)}] = \frac{A_{(E)} - A_{(W)}}{2\Delta x} \quad (4.3)$$

$$\delta_Y[A_{(P)}] = \frac{A_{(N)} - A_{(S)}}{2\Delta y} \quad (4.4)$$

8) 下添字が大文字の場合は、梶島 (1994) における δ' に相当する。

4.5.2 平均演算子

平均演算子の添字は平均の方向を表す。はじめの3つは空間平均演算子である。3つ目の xy は周囲4方向の格子点平均を表す。残り2つは時間平均演算子である。上添字 n は時間ステップ数を表す整数であり、運動方程式を評価する場合は $\kappa = n$ (偶数)、連続式を評価する場合は $\kappa = n + 1$ (奇数) とする (図 4.2 参照)。

$$\overline{A_{(P)}}^x = \frac{A_{(e)} + A_{(w)}}{2} \quad (4.5)$$

$$\overline{A_{(P)}}^y = \frac{A_{(n)} + A_{(s)}}{2} \quad (4.6)$$

$$\overline{A_{(P)}}^{xy} = \frac{A_{(ne)} + A_{(nw)} + A_{(sw)} + A_{(se)}}{4} \quad (4.7)$$

$$\overline{A_{(P)}}^t = \frac{A_{(P)}^\kappa + A_{(P)}^{\kappa-2}}{2} \quad (4.8)$$

$$\overline{A_{(P)}}^{+t} = \frac{A_{(P)}^{\kappa+2} + A_{(P)}^\kappa}{2} \quad (4.9)$$

4.6 運動方程式

ベクトル評価点 P の位置は X 成分と Y 成分で異なる (図 4.3)。そのため、ステンシルが異なることになり、基礎方程式では方向に依存しない量 (例えば、流速のノルム (速さ)) であっても、差分方程式が異なることに注意する必要がある。

4.6.1 X 成分

評価点 P とステンシルを図 4.6 に示す。 $t = (n + 2)\Delta t$ における流量 M^{n+2} を、この時間ステップにおける既知の量である $t = n\Delta t$ のベクトル、 $t = (n + 1)\Delta t$ のスカラーを用いて求める。

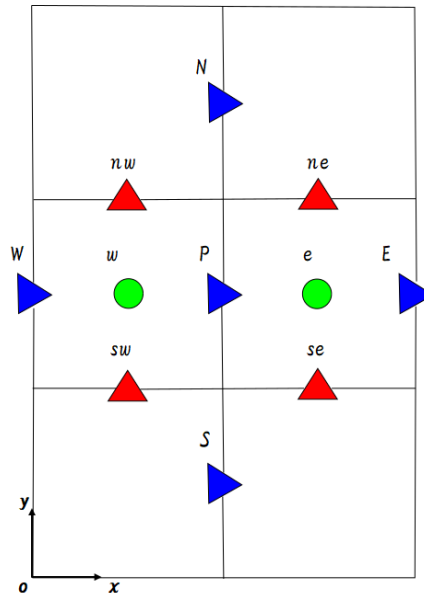


図 4.6 X 方向のベクトル評価点を求める際のステンシル

運動方程式を離散化すると、

$$\frac{M_{(P)}^{n+2} - M_{(P)}^n}{2\Delta t} + \beta \cdot XDX + \beta \cdot XDY = -\overline{gh_{(P)}}^x \delta_x [(h+z)_{(P)}] - f_r \hat{u}_{(P)} \|U_{(P)}\| \quad (4.10)$$

となる。ここに、 XDX , XDY は、慣性（移流）項 $\partial(uM)/\partial x$, $\partial(vM)/\partial y$ を離散化した代数式をそれぞれ表す。高橋ほか（1986）による移流項の上流差分は、一般的な上流化の差分法と比べて場合分けが多い。本資料のソースコードに差分式がそのまま表現されているため⁹⁾、ここでは示さない。詳細は、添字とステンシルの方位記号の違いを図 4.6 で確かめながら、ソースコードあるいは元の論文を確認されたい。また、

$$\hat{u}_{(P)} = \overline{M_{(P)}}^{+t} / \overline{h_{(P)}}^x = \frac{[M_{(P)}^{n+2} + M_{(P)}^n]/2}{\overline{h_{(P)}}^x} \quad (4.11)$$

$$\|U_{(P)}\| = \sqrt{\hat{u}_{(P)}^2 + \overline{v_{(P)}}^{xy}} \quad (4.12)$$

であり流量 M^{n+2} と陰的に時間平均するのは Vasiliev 不安定回避のためである。

$M_{(P)}^{n+2}$ について解くと、

$$M_{(P)}^{n+2} = \frac{\left[1 - \Delta t \frac{f_r \|U_{(P)}\|}{\overline{h_{(P)}}^x}\right] M_{(P)}^n - 2\Delta t [\beta \cdot XDX + \beta \cdot XDY + g\overline{h_{(P)}}^x \delta_x [(h+z)_{(P)}]]}{1 + \Delta t \frac{f_r \|U_{(P)}\|}{\overline{h_{(P)}}^x}} \quad (4.13)$$

を得る。

流動抵抗係数 f_r は次に示すように流砂形態に応じて変化する。ただし、表現をソースコードに合わせるため、 $\rho_T/\sigma = C_L + (1 - C_L)\rho_m/\sigma$ の関係を用いる。

- 土石流： $0.4 C_{*L} < \overline{C_{L(P)}}^x$ ¹⁰⁾

$$f_r = \frac{1}{8} \left[\overline{C_{L(P)}}^x + (1 - \overline{C_{L(P)}}^x) \frac{\overline{\rho_{m(P)}}^x}{\sigma} \right]^{-1} \left[\left(\frac{C_{*DL}}{\overline{C_{L(P)}}^x} \right)^{1/3} - 1 \right]^{-2} \left(\frac{\overline{h_{(P)}}^x}{d_L} \right)^{-2} \quad (4.14)$$

- 掃流状集合流動： $0.02 \leq \overline{C_{L(P)}}^x \leq 0.4 C_{*L}$

$$f_r = \frac{1}{0.49} \left(\frac{\overline{h_{(P)}}^x}{d_L} \right)^{-2} \quad (4.15)$$

- 掃流砂流¹¹⁾： $\overline{C_{L(P)}}^x < 0.02$

$$f_r = \frac{gn^2}{(\overline{h_{(P)}}^x)^{1/3}} \quad (4.16)$$

さらに、ソースコードと表現を合わせるために、

$$\begin{aligned} fr_{q1} &= 1 + \Delta t \frac{f_r \|U_{(P)}\|}{\overline{h_{(P)}}^x} \\ fr_{q2} &= \left[1 - \Delta t \frac{f_r \|U_{(P)}\|}{\overline{h_{(P)}}^x} \right] \end{aligned} \quad (4.17)$$

とすると、運動方程式の差分方程式として

$$M_{(P)}^{n+2} = \frac{fr_{q2} M_{(P)}^n - 2\Delta t [\beta \cdot XDX + \beta \cdot XDY + g\overline{h_{(P)}}^x \delta_x [(h+z)_{(P)}]]}{fr_{q1}} \quad (4.18)$$

を得る¹²⁾。

4.6.2 Y 成分

評価点 P とステンシルを図 4.7 に示す。 $t = (n+2)\Delta t$ における流量 N^{n+2} を、この時間ステップにおける既知の量である、 $t = n\Delta t$ のベクトル、 $t = (n+1)\Delta t$ のスカラーを用いて求める。

9) 式 4.10 に対応するソースコードは、`qmflux.f90` 1.40-130 に記される。

10) 土石流形態の下限土砂濃度を $C_{dc} = C_c^{(d)} = 0.4C_{*L} (= C_{*L}/(1 + \kappa^{-1}))$ の関係があるため、ソースコードでは `cdc` の変数が用いられる。Y 方向の河床せん断応力の計算における部分も同様である。

11) 掃流状集合流動と掃流砂流の境界の平衡土砂濃度 $C_{sc} = C_c^{(s)}$ は、前章に示したように実験に基づく値である。0.02 が多いもの（高橋ほか、1996）、論文によって若干の違いがある。本資料のプログラムでは、 $C_{sc} = C_c^{(s)}$ を表す変数 `csc` の値をパラメータの入力ファイル (`param.dat`) によってユーザが変更できる。

12) `qmflux.f90` 土石流：1.157-163, 掃流状集合流動：1.151-155, 掃流砂流：1.144-149

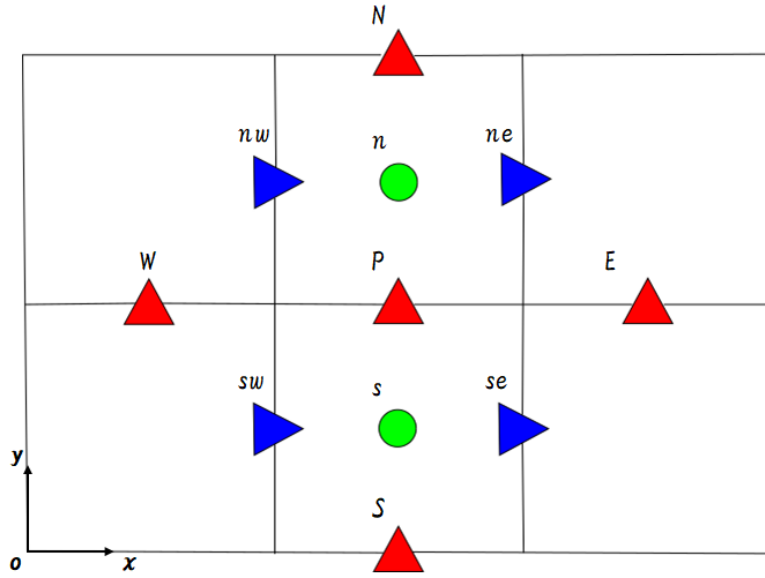


図 4.7 Y 方向のベクトル評価点を求める際のステンシル

運動方程式を離散化すると,

$$\frac{N_{(P)}^{n+2} - N_{(P)}^n}{2\Delta t} + \beta \cdot YDX + \beta \cdot YDY = -\overline{gh_{(P)}^y} \delta_y [(h+z)_{(P)}] - f_r \hat{v}_{(P)} \|V_{(P)}\| \quad (4.19)$$

となる。ここに, YDX, YDY は, 慣性 (移流) 項 $\partial(uN)/\partial x, \partial(vN)/\partial y$ を離散化した代数式をそれぞれ表す。高橋ほか (1986) による移流項の上流差分は, 一般的な上流化の差分法と比べて場合分けが多い。本資料のソースコードに差分式がそのまま表現されているため¹³⁾, ここでは示さない。詳細は, 添字とステンシルの方位記号の違いを図 4.7 で確かめながら, ソースコードあるいは元の論文を確認されたい。また,

$$\hat{v}_{(P)} = \overline{N_{(P)}^{+t}} / \overline{h_{(P)}^y} = \frac{[N_{(P)}^{n+2} + N_{(P)}^n]/2}{\overline{h_{(P)}^y}} \quad (4.20)$$

$$\|V_{(P)}\| = \sqrt{\overline{u_{(P)}^{xy}} + \hat{v}_{(P)}^2} \quad (4.21)$$

であり, 流量 $N_{(P)}^{n+2}$ と陰的に時間平均するのは, Vasiliev 不安定回避のためである。

$N_{(P)}^{n+2}$ について解くと,

$$N_{(P)}^{n+2} = \frac{\left[1 - \Delta t \frac{f_r \|V_{(P)}\|}{\overline{h_{(P)}^y}}\right] N_{(P)}^n - 2\Delta t \left[\beta \cdot YDX + \beta \cdot YDY + \overline{gh_{(P)}^y} \delta_y [(h+z)_{(P)}]\right]}{1 + \Delta t \frac{f_r \|V_{(P)}\|}{\overline{h_{(P)}^y}}} \quad (4.22)$$

を得る。

流動抵抗係数 f_r は次に示すように流砂形態に応じて変化する。ただし, 表現をソースコードと合わせるため, $\rho_T/\sigma = C_L + (1 - C_L)\rho_m/\sigma$ の関係を用いる。

- 土石流: $0.4 C_{*L} < \overline{C_{L(P)}^y}$

$$f_r = \frac{1}{8} \left[\overline{C_{L(P)}^y} + (1 - \overline{C_{L(P)}^y}) \frac{\overline{\rho_{m(P)}^y}}{\sigma} \right]^{-1} \left[\left(\frac{C_{*L}}{\overline{C_{L(P)}^y}} \right)^{1/3} - 1 \right]^{-2} \left(\frac{\overline{h_{(P)}^y}}{d_L} \right)^{-2} \quad (4.23)$$

- 掃流状集合流動: $0.02 \leq \overline{C_{L(P)}^y} \leq 0.4 C_{*L}$

$$f_r = \frac{1}{0.49} \left(\frac{\overline{h_{(P)}^y}}{d_L} \right)^{-2} \quad (4.24)$$

- 掃流砂流: $\overline{C_{L(P)}^y} < 0.02$

$$f_r = \frac{gn^2}{(\overline{h_{(P)}^y})^{1/3}} \quad (4.25)$$

13) qnflux.f90 1.41-131

さらに、ソースコードと表現を合わせるために、

$$\begin{aligned} fr_{q1} &= 1 + \Delta t \frac{f_r \|V_{(P)}\|}{\overline{h_{(P)}}^y} \\ fr_{q2} &= \left[1 - \Delta t \frac{f_r \|V_{(P)}\|}{\overline{h_{(P)}}^y} \right] \end{aligned} \quad (4.26)$$

とすると、運動方程式の差分方程式として

$$N_{(P)}^{n+2} = \frac{fr_{q2} N_{(P)}^n - 2\Delta t [\beta \cdot YDX + \beta \cdot YDY + g\overline{h_{(P)}}^y \delta_y [(h+z)_{(P)}]]}{fr_{q1}} \quad (4.27)$$

を得る¹⁴⁾。

4.7 連続式

評価点 P とステンシルを図 4.8 に示す。運動方程式で流量を得た後、時間ステップを Δt 進めて、連続式を解く。連続式では $t = (n+3)\Delta t$ におけるスカラー量 h^{n+3} , C_L^{n+3} , C_F^{n+3} , z^{n+3} , C_{*L}^{n+3} , C_{*F}^{n+3} を、この時間ステップにおける既知の量である $t = (n+2)\Delta t$ のベクトル, $t = (n+1)\Delta t$ のスカラーを用いて求める。

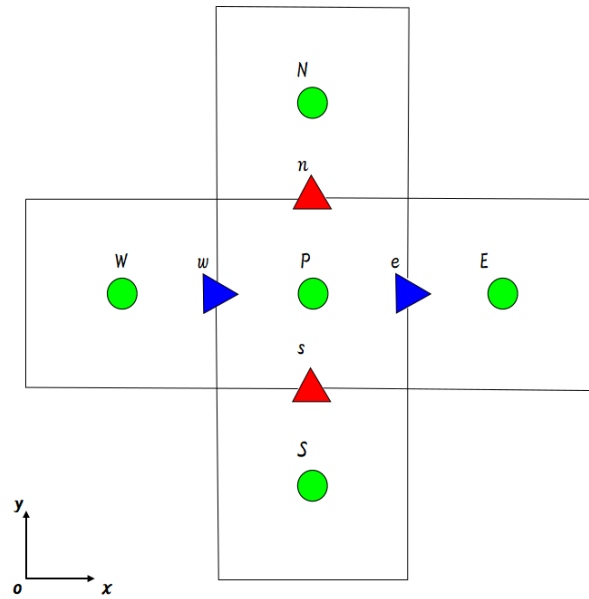


図 4.8 スカラー評価点のステンシル (図 4.5 を再掲)

4.7.1 流れ全体の連続式

流れ全体の連続式を離散化すると、

$$\frac{h_{(P)}^{n+3} - h_{(P)}^{n+1}}{2\Delta t} + \delta_x [M_{(P)}^{n+2}] + \delta_y [N_{(P)}^{n+2}] = i_{(P)}^{n+1} \quad (4.28)$$

となる。 $h_{(P)}^{n+3}$ について解くと、

$$h_{(P)}^{n+3} = h_{(P)}^{n+1} + 2\Delta t \{-\delta_x [M_{(P)}^{n+2}] - \delta_y [N_{(P)}^{n+2}] + i_{(P)}^{n+1}\} \quad (4.29)$$

を得る¹⁵⁾。

右辺の侵食・堆積速度式 $i_{(P)}^{n+1}$ を求めるには平衡土砂濃度 C_∞ が必要である。その平衡土砂濃度を求めるには勾配が必要である。本資料では勾配には最急勾配の水面勾配を用いた。(ただし、ソースコードをコメントすることで河床勾配に変更可能) なお、プログラムにおいて、侵食・堆積速度式は、流れの連続式と同じ時間ステップの値を使用するため、砂礫の連続式と河床位方程式を解いた後に、次の時刻に備えて計算する。

14) qnflux.f90 土石流: 1.158-164, 掃流状集合流動: 1.152-156, 掃流砂流: 1.145-150

15) hdepth.f90

4.7.1.1 勾配

勾配はソースコードに合わせて $S_{(P)}$ と表現する¹⁶⁾ (前章では $\tan \theta$)。

水面勾配 水面勾配は、最急勾配として

$$S_{(P)} = \left(\delta_X(z_{(P)} + h_{(P)}) + \delta_Y(z_{(P)} + h_{(P)}) \right)^{1/2} \quad (4.30)$$

と計算する¹⁷⁾。

河床勾配 河床勾配は、最急勾配として

$$S_{(P)} = \left(\delta_X z_{(P)}^{n+1} + \delta_Y z_{(P)}^{n+1} \right)^{1/2} \quad (4.31)$$

と計算する¹⁸⁾。

4.7.1.2 平衡土砂濃度

ソースコード¹⁹⁾と表現を合わせるために、砂礫の比重を $s = \sigma / \rho_m$, 水中重量 γ_s (例えば, 国土技術政策総合研究所, 2016, p.6) を

$$\gamma_s = \frac{\sigma - \rho_m(P)}{\rho_m(P)} = \frac{\sigma}{\rho_m(P)} - 1 = s - 1 \quad (4.32)$$

と表す²⁰⁾。これを用いると、流砂形態に応じた平衡土砂濃度は次のように表される。

- 土石流: $0.138 < S_{(P)}$

$$C_{\infty(P)} = \frac{S_{(P)}}{\gamma_s(\tan \phi - S_{(P)})} \quad (4.33)$$

ただし, $C_{\infty(P)}$ は $0.9C_*$ を上限とする。

- 掃流状集合流動: $0.03 < S_{(P)} < 0.138$

$$C_{\infty(P)} = 6.7 \left[\frac{S_{(P)}}{\gamma_s(\tan \phi - S_{(P)})} \right]^2 \quad (4.34)$$

- 掃流砂流: $S_{(P)} < 0.03$

$$C_{\infty(P)} = \frac{(1 + 5S_{(P)})S_{(P)}}{\gamma_s} \left(1.0 - \bar{\alpha}^2 \frac{\tau_{*c}}{\tau_*} \right) \left(1.0 - \bar{\alpha} \sqrt{\frac{\tau_{*c}}{\tau_*}} \right) \quad (4.35)$$

ここに, $\bar{\alpha}$, τ_* , τ_{*c} はそれぞれ次式で示される。

$$\bar{\alpha}^2 = \frac{2(0.425 - \frac{(s+1)S_{(P)}}{s})}{1 - \frac{(s+1)S_{(P)}}{s}} \quad (4.36)$$

$$\tau_* = \frac{u_*^2}{sgd_L} \quad (4.37)$$

$$\tau_{*c} = 0.05 \quad (4.38)$$

また, u_*^2 は次式で示される。

$$u_*^2 = gh_{(P)}^{n+3} S_{(P)} \quad (4.39)$$

16) プログラム内では、プログラム作成時にエネルギー勾配で計算しようとした名残で、エネルギー勾配を意味する $S_{ei(P)}$ にちなみ sei を用いている。

17) `gradws.f90`

18) `gradbed.f90`

19) `cinfty.f90`

20) 水中重量を s とする文献もある (例えば, 土木学会水工学委員会, 2019, p.283)。他の文献と比較する際に注意を要する。

4.7.1.3 侵食・堆積速度式

侵食・堆積速度式は、上で求めた平衡土砂濃度を次の式に代入して計算する²¹⁾。

- 侵食： $i \geq 0; C_{L(P)} \leq C_\infty$

$$i_{(P)} = \delta_e \frac{C_{\infty(P)} - C_{L(P)}}{C_* - C_{\infty(P)}} U \quad (4.40)$$

- 堆積： $i < 0; C_\infty < C_{L(P)}$

$$i_{(P)} = \delta_d \frac{C_{\infty(P)} - C_{L(P)}}{C_{*DL}} U \quad (4.41)$$

ただし、 δ_e と δ_d はそれぞれ侵食速度係数、堆積速度係数である（ここでは差分演算子ではないことに注意）。

4.7.2 粗粒分砂礫の連続式

水と土砂の混合物の流れにおける粗粒分砂礫の連続式を、ソースコードの表現に合わせて、高橋ほか (1991) の表現に直す。単位面積あたりの粗粒分砂礫の体積 $V_L = C_L h$ (V と表記されるが流動深 h のうち粗粒分砂礫を表すことに注意)、 $X \cdot Y$ 成分の単位幅当たりの粗粒分砂礫の単位幅流量 $q_{bLx} = C_L M$, $q_{bLy} = C_L N$ とすると、

$$\frac{\partial V_L}{\partial t} + \frac{\partial q_{bLx}}{\partial x} + \frac{\partial q_{bLy}}{\partial y} = \begin{cases} i C_{*L} & (i \geq 0) \\ i C_{*DL} & (i < 0) \end{cases} \quad (4.42)$$

となる。これを離散化すると、

- 侵食： $i \geq 0$

$$V_{L(P)}^{n+3} = V_{L(P)}^{n+1} + 2\Delta t \{-\delta_x [q_{bLx(P)}^{n+2}] - \delta_y [q_{bLy(P)}^{n+2}] + i_{(P)} C_{*L(P)}\} \quad (4.43)$$

- 堆積： $i < 0$

$$V_{L(P)}^{n+3} = V_{L(P)}^{n+1} + 2\Delta t \{-\delta_x [q_{bLx(P)}^{n+2}] - \delta_y [q_{bLy(P)}^{n+2}] + i_{(P)} C_{*DL(P)}\} \quad (4.44)$$

を得る。ただし、

$$\delta_x [q_{bLx(P)}^{n+2}] = C_{L(P)} \delta_x [M_{(P)}^{n+2}] \quad (4.45)$$

$$\delta_y [q_{bLy(P)}^{n+2}] = C_{L(P)} \delta_y [N_{(P)}^{n+2}] \quad (4.46)$$

である。

ここより、流れの粗粒分砂礫の土砂濃度は、

$$C_{L(P)}^{n+3} = V_{L(P)}^{n+3} / h_{(P)}^{n+3} \quad (4.47)$$

と求められる²²⁾。

4.7.3 細粒分砂礫の連続式

水と土砂の混合物の流れにおける細粒分砂礫の連続式を、ソースコードの表現に合わせて、高橋ほか (1991) の表現に直す。単位面積あたりの細粒分砂礫の体積 $V_F = (1 - C_L) C_F h$ (V と表記されるが流動深 h のうち細粒分砂礫成分であることに注意)、 $X \cdot Y$ 成分の単位幅当たりの細粒分砂礫の単位幅流量 $q_{bFx} = (1 - C_L) C_F M$, $q_{bFy} = (1 - C_L) C_F N$ を用いると、

$$\frac{\partial V_F}{\partial t} + \frac{\partial q_{bFx}}{\partial x} + \frac{\partial q_{bFy}}{\partial y} = \begin{cases} i(1 - C_{*L}) C_{*F} & (i \geq 0) \\ i(1 - C_{*DL}) C_F & (i < 0) \end{cases} \quad (4.48)$$

となる。これを離散化すると、

- 侵食： $i \geq 0$

$$V_{F(P)}^{n+3} = V_{F(P)}^{n+1} + 2\Delta t \{-\delta_x [q_{bFx(P)}^{n+2}] - \delta_y [q_{bFy(P)}^{n+2}] + i_{(P)} (1 - C_{*L(P)}) C_{*F(P)}\} \quad (4.49)$$

21) calaii.f90

22) calclcf.f90

- 堆積： $i < 0$

$$V_{F(P)}^{n+3} = V_{F(P)}^{n+1} + 2\Delta t \{ -\delta_x [q_{bFx(P)}^{n+2}] - \delta_y [q_{bFy(P)}^{n+2}] + i_{(P)}(1 - C_{*DL(P)})C_{F(P)} \} \quad (4.50)$$

を得る。ただし、

$$\delta_x [q_{bFx(P)}^{n+2}] = (1 - C_{L(P)})C_{F(P)}\delta_x [M^{n+2}] \quad (4.51)$$

$$\delta_y [q_{bFy(P)}^{n+2}] = (1 - C_{L(P)})C_{F(P)}\delta_x [N^{n+2}] \quad (4.52)$$

である。

これより、流れの細粒分砂礫の土砂濃度は、

$$C_{F(P)}^{n+3} = \frac{V_{F(P)}^{n+3}}{h_{(P)}^{n+3}(1 - C_{L(P)}^{n+3})} \quad (4.53)$$

と求められる²³⁾。

4.7.3.1 質量密度の更新

$t = (n + 3)\Delta t$ において、粗粒分砂礫と細粒分砂礫の連続式により得られた粗粒分砂礫と細粒分砂礫の土砂濃度を用いて、間隙流体と流れ全体の質量密度を次のように更新する²⁴⁾。

$$\rho_{m(P)} = C_{F(P)}^{n+3}\sigma + (1 - C_{F(P)}^{n+3})\rho \quad (4.54)$$

$$\rho_{T(P)} = C_{L(P)}^{n+3}\sigma + (1 - C_{L(P)}^{n+3})\rho_{m(P)} \quad (4.55)$$

4.7.4 河床位方程式

河床位方程式（河床位の連続式）により、侵食・堆積速度に応じた標高の変化を得られるため、河床変動を表現できる。

4.7.4.1 河床位方程式

河床位方程式の差分式は

$$z_{(P)}^{n+3} = z_{(P)}^{n+1} - 2\Delta t \cdot i_{(P)} \quad (4.56)$$

となる²⁵⁾。

4.8 境界条件

4.8.1 流出境界

流出境界条件は、X・Y 成分それぞれの計算流域外への流出量を求めるものである。本資料では、長頂堰の公式を用いて流出量を計算する。

この計算は運動方程式を解いた後に同じ時間ステップにおいて、境界の格子辺で計算する。計算領域外の格子点で流動深が定義されないため、計算領域内側に隣接する格子点の流動深だけを用いる²⁶⁾。

4.8.1.1 X 成分

X 成分の流出量 Q_{out} は単位幅流量 M に通過する Y 方向の格子幅と時間刻みを乗じて計算する。

$$Q_{out} = M_{(P)}2\Delta tdy \quad (4.57)$$

ただし、± X 方向の境界のいずれかに応じて、単位幅流量は、

$$M_{(P)} = \pm 1.6h_{(w)}^{3/2} \quad (4.58)$$

23) calclcf.f90

24) calrho.f90

25) calbedelev.f90

26) boundaryq.f90

と計算する (図 4.9)。

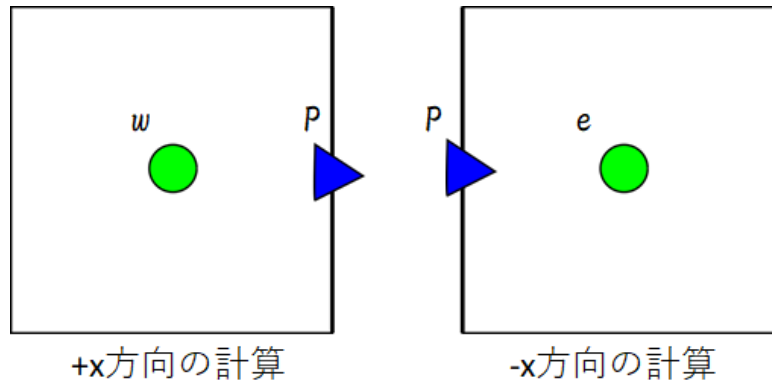


図 4.9 X 成分の流出流量の算定に用いる格子点

4.8.1.2 Y 成分

X 成分と同様に, 次のように計算する。

$$Q_{out} = N_{(P)} 2\Delta t dx \quad (4.59)$$

ただし, $\pm Y$ 方向の境界のいずれかに応じて, 単位幅流量は,

$$N_{(P)} = \pm 1.6h_{(s)}^{3/2} \quad (4.60)$$

と計算する (図 4.10)。

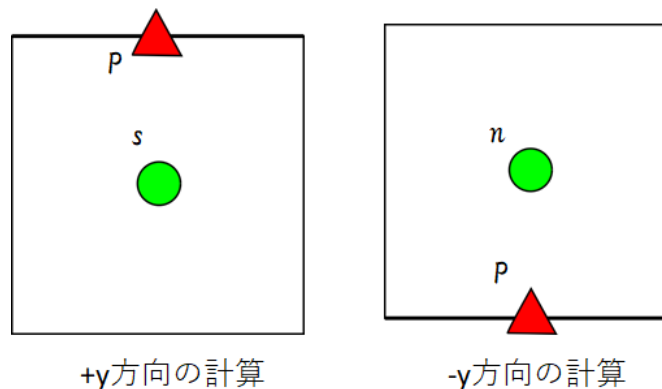


図 4.10 Y 成分の流出流量の算定に用いる格子点

4.8.2 流入境界

流入境界条件は, 計算領域内のハイドログラフ流入点においてハイドログラフを与える境界条件である。ハイドログラフは次のように等流状態を想定した流れとして計算領域に与えるものとする。

- 入力するハイドログラフは流量の大きさ (スカラー) で与えるため, 格子幅で除して単位幅流量 q_{in} とする。
- 与えられた流量が等流で流下するものとみなし, マニング平均流速公式を用いて等流水深 $h = [q_{in}n/\sqrt{I}]^{3/5}$ を求める。ここに, n は粗度係数, I は河床勾配であり, それぞれパラメータとハイドログラフの入力ファイルで与える²⁷⁾。
- 流速の大きさ $u = q_{in}/h$ とする。
- 流速の向きは, パラメータファイル (params.dat) で初期値として与える²⁸⁾。

27) calset.f90

28) inputparam.f90。次章の入力ファイルの解説を参照のこと。

これらの値（水深、流速、流向）は、入力するハイドログラフ（区分時間とその間の流量、ハイドログラフの本数）に応じて、逐次計算の前に配列に記憶する²⁹⁾。 $t = (n + 2)\Delta t$ （運動方程式を解く時間ステップ）において流量とその流出方向をその配列から呼び出し、 $t = (n + 3)\Delta t$ （連続式を解く時間ステップ）において等流水深をその配列から呼び出す³⁰⁾。

4.8.3 水際境界

非常に小さな流動深 ε を閾値として、流動深の大小と比較して水域か陸域かを判定する。つまり、 $h \geq \varepsilon$ なら流れが存在する水域、 $h < \varepsilon$ なら流れが存在しない陸域と判定する。水域の境界は本資料では水際と呼ぶ（図 1.3 を参照）。 ε は、岩佐ほか（1980）と同様に 0.001 [m] を初期値とする³¹⁾。

4.8.4 壁境界

壁境界は設定しない。

4.9 物理的に不整合な計算結果の対処法

この節では、本資料のプログラムにおける物理的に不整合な計算結果の対処手法として、流動深と流量の修正方法、土砂濃度の修正方法、侵食・堆積速度式の修正方法、について述べる³²⁾。

4.9.1 流動深と流量の修正

流れ全体の連続式を解いたときに負の流動深が発生した場合、流動深が零になるようにその格子に流入する流量を調整する（岩佐ほか，1980）。サブルーチン `hdepth` において定義された再帰関数によって実現されている。

4.9.2 土砂濃度の修正

粗粒分・細粒分砂礫の連続式を計算するサブルーチン `calclcf` において、粗粒分土砂濃度 C_L が、 C_{90} より大きい場合 C_{90} に修正し、零より小さいときは零に修正する。その余剰土砂濃度の修正は連続式の破れとなるため、河床標高に戻す処理が実装されている³³⁾。この機能を用いる場合、入力パラメータ `flg_rebed` を 1 に設定する。細粒分砂礫の土砂濃度 C_F が河床堆積層の細粒分砂礫の充填土砂濃度 C_{*F} （入力パラメータのひとつで `param.dat` で与える）を超えた場合は、その値に設定する。細粒分砂礫の土砂量での河床標高の修正は実装していない。

4.9.3 侵食・堆積速度式の修正

本資料のプログラムでは、侵食・堆積速度を計算するサブルーチン `calaii` において、1 時間刻みで侵食可能な最大侵食深と計算される最大侵食速度より大きな侵食速度が生じた場合は、侵食速度は最大侵食速度に修正する。また、河床勾配が内部摩擦角 $\tan \phi$ 以上の場合、高橋の土石流理論においては土石流が発生せず平衡土砂濃度の値が計算できなくなる。このとき、粒子の骨格応力により土が破壊しないものとみなし、侵食速度を零に設定する。

29) `calset.f90`

30) `boundaryq.f90`, `boundaryh.f90`

31) パラメータ入力ファイルで変更することができる。ソースコードでは `th` (threshold) という変数であり、いくつもの箇所での判定を行うため、対応するソースコードを記さない。

32) 計算機科学上のアルゴリズムなどの理論と実装、利用者の関係について、杉原（1994, p.iii-iv）は、「数値計算には誤差は含まれないという理想的な世界を仮定した上でこの体系が成り立っているために、誤差が発生する通常の計算機環境では、それらのアルゴリズムは、必ずしも理論どおりの性能を発揮しない（略）。しかし、実際に使えるプログラムをつくらなければならないソフトウェア技術の立場からは、（略）『誤差が発生すると暴走する危険性のあるプログラム』は『虫を含んだプログラム』と同義語であり、（略）数値誤差の問題（略）の解決はプログラマに押し付けられている。（略）プログラムづくりの現場では最も普通に行われている対処方法は（略）対処療法に過ぎず、根本的解決にはなっていない。（略）プログラムづくりの段階でも解決できないとなると、結局、最後のしわ寄せはユーザがかぶることになる。」と述べている。本節に示すのはここでいう対処療法であり根本的解決法ではない。

33) 鈴木（2013）は土石流の計算技術の観点から連続式を満たさない場合の処理について、水深が負になるあるいは土砂濃度が過剰になる場合に連続式を満たすように、侵食・堆積速度式の調整方法を提案した。太田ほか（2018）はこれを発展させ、流砂量の誤差、土石流の土砂濃度が上限を超える場合の対処法、緩勾配地点における解の不安定性を解消する時間刻み依存性の問題を考慮した 2 次元浅水流型の土石流解析法を提案した。本資料では鈴木（2013）の土砂濃度の修正に伴う河床変動を実装した。パラメータ入力ファイル `param.dat` において `flg_rebed=1` とすれば、鈴木（2013）が述べる河床調節法の一部が機能する。ただし、本プログラムは土砂濃度の計算が不安定なためか、連続式を満たしていないためか、大きな河床変動が生じる場合がある。使用には注意してほしい。なお、粗粒分土砂により河床標高は修正しない。

第5章 プログラムへの実装

本章では Fortran 言語を用いたプログラムへの実装に関する事項を述べる。本章の構成は、未知量と出力、Fortran の実行手順と OpenMP 並列化、ソースコードの全体フロー、サブルーチン解説、並列化に適したデータ構造、入力ファイル書式である。本章 5.4 節以降はソースコードを読む際に参考にされたい。

プログラミング言語に関する文字列（ファイル名、サブルーチン名、変数名）はゴシック体で表す。

5.1 未知量と出力

水と土砂の流れの数値解析を行う目的として、例えば次の事項を推定することが考えられる。括弧内にその情報を得るために必要な物理量を示す。

- 水と土砂の混合物の流れが到達する範囲か（流動深や流量、掃流力）
- ある地点にどれくらい時間で土石流が到達するか（到達時間）
- 河道を越える流動深になる地点はどこか（流動深）
- 砂防堰堤など土木構造物に作用する力の大きさはどの程度か（流体力）

流体力などを出力する手順を簡潔に示す。

括弧に示す基礎方程式を解くと、次の未知量の近似解（数値）が得られる。

1. 単位質量・単位幅あたりの流量 q_x, q_y （運動方程式）
2. 流動深 h （流れ全体の連続式）
3. 粗粒分砂礫の土砂濃度 C_L （粗粒分砂礫の連続式）
4. 細粒分砂礫の土砂濃度 C_F （細粒分砂礫の連続式）
5. 河床の標高 z （河床位方程式）

その近似解を使えば、例えば、次の値を得ることができる。

1. 流動深
2. 流量
3. 到達時間（時間ステップ毎に結果を出力）
4. 流速 $(u, v)^T = \mathbf{q}/h$
5. 流体力 $(f_x, f_y)^T = \rho_T h \|\mathbf{u}\| (u, v)^T$
6. 掃流力 $\tau_b = f_r \rho_T \|\mathbf{u}\| (u, v)^T$

ここに、 T は転置、太文字はベクトルである。

本資料のプログラムは土石流など水と土砂の混合物の流れが到達する範囲を得ることを目的に開発されたものであり、出力値は限定されている。

5.2 Fortran と OpenMP の概説

本節では Fortran 言語や OpenMP を簡潔に述べる。詳細は、例えば牛島 (2007); Metcalf *et al.* (2011) や牛島 (2006) を参照されたい。

5.2.1 Fortran の実行と構成

Fortran は 1954 年から存在する科学技術計算用の強い静的型付けのプログラミング言語である。

Fortran で記述されたソースコードの基本的な構成は次の通りである。

- `program` から始まり、上から順に命令が実行され、`end program` で終わる。`program` から `end program` ままでを主プログラムという。

- 主プログラムが副プログラムを呼び出すと、呼び出した主プログラムの実行が一時停止され、呼び出された副プログラムの処理が終了されるのを待つ。副プログラムが別の副プログラムを呼び出すときは、前述の主副と同様の関係が成立する。つまり、副プログラムの呼び出しはスタックを構成する。副プログラムとしてはサブルーチン (subroutine) とモジュール (module) があり¹⁾、call や use で呼び出す。
- 主プログラム、副プログラムともに、個々のプログラムは基本的に次のように構成される²⁾。
 - モジュール使用の宣言 (use module)
 - 変数の宣言
 - 変数の初期化 (宣言時に同時に行う場合もある)
 - 実行 (副プログラムの呼び出しを含む)
 - 結果を呼び出し元に返す
- 本資料のソースコードは、手続き型プログラミングと呼ばれるプログラミング・パラダイムで設計されており、古典的な書き方のプログラムである。変数の大半は、ソースコード arrayc.f90 に書かれた module arrayc において、グローバル変数として宣言される。主・副プログラムにおける冒頭の use arrayc によって主・副プログラム間で変数の値が共有される。そのため、本資料のソースコードの副プログラムでは、結果を呼び出し元に返す処理はないものが多い。

5.2.2 OpenMP の概説

- 並列計算は分散メモリ型、共有メモリ型に大別される。OpenMP は後者の概念を実装し標準化された並列計算の手法のことである。
- 共有メモリ型並列計算は、複数 CPU から1つのメモリが見えるパーソナルコンピュータ等で走らせるプログラムを並列化する場合に利用される。
- OpenMP では、ソースコードに、!\$omp などの形式でコメント (実行に無関係な文字列) として指示文 (ディレクティブ) を記し、並列する指示内容と範囲を「!\$omp 指示内容」と「!\$end omp 指示内容」で指定する。
- OpenMP に対応しているコンパイラを用いて、コンパイル時に並列化指示をすれば並列化された実行ファイルが生成される。そのコンパイル時に並列化する指示がなければ、OpenMP に関する指示文はコメントとみなされるため、同一ソースコードを非並列・並列の両環境で利用できる利点がある。

5.3 プログラム全体フロー

- 本資料のソースコードでは、変数は module arrayc でグローバル変数として宣言される。
- program から end program のプログラム本体および全体フローは main.f90 に書かれている。
- プログラム全体フローは、次のように構成される：
 1. 変数宣言 (グローバル変数)
 2. 入力部
 3. 初期化
 4. 時間ステップ $2\Delta t$ 逐次計算部
 - X,Y 方向の運動方程式
 - (Δt 進みスカラー評価点に移り) 流れ全体の連続式
 - (流れの粗粒・細粒分砂礫の連続式と河床位方程式)
 - 中間出力データの作成と出力
 - 終了時刻になるまで逐次計算部の繰返し
 5. 最終結果の出力

1) 他プログラミング言語であれば、関数に相当するものが subroutine、構造体やクラスに相当するものが module と読みかえれば概ね問題ないと考えられる。本資料では Fortran における function は用いない

2) 前章の差分式で述べたのは「実行」に関する部分である。本資料のソースコードを読めばわかるように、副プログラムのソースコードの大部分は、変数の宣言、初期化、位置の走査、評価点 P の呼び出しである。

5.4 サブルーチン解説

本節では、プログラム全体フローに沿って、いくつかのサブルーチンを機能毎にまとめて述べる。プログラム全体フローとの対応を括弧に示す。なお、流砂計算の実施を決める状態変数 `noSediment` は、入力ハイドロに土砂濃度がなければ真 (True)、与えられた場合には偽 (False) となる。

1. 前処理 (入力部・初期化)
2. 運動方程式の処理
3. 流れ全体の連続式の処理
4. `noSediment` が偽 (.false.) であれば、流れの粗粒・細粒分砂礫の連続式と河床位方程式の処理
5. 後処理 1 (中間出力データの作成と出力, 次の逐次計算への準備)
6. 後処理 2 (最終結果の出力)

本節では、上に示す 6 つの区分のそれぞれに対して、概要、属するサブルーチン、各サブルーチンの機能を述べる。個々のサブルーチンでは、機能を 1, 2 行程度で示し、ソースコードの順番に沿って内容を示す。

次節に入る前に、ほとんどのサブルーチンの解説において、`g, xs, ys` の記号を使用して全体の格子番号から着目点とステンシルの格子番号を取得する操作を行うため、ここで簡潔に述べる。詳細は後述「データ構造」に示す。

図 5.1 に示すように、全体格子点番号のうち、格子中央の格子点番号を `g` に、格子辺の X, Y 方向の格子点をそれぞれ `xs` と `ys` の配列に記憶する。連続式や運動方程式を各サブルーチンにおいて、差分方程式を評価する点 `P` (ベクトルなら格子辺, スカラーなら格子中央) を `g` (連続式の時) あるいは `xs, ys` (運動方程式の時) から取り出す (図 5.2)。

評価点 `P` を決めた後、差分方程式を解くために用いるステンシル (E, N, W, S など) の格子点が必要となる。これらの格子点番号には、例えば連続式を解く時間ステップでは、`g` に接続する周囲の格子点が記憶された隣接リスト (詳細は後述「データ構造」) を用いてアクセスする。

なお、図 5.1 において、格子番号は、数学表記と異なり、X 方向を `j`, Y 方向を `i` と書く。これは、プログラム内では 2 次元配列 (例えば、`xs` と `ys`) が `i` 行 `j` 列の行列であるから、`i` が Y 方向, `j` が X 方向に対応するためである。

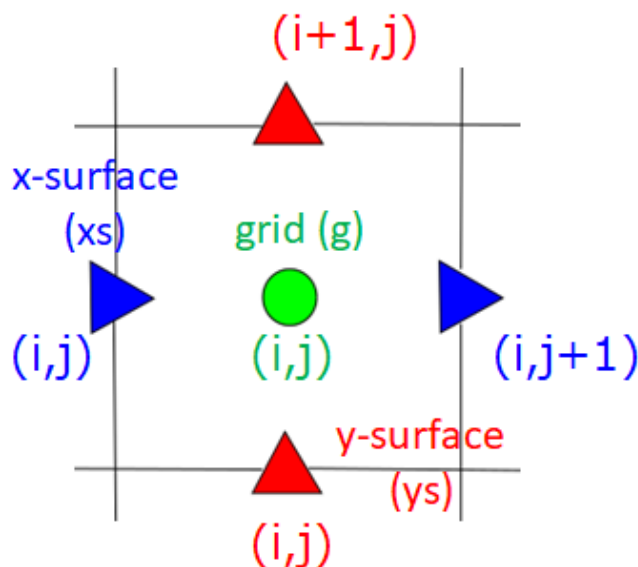


図 5.1 評価点・格子点・格子辺の呼称とインデックス

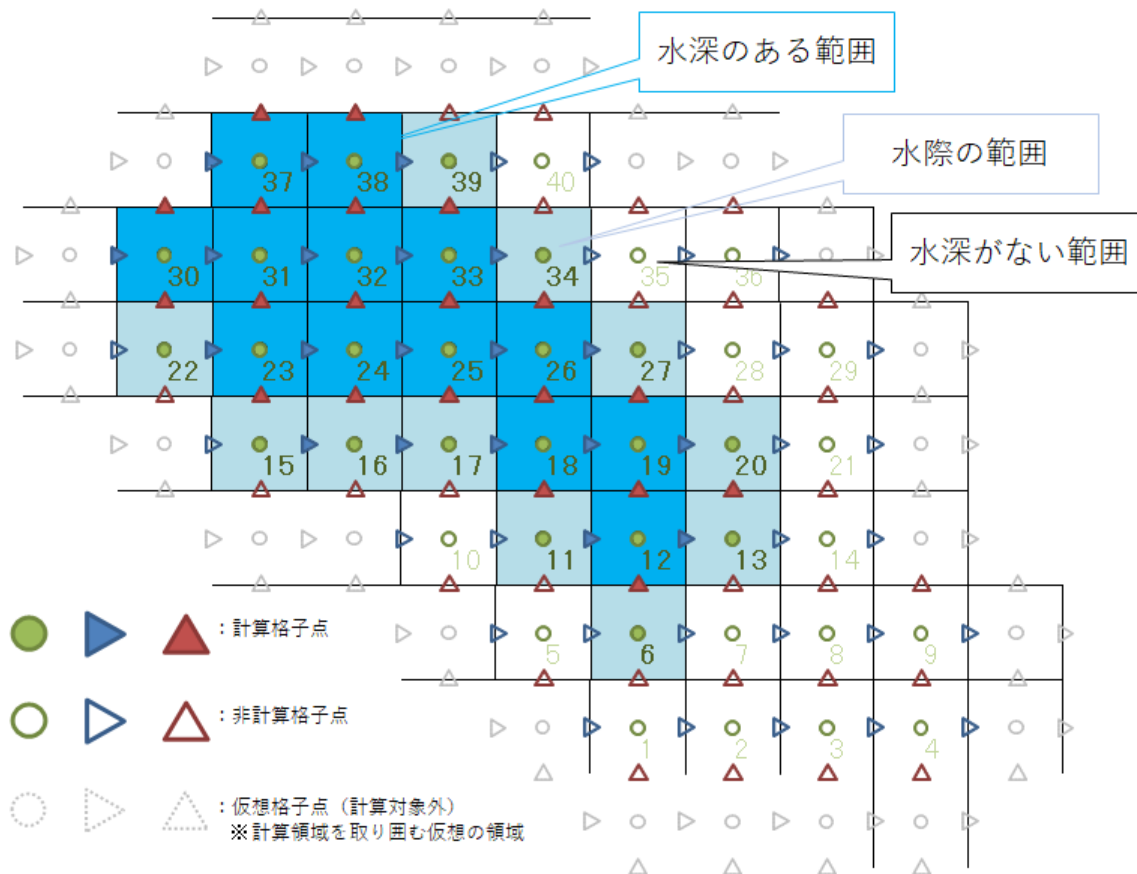


図 5.2 計算領域における水域・陸域格子と格子点・非格子点・仮想格子点の関係

5.4.1 前処理

前処理の段階においては、入力ファイルを読み込み、グローバル変数の配列の大きさや初期値を与えるなどを行う。

- 入力部の構成サブルーチン：flopen, inputdem, inputhydr, inputparam
- 初期化の構成サブルーチン：clear, calset, wetdry

5.4.1.1 flopen

このサブルーチンは、filename.dat から入出力ファイル名を読み込み、各ファイルを open 文でユニット番号 (unit number) と関連づけを行う。

入出力ファイル名は filename.dat によって任意に設定できる。本章では以下に示す名称を用いる。

- ユニット番号と関連付けられたファイル一覧を示す。
 - 10: 入出力ファイル名の入力ファイル filename.dat (flopen の中で close する。)
 - 80: エラーが発生した場合に確認するための出力ファイル errorout.dat
 - 11: サブルーチン inputdem で用いる地形データ入力ファイル dem.dat
 - 20: サブルーチン inputdem で用いる境界データ入力ファイル boundary.dat
 - 22: サブルーチン inputhydr で用いる計算総時間などを決める入力ファイル time.dat
 - 12: サブルーチン inputhydr で用いるハイドログラフ入力ファイル hydr.dat
 - 13: サブルーチン inputparam で用いるパラメータ入力ファイル param.dat
 - 32: サブルーチン snapshot で用いる計算終了時の出力ファイル end.csv や時系列出力データ
 - 33: サブルーチン outmaxvalue で用いる計算中の最大値の出力ファイル max.csv
 - 34: サブルーチン inputhydr, inputparam で用いる入力値確認用の出力ファイル inputdata.csv

- 99: サブルーチン `tau` で用いる限界掃流力 τ_{*c} 以上の位置を示す出力ファイル `tau_strc.csv`
- 80: メイン program で用いる領域外への総流出流量を示す出力ファイル `outflow_volume.dat`

5.4.1.2 inputdem

このサブルーチンは、地形データと境界データの読み込みを行う。

- 地形データを `dem.dat` から読み込む。
- 計算領域のセル総数、行数、列数、X 方向・Y 方向の格子幅を読み込む。
- `arrayc` で宣言したグローバル変数の配列の大きさを決定する。
- 格子点の隣接リストを構築する。
- 流出境界（と壁境界）を `boundary.dat` から読み込む。

5.4.1.3 inputhydr

このサブルーチンは、計算総時間と時間刻み、ハイドログラフの入力を行う。

- 計算時間と時間刻みを `time.dat` から読み込む。
- 流入境界であるハイドログラフとその位置・向きを `hydr.dat` から読み込む。

5.4.1.4 inputparam

このサブルーチンは、パラメータの読み込みを行う。

- パラメータを `param.dat` から読み込む。

5.4.1.5 clear

このサブルーチンは、変数の初期化を行う。

- 未知量、出力するための変数などの初期値を 0 とする。

5.4.1.6 calset

このサブルーチンは、入力データを加工し、逐次計算の前処理を行う。

- 河床堆積層の土砂濃度、河床堆積層厚、格子内の初期土砂量を初期化する。
- `inputhydr` で読み込んだハイドログラフの 1 波形毎に、X・Y 成分の単位幅流量、マニング平均流速公式による流動深、流速を計算し配列に記録する。
- ハイドログラフ 1 波形毎に、計算領域に流入する総流量、総土砂量を計算する。
- 総土砂量が零（きわめて小さい値）でなければ、土砂を計算するフラグ `noSediment` を `False` に設定する。
- 流動深を基に、副プログラム `wetdry` を呼び出して、全域の格子点に水域 (`wet`) と陸域 (`dry`) を設定する。

5.4.1.7 wetdry

このサブルーチンは、格子点に水域 (`wet`) と陸域 (`dry`) の属性付けを行う。

- 流動深に基づき、その存在の有無により格子がもつ水域／陸域属性を更新する。
- スカラー評価点 (`g`)、X 方向の格子辺 (`xs`)、Y 方向の格子辺 (`ys`) を陸域に初期化する。
- 粗度係数を全格子に与える。
- 全格子に対して、水域であれば、その周囲のステンシルへの隣接リストを構築する。
- 水域のうちハイドログラフ流入点の格子は陸域に再設定して流出流量の計算点に初期化してから、水深が ϵ より大きい場合は水域格子に再設定する。
- 計算領域の周囲に設定する仮想格子を陸域（計算しない格子）に設定する。
- `g`, `xs`, `ys` におけるそれぞれの水域の数と、全体格子番号からその格子にアクセスするためのリスト（イン

デックス)を作成・更新する。

なお、このサブルーチンで実装したデータ構造は、高橋ほか(2018)が計算速度を高速するために工夫したものである。初期化と逐次計算の1時間ステップ間(2Δt)に一度だけ水域と陸域を判定し、水域を差分方程式を解く対象格子とすることで、その時間ステップ間における格子走査過程の際に計算有無を判定しなくすむため、複数CPUに計算を振り分ける並列計算が可能となる。これは、従来の計算プログラムでは、運動方程式(x方向およびy方向)および連続式を逐次計算するたびに、全格子点を走査して、水域格子点と陸域格子点を判定し計算の有無を判定したために、複数CPUに計算を配分できず並列計算の長所を活かせないことを改善したものである。

5.4.2 運動方程式の処理

運動方程式の処理の段階においては、水域を対象として、 $t = (n + 2)\Delta t$ の流量を算出し、流入・流出境界条件により一部の格子で流量を修正することを行う。

- 構成サブルーチン：qmflux, qnflux, boundaryq

5.4.2.1 qmflux

このサブルーチンは、X方向の運動方程式を解き、次の時間ステップの流量 $M_{(P)}^{n+2}$ を算出する(式4.18)。

- 未知量 $M_{(P)}^{n+2}$ の変数qm2(i)を初期化する。
- 全体の格子番号から水域格子の番号を取得し、さらにその周囲のステンシルの格子番号を取得する。
- $t = n\Delta t$ における、X方向の流速、Y方向の流速、X方向の時間平均流量、 XDX 、 XDY を計算する。
- 流動深の時間平均 $\overline{h_{(P)}}^x$ 、水位の差分 $\delta_x[(h+z)_{(P)}]$ 、土砂濃度の平均値 $\overline{C_{L(P)}}^x$ 、Y方向の流速の平均値 $\overline{v_{(P)}}^{xy}$ を計算する。
- 流砂形態に応じて、流動抵抗係数 f_r を求める。ただし、ソースコードでは、 $\frac{fr\|U_P\|}{h_P^x}$ を求めて frx に代入されるため、差分方程式の4章における f_r とソースコードは1対1に対応していない。
- fr_{q1} と fr_{q2} を計算し、 $t = (n + 2)\Delta t$ の流量 $M_{(P)}^{n+2}$ を求める。

5.4.2.2 qnflux

このサブルーチンは、Y方向の運動方程式を解き、次の時間ステップの流量 $N_{(P)}^{n+2}$ を算出する(式4.27)。

- 計算手順は、qmfluxと同様のため省略する。

5.4.2.3 boundaryq

このサブルーチンは、流入境界、流出境界に位置する $M_{(P)}^{n+2}$ 、 $N_{(P)}^{n+2}$ の値を境界条件に従って修正する。

5.4.3 流れ全体の連続式の処理

流れ全体の連続式の処理においては、水域を対象として、 $t = (n + 3)\Delta t$ の流動深を計算し、流入・流出境界条件により一部の格子で流動深を修正することを行う。

- 構成サブルーチン： hdepth, boundaryh

5.4.3.1 hdepth

このサブルーチンは、流れ全体の連続式を解き、次の時間ステップの流動深 $h_{(P)}^{n+3}$ の計算を行う。

- 未知量 $h_{(P)}^{n+3}$ の変数hn3(i)を初期化する。
- 水域における格子を走査する。
- スカラー評価点(g)に関するステンシルの流量を得る。
- $\delta_x[M_{(P)}^{n+2}]$ 、 $\delta_y[N_{(P)}^{n+2}]$ を計算し、hn3を求める。
- その流動深が負にならないように、その格子の流量の収支と流動深のバランスを調整する(負でなくなるまで再帰計算)。

5.4.3.2 boundaryh

このサブルーチンは、流入境界位置における流動深 $h_{(P)}^{n+3}$ を、ハイドログラフから計算した等流水深に変更する。

5.4.4 流れの粗粒・細粒分砂礫の連続式と河床位方程式の処理

本項に示す複数のサブルーチンは、入力ハイドロに土砂が設定されたか判定する noSediment が偽 (False) の時に実行される。

流れの粗粒・細粒分砂礫の連続式と河床位方程式の処理の段階においては、平衡土砂濃度を求め、流れの粗粒・細粒分砂礫の連続式を解き各種土砂濃度を算出し、河床位方程式を解き河床標高の変化を計算する。この間、流れ全体の連続式 hdepth で使用した侵食・堆積速度式を使用し、新たに得た土砂濃度や平衡土砂濃度を用いて、次時間ステップの侵食・堆積速度を計算する。

- 構成サブルーチン：qbflux, boundaryqb, gradbed, cinfty, calclcf, calrho, calbedelev, calaii
- 次の順番にサブルーチンが実行される。
 - 格子からの流出土砂量の算出 (qbflux)
 - その流出土砂量のハイドログラフで流入した土砂量に更新 (boundaryqb)
 - 勾配の算出 (gradws)
 - 平衡土砂濃度の算出 (cinfty)
 - 流れの粗粒分・細粒分砂礫の連続式から土砂濃度の計算 (calclcf)
 - 流れの間隙流体密度とみかけの密度の計算 (calrho)
 - 河床位方程式を用いた河床位変化量の算出 (calbedelev)
 - 平衡土砂濃度を用いた侵食・堆積速度式の算出 (calaii)

5.4.4.1 qbflux

このサブルーチンは、粗粒分・細粒分砂礫の連続式を解く準備として、流出土砂量を計算する。

- 流砂量フラックスのベクトルを初期化する。
- 水域における格子辺の番号を取得する。
- X 方向の格子辺 xs において、粗粒・細粒分土砂の順番で、土砂濃度と流量の積として流砂量フラックスを算出し、格子内に存在した以上の土砂が流出しないよう流砂量を修正する。
- Y 方向の格子辺 ys において、同様に計算する。

5.4.4.2 boundaryqb

このサブルーチンは、流出境界における流出土砂量に境界条件を適用して修正を行う。

- 初期化する。
- 流出境界の格子辺番号を取得する。
- 流出境界において、計算領域外に流出する粗粒分・細粒分の流出土砂量を計算する。

5.4.4.3 gradws

このサブルーチンは、水面勾配 sei の計算を行う。

- 初期化する。
- 水域の格子点 g における、水面勾配を計算する。

5.4.4.4 cinfty

- 平衡土砂濃度 c1p を初期化する。
- 水域の格子点 g における勾配の取得し、水中重量 wsgなどを計算する。
- 勾配に応じて流砂形態を判定し、それぞれの平衡土砂濃度（それぞれ式 4.35, 式 4.34, 式 4.33）を計算する。

- 平衡土砂濃度を次のように修正する。
 - $0.9C_*$ ($c90$) 以上であれば、この値に修正
 - $c90$ が負であれば 0 に修正

5.4.4.5 calclcf

このサブルーチンは、流れの粗粒・細粒砂礫の連続方程式を解き、粗粒・細粒砂礫の格子内の体積（流動深）を算出し、土砂濃度を計算する。

- 流れの土砂濃度 $c13$, $cf3$, を初期化する。
- 水域の格子点 g において、流れの粗粒分砂礫の連続式を解き V_L ($vv13$) を得て、粗粒分砂礫の土砂濃度 $C_{L(P)}^{n+3}$ ($c13$) を得る。
- 同様に、流れの細粒土砂の連続式を解き、 V_F ($vvf3$) を得て、細粒分砂礫の土砂濃度 $C_{F(P)}^{n+3}$ ($cf3$) を得る。一端計算した細粒分砂礫の土砂濃度は、負であれば零に、 C_{*F} より大きければその値に修正する。河床調整は行わない。

5.4.4.6 calrho

このサブルーチンは、calclcf で計算された土砂濃度を用いて、間隙流体の質量密度 rof 、流れ全体のみかけの質量密度 rot を計算する。

5.4.4.7 calbedelev

- 水域の格子点 g の侵食・堆積速度 aii により、河床位方程式（式 4.56）を解き、 $z_{(P)}^{n+3}$ ($z13$) を計算する。

5.4.4.8 calaii

- 水域の格子点 g における粗粒分砂礫の土砂濃度 $c13$ を取得する。
- $c13$ と平衡土砂濃度 $c1p$ とを比較し、式 4.40 や式 4.41 に基づいて、侵食・堆積速度 $i_{(P)}$ (aii) を算出する。

5.4.5 後処理 1

後処理 1 の段階においては、中間出力データの作成と次の逐次計算の準備を行う。

- 構成サブルーチン：wetry, velcal, flowpower, pickcal, tractive, replace

計算結果の可視化のために、逐次計算時および計算終了時に物理量を CSV ファイル等へ出力する準備として、次の物理量を計算する。

1. 流動深
2. 流速, 流量
3. 土砂濃度
4. 力

5.4.5.1 wetry

このサブルーチンについては前処理の部分で解説した。後処理の段階で呼び出す場合、 $2\Delta t$ における基礎方程式群を解いた後の流動深分布によって、水域・陸域の属性値を更新する。

5.4.5.2 velcal

このサブルーチンは、スカラー評価点における流速の大きさを算出する。

- 初期化する。
- 格子辺 xs において、
 - 4 周囲の格子点 g の流動深の平均値 \bar{h}^{xy} を計算する。

- その格子辺における 1 時間ステップ前と 2 時間ステップ前の流量を平均して流速 $u_{(e)}$ ($uu1$) などを計算する。
- 格子辺 ys でも同様に流速 $v_{(s)}$ ($vv1$) などを計算する。
- 流入境界と壁境界で流速と修正する。
- 水域の格子点 g における平均流速 $\|U\| = \sqrt{u^2 + v^2}$ (uva) を計算する。

5.4.5.3 flowpower

このサブルーチンは、出力の前処理として流体力を計算する。

- 水域・陸域に区別なく全格子点 g を対象に、流体力ベクトル $f = \rho_T h U u$ (ftx, fty) とその大きさ $\|f\|$ (fta) を計算する。

5.4.5.4 pickcal

このサブルーチンは、すべての格子における計算時間を通じた最大値などを出力するための準備を行う。

- 水域・陸域に区別なく全格子点 g を対象に、次の最大値・最小値等を計算する。格子辺の値（流量など）も本プログラムでは格子点 g で出力するため、格子辺の値（流量など）は平均化処理を行う。
 - 流量ベクトル (qmx, qny) とその大きさ qqa
 - 粗粒分流砂量ベクトル ($qblmx, qbly$) とその大きさ $qbla$
 - 細粒分流砂量ベクトル ($qbfx, qbfy$) とその大きさ $qbfa$
 - 浸水開始時刻 $arvt$
 - 最大流動深 $hmax$
 - 最大河床堆積層厚 $zmax$ とその時刻 $tzmax$
 - 最小河床堆積層厚 $zmin$ とその時刻 $tzmin$
 - 最大流動深 hz
 - 最大水面標高 hz とその時刻 $thzmax$
 - 最大流速 (x 方向 uux , y 方向 vvy , 大きさ uva) とその時刻 $tvmax$
 - 最大流量 (x 方向 qmx , y 方向 qny , 大きさ qqa) とその時刻 $tqmax$
 - 最大流体力 (x 方向 ftx , y 方向 fty , 大きさ fta) とその時の時刻 $tfmax$, 流動深 $hafmax$, 河床変動量 $zafmax$
 - x 方向の CFL 値 $(u + \sqrt{gh})/(\Delta x/2\Delta t)$ $ucfl$
 - y 方向の CFL 値 $(v + \sqrt{gh})/(\Delta y/2\Delta t)$ $vcfl$
 - 2 つの CFL の大きさ $c1f$

5.4.5.5 tractive

このサブルーチンは、掃流力 $\tau_b = f_r \rho_T \|u\|^2$ の計算を行う。ただし、 f_r の対象流砂形態は掃流砂流のみである。

- 掃流力の配列 τ_* ($taus$) を初期化する。
- 水域の格子点 g において、 $\tau_* = u_*^2/(sgd_L)$ を計算する。
- その最大値を $tapc$ に記録する。

5.4.5.6 replace

このサブルーチンは、次の逐次計算への準備として、時間の関する物理量の配列を入れ替えを行う。

- 新しく計算された時刻の様々な値を前の時間ステップの配列に代入し、次の時間ステップ計算の準備する。

5.4.6 後処理 2

後処理 2 の段階では、計算結果を CSV ファイルとしてハードディスクに書き込む処理を行う。

- 構成サブルーチン： `snapshot`, `outmaxvalue`, `tau`

5.4.6.1 snapshot

このサブルーチンは、計算した結果を CSV ファイルに出力する。逐次計算終了後だけでなく、逐次計算開始前にも必ず実行され、逐次計算中は `time.dat` の 1 行目 `iout` に指定があれば実行される。

- 出力する CSV ファイル名は、`filename.dat` で設定した接頭辞（拡張子を除く）により、「接頭辞+_+time.dat の行に相当する連番+.csv」となる。
- 逐次計算中に出力は、`inputparam` で読む入力ファイル `param.dat` における `iout` を 1 に設定すればよい（1 以外であれば出力しない）。出力間隔は、`time.dat` と `hydr.dat` の行毎である。
- 出力時間ごとに 1 つの CSV ファイルが生成され、次に示す出力項目が全格子分出力される。

0.格子番号

1.平面直角座標系における座標 (x,y)

2.平面直角座標系における座標 (x,y)

3.経度, 緯度

4.経度, 緯度

5.標高

6.最大水面標高

7.流動深

8.流速 (x)

9.流速 (y)

10.流量 (x)

11.流量 (y)

12.その位置の水域・陸域判定結果 (1:wet, 0:dry)

13.勾配

14.河床標高の変化量 (正は堆積, 負は侵食)

15.粗粒分砂礫の X 方向の流砂量

16.粗粒分砂礫の Y 方向の流砂量

17.細粒分砂礫の X,Y 方向の流砂量

18.細粒分砂礫の X,Y 方向の流砂量

19.粗粒分土砂濃度

20.細粒分土砂濃度

21.侵食・堆積速度 (正は侵食, 負は堆積)

22.平衡土砂濃度

- 出力ファイルは `map__00001.csv` などの連番で出力される。
- GIS などによる可視化を想定し、座標も記録された CSV ファイルが出力される。
- 本資料補足スクリプト `mkvtk.py` より VTK ファイルを作成すれば、`ParaView` などの可視化アプリケーションで動画として結果をみる事ができる。

5.4.6.2 outmaxvalue

このサブルーチンは、計算中の最大値等の結果を記録した CSV ファイルを出力する。

- CSV ファイルに計算中の最大値が、逐次計算が終了した後に出力される。
- CSV ファイル名は、`filename.dat` で指定した名前となる。
- 計算終了後にファイルが生成され、次に示す出力項目が全格子分出力される。
 - 座標 (x,y)

- 経度, 緯度
- 初期標高
- 最大流動深
- 最大流動深までの計算開始後からの時間
- 最大河床堆積高 (初期標高に対する河床高変化量)
- 最大河床堆積高までの計算開始後からの時間
- 最大河床侵食高 (初期標高に対する河床高変化量)
- 最大河床侵食高までの計算開始後からの時間
- 最大水位 (標高+流動深)
- 最大水位までの計算開始後からの時間
- 最大水位時の流動深
- 最大水位時の河床標高
- 最大流速の大きさ
- 最大流速の大きさまでの計算開始後からの時間
- 最大流速 (x)
- 最大流速 (y)
- 最大流量の大きさ
- 最大流量の大きさまでの計算開始後からの時間
- 最大流量 (x)
- 最大流量 (y)
- 最大流体力の大きさ
- 最大流体力の大きさまでの計算開始後からの時間
- 最大流体力 (x)
- 最大流体力 (y)
- 最大流体力の時の流動深
- 最大流体力の時の標高
- 最大到達時間
- 最大掃流力
- 土砂移動判定フラグ

- GIS などによる可視化を想定し, 座標も記録された CSV ファイルが出力される³⁾。

5.4.6.3 tau

このサブルーチンは, 内田ほか (2011) の方法の実装である。各位置の最大掃流力と限界掃流力の比較 (フラグ 1) およびハイドログラフ流入点からの連続性 (フラグ 2) を計算する⁴⁾。

- 逐次計算終了後に 1 回だけ計算する。
- サブルーチン `tractive` で計算される最大掃流力と, `param.dat` で設定した岩垣の限界掃流力 (土木学会 水工学委員会, 2019, p.284) とを比較して, 有効掃流力 (掃流力から限界掃流力を引いた値) が正の格子 (限界掃流力を上回る掃流力が作用し, 土砂が流れる格子) を抽出し, 1 のフラグを立てる。
- その後, ハイドロ流入点 (流入境界) から 8 方向に隣接する格子を連ねて, 2 のフラグを立てる。
- `filename.dat` で設定した出力ファイル名 (`[tau_strc].csv`; [] は任意に設定可能) に, それぞれの粒径に対するフラグが記録される。
- 有効掃流力が正の格子を 1, さらにハイドロ流入点 (流入境界) から連続する格子に 2 のフラグをつける⁵⁾。

3) JGD2011 における平面直角座標系の測地系の地形データを入力データに用いた場合, 「経度, 緯度」の列も平面直角座標となる。JDG2011 における緯度経度の測地系の地形データを入力データに用いた場合, 「座標」, 「経度, 緯度」双方とも適切な値が記録される。緯度経度から平面直角座標系の座標への変換には, hubey の式という簡易式を使用した。GIS などによる可視化には「経度, 緯度」の座標値を用いるとよい。

4) 第 2 章 計算事例の脚注参照

5) 第 1 章の計算事例に示した結果は, 1cm の粒径に対する限界掃流力を上回る掃流力が作用したことを示す 2 のフラグを有する格子を可視化したものである。

5.5 データ構造

本節では、高橋ほか (2018) による計算速度の向上のために工夫したデータ構造の設計について述べる。5.4 節で述べた格子番号取得操作の詳細である。並列化することを想定し、格子を走査中に条件分岐が生じないように工夫している。前述のサブルーチン `wetdry` において、並列化処理に適した形で流動深の存在域 (水域) を抽出するアルゴリズムについても略述する。

5.5.1 従来の附番記法

図 5.3 に示すように、従来附番の方法では、矩形の計算領域に対し構造格子を構築し、行列それぞれの方向に (i, j) の番号を付ける。スタガード格子により半格子ずれた格子点は文章中に記す際には隣接する (i, j) に $1/2$ を加えて表現する (例えば、高橋ほか 1986)。一方、プログラミング言語でソースコードを書くときは、整数で番号を表記しなければならないため、図 5.3 の括弧に示したように、格子点の番号と、半格子前に位置する格子辺の番号が一致する。各方向のベクトル評価点の数は、スカラー評価点の数より多い。

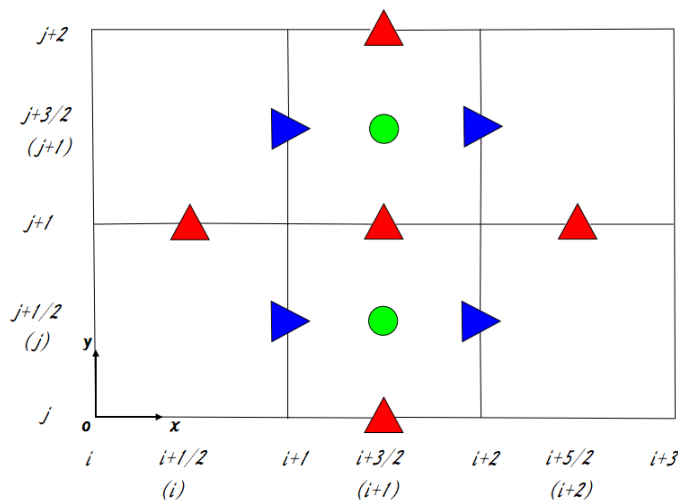


図 5.3 従来記法

5.5.2 本資料の附番記法

本節では、全体の格子点番号から、計算評価点 P に設定する水域の格子点 g あるいは格子辺 xc または yc へのアクセス方法と、その評価点周囲の格子点であるステンシルへのアクセス方法を述べる。図 5.1 に示した通り、格子点 $grid(g)$ 、格子辺 $edge(e)$ あるいは X と Y 方向を区別してそれぞれ x -surface(xs)、 y -surface(ys) と呼ぶことにする ($surface$ は検査体積の表面というイメージ)。

ここに述べるデータ構造の具体的な作成方法は、次節「地形データ変換と隣接リストの作成」で述べる。

5.5.2.1 計算領域における格子番号の全体像

計算領域において、格子点を $n = 1, \dots, N$ とし、格子辺 (x 方向ベクトル, y 方向ベクトル) にもそれぞれ通し番号を振り、周囲に仮想格子を設ける (図 5.4)。

計算領域全体の任意の格子番号から、評価点 g , xc , yc にアクセスし、その評価点に関するステンシルの格子点へアクセスできれば、差分方程式を解くことができる。この接続関係を配列によるグラフ形式のデータ構造として表現した。

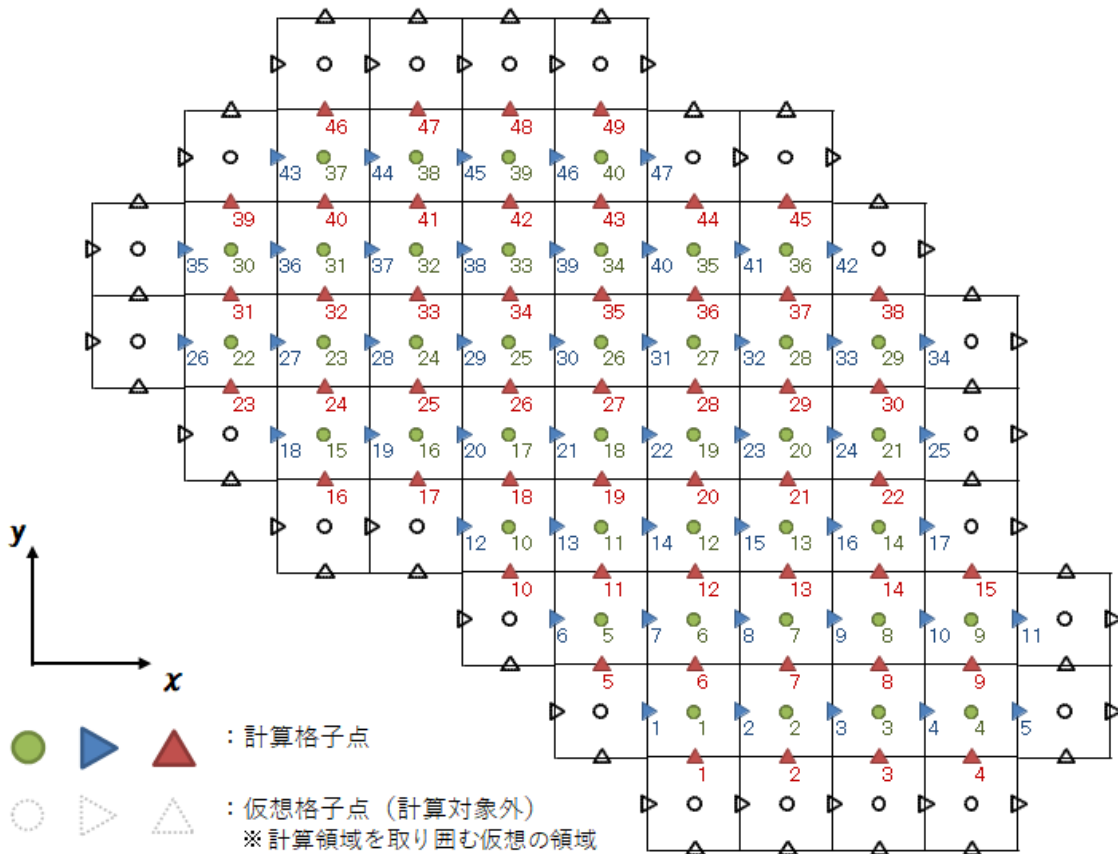


図 5.4 計算領域の格子番号附番の概念図

5.5.3 格子に関する用語

データ構造を説明するために、本資料で用いる格子や離散点に関する各種用語を整理する。

- 格子：領域を分割したひとつひとつの領域のこと。セルともいう。差分法では本来離散点しか考えないが、スタガード格子であるため有限体積法にならって格子を1つの検査体積とみなす。
- 格子番号：計算領域における格子点や格子辺に付けた番号のこと
- 格子点：格子の中央の点あるいはその番号のこと
- 格子辺：格子の辺の点あるいはその番号のこと。スタガード格子により半格子ずれた格子点は、検査体積の表面、つまり格子の辺に位置する。X方向とY方向で異なる番号が振られる。
- 評価点：差分方程式を解く際に着目する点のことで前章でPと表記した点のこと。連続式では格子点のうち1点が対象となるためスカラー評価点とよび、運動方程式では格子辺のうち1点が対象となるためベクトル評価点とよぶことにする。
- 水域（格子）：流動深が閾値 ε より大きく差分方程式を解く対象格子あるいはその番号のこと
- 陸域（格子）：流動深が閾値 ε より小さく差分方程式を解く対象ではない格子あるいはその番号のこと
- 仮想格子：計算領域の外周に配置した格子のこと
- 隣接リスト：格子点と格子点，格子点と格子辺，格子辺と格子辺などのグラフ関係が記憶された配列のこと。次節で述べるように数種類の隣接リストを使って、評価点からステンシル内の格子点や格子辺にアクセスする。

5.5.4 格子番号の取得方法

5.5.4.1 格子番号から水域における評価点へのアクセス方法

領域全体の格子番号から、水域における評価点にアクセスする方法を示す。

ある時刻において、水域格子点 g の総数を N_{wg} (wg は wet grid) とする。水域格子の番号は、計算領域の左下からの連番とする (図 5.4)。そのうちのひとつ ig を領域全体の格子番号 i から取得したいときには、 $ig =$

$idxWg(i)$ (index Wet grid) の配列を用いる。

同様に、X 方向の水域格子辺 xs における総数を $Nwxs$ とする。計算領域の格子番号 i から水域格子辺 ixs を取得したい場合、 $ixs = idxWxs(i)$ (index Wet x-surface など) の配列を用いる。Y 方向も同様に $iys = idxWys(i)$ の配列を用いる⁶⁾。

5.5.4.2 評価点からステンシル内の格子点・格子辺へのアクセス方法

評価点 P は、連続式を評価するときには g 、運動方程式を評価するには xs または ys である。それぞれの場合について、ステンシル内の格子点番号の取得方法を示す。

スカラー評価点の場合 スカラー評価点 P は格子点 g である。ステンシルを図 5.5 に示す。

ステンシル内の格子点番号 (index) の取得には、「 P の格子番号 (現在位置)」、「隣接方向」と「隣接格子点の種別」の 3 つの情報が必要である。そこで、「 i [隣接格子点の種別] c (隣接方向, P の格子番号)」という名前の配列を用意する。冒頭の i は index を、末尾の c はステンシル内の接続 connectivity を表し、[隣接格子点の種別] は $[g, xs, ys]$ のいずれかである。「隣接格子点の種別」に応じて配列を用意するとすれば、次の配列があれば評価点 P に関するステンシルの格子番号を取得できる。

- P に 1 格子幅で隣接する格子点 g の格子番号は、 igc (隣接方向, P 格子番号) で取得する。 P から見た隣接方向は $[E, N, W, S]$ のいずれかである
- P に半格子幅で隣接する格子辺 xs または ys の格子番号は、 iec (隣接方向, P の格子番号) で取得する。ここに、 e は辺 (edge) を、 P から見た隣接方向は、X と Y 方向のそれぞれに対し $[e, w]$ のいずれか、 $[n, s]$ のいずれかである。

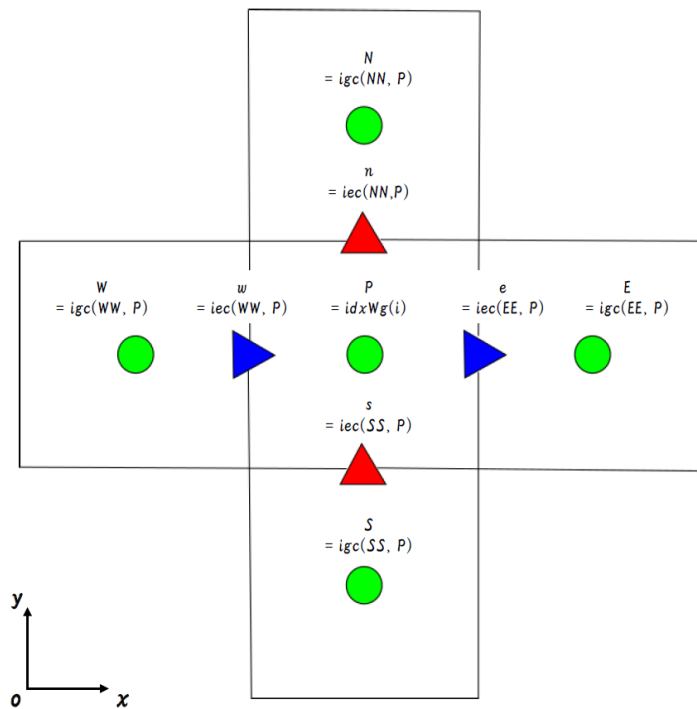


図 5.5 スカラ接続のステンシル

ベクトル評価点 (X 方向) の場合 ベクトル評価点 P (X 方向) は、格子辺 xs である。ステンシルを図 5.6 に示す。

ステンシル内の格子点の格子番号 (index) の取得には、「 P の格子番号 (現在位置)」、「隣接方向」、「隣接格子点の種別」の 3 つの情報が必要である。そこで、「 ixs [隣接格子点の種別] c (隣接する位置, P の格子番号)」という名前の配列を用意する。冒頭の i は index を、つぎの xs は x-surface を、末尾の c はステンシル内の接続 connectivity を表し、[隣接格子点の種別] は $[g, x, y]$ のいずれかである。「隣接格子点の種別」に応じて配列を用意するとすれば、次の配列があれば評価点 P に関するステンシルの格子番号を取得できる。

6) wetdry サブルーチンの 1.125 以降に記される。

- P に 1 格子幅で隣接する格子辺 x_s の格子番号は、 $ixsxc$ (隣接方向,P 格子番号) で取得する。ここに、[隣接格子点の種別] の x は x -surface を、P から見た隣接方向は [E,N,W,S] のいずれかである。
- P に斜め方向に隣接する格子辺 yc の格子番号は、 $ixsyc$ (隣接方向,P 格子番号) で取得する。ここに、[隣接格子点の種別] の y は y -surface を、P から見た隣接方向は [ne,nw,sw,se] のいずれかである。
- P に半格子幅で隣接する格子中央点 g の格子番号は、 $ixsgc$ (隣接方向,P 格子番号) で取得する。ここに、[隣接格子点の種別] の g は grid を、P から見た隣接方向は [e,w] のいずれかである。

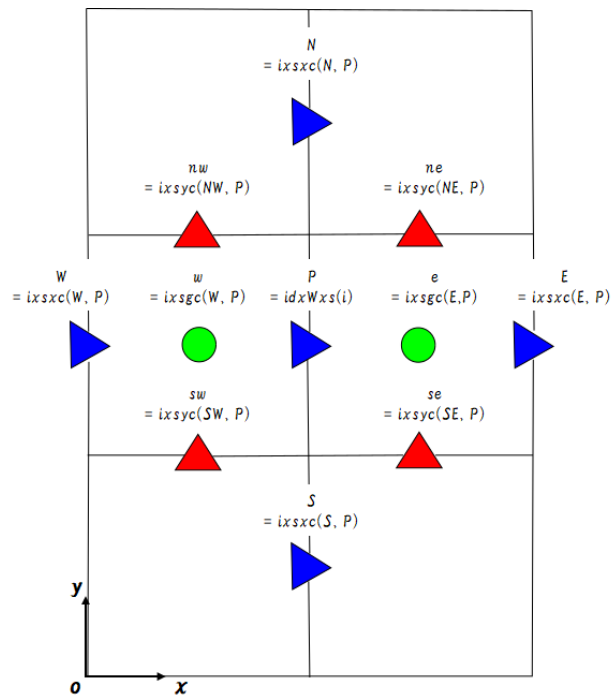


図 5.6 ベクトル接続 X 方向のステンシル

ベクトル評価点 (Y 方向) の場合 ベクトル評価点 P (Y 方向) は格子辺 ys である。ステンシルを図 5.7 に示す。

ステンシル内の格子番号 (index) の取得には、「P の格子番号 (現在位置)」、[隣接方向]、[隣接格子点の種別] が必要である。そこで、「 iys [隣接格子点の種別] c (隣接する位置, P の格子番号)」という名前の配列を用意する。冒頭の i は index を、つぎの ys は y -surface を、末尾の c はステンシル内の接続 connectivity を表し、[隣接格子点の種別] は [g, x, y] のいずれかである。「隣接格子点の種別」に応じて配列を用意するとすれば、次の配列があれば評価点 P に関するステンシルの格子番号を取得できる。

- P に 1 格子幅で隣接する格子辺 ys の格子番号は、 $iysyc$ (隣接方向,P 格子番号) で取得する。ここに、[隣接格子点の種別] の y は y -surface を、P から見た隣接方向は [E,N,W,S] のいずれかである。
- P に斜め方向に隣接する格子辺 xc の格子番号は、 $iysxc$ (隣接方向,P 格子番号) で取得する。ここに、[隣接格子点の種別] の x は x -surface を、P から見た隣接方向は [ne, nw, sw, se] のいずれかである。
- P に半格子幅で隣接する格子中央点 g の格子番号は、 $iysgc$ (隣接方向,P 格子番号) で取得する。ここに、[隣接格子点の種別] の g は grid を、P から見た隣接方向は [N,S] のいずれかである。

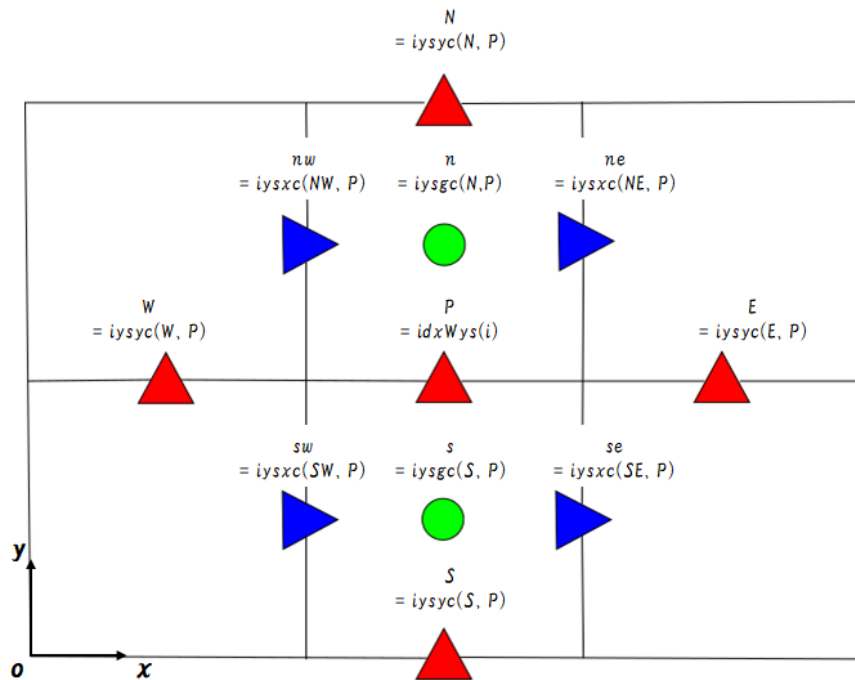


図 5.7 ベクトル接続 Y 方向のステンシル

5.6 地形データ変換と隣接リストの作成

入力データのうち、地形データ `dem.dat` と境界データ `boundary.dat` は、本資料に添付する地形データの事前処理プログラム `makeConnectiv.f90` (以下、地形変換プログラムという) で作成する。この地形変換プログラムによって、前節に示した高橋ほか (2018) のデータ構造を構築できる。

地形変換プログラムの入力ファイルは、GIS の一般的な形式の一つである AAIGrid 形式のデータである。GIS 等によって、例えば Geotiff 形式の地形データを AAIGrid 形式に変換できれば、本資料の土石流等氾濫解析プログラムの入力データを得ることができる。

5.6.1 地形変換プログラムの入出力データ

地形変換プログラム `makeConnectiv.f90` の入出力データは次の通りである。

- (input) `dem.asc` (名前固定; NODATA は-9999 とする)
- (output)
 - `dem.dat`: 土石流等氾濫解析プログラムの入力データで、位置および標高、隣接リストの情報を持つ。
 - `boundary.dat`: 土石流等氾濫解析プログラムの入力データで、境界の格子辺の番号と向きの情報を持つ。
 - `gridindex.asc`: 流入地点の格子点番号を取得するためのラスタデータ。本資料に添付する `getIndex.py` や GIS の画面上でハイドログラフ流入点の格子点番号を取得するために用いる。

5.6.2 格子番号の附番方法

AAIGrid 形式は、ヘッダーに格子の行列数・格子幅・NODATA とみなす標高値などのメタ情報を持つ、行列形式で格子点の標高が記録されたデータ形式をもつテキストファイルである。この入力地形データから、計算領域の格子点、格子辺に附番する方法を述べる。また、`dem.dat` あるいは `boundary.dat` として出力される地形データおよび隣接リストのデータ構造を述べる。

5.6.2.1 格子点の附番

計算領域全体におけるスカラー評価点となる格子点 `g` に附番する。なお、入力データである AAIGrid 形式では、領域が矩形で与えられる形式であるため、一部の領域を切り取った場合、計算対象外の標高を

NODATA(本資料では-9999)に設定する。よって、NODATA 部分(標高 0.0 未満)には格子番号を振らない。また、隣接する格子辺に 1 のフラグを付ける。

```
! Numbering grid
Ngrid = 0                !計算対象の格子点数の初期化
do j = 1, ncols
  do i = 1, nrows
    if(ezi(i, j) >= 0.d0) then !標高値が正であれば対象
      Ngrid = Ngrid + 1      !格子点数を加算
      numg(i, j) = Ngrid    !格子点番号を設定
      ix(i, j) = 1         !西側の X 方向格子辺を有効化
      ix(i, j+1) = 1       !東側の X 方向格子辺を有効化
      iy(i, j) = 1         !南側の Y 方向格子辺を有効化
      iy(i+1, j) = 1       !北側の Y 方向格子辺を有効化
    end if
  end do
end do
```

5.6.2.2 格子辺の附番

ベクトル評価点となる格子辺 x_s , y_s に附番する。

```
! Numbering edge
Nxs = 0                !計算対象 X 方向格子辺数ゼロクリア
Nys = 0                !計算対象 Y 方向格子辺数ゼロクリア
do j = 1, ncols + 1
  do i = 1, nrows
    if(ix(i, j) == 1) then !X 方向格子辺が有効
      Nxs = Nxs + 1        !X 方向格子辺数の加算
      numxe(i, j) = Nxs   !格子辺番号を設定
    end if
  end do
end do
do j = 1, ncols
  do i = 1, nrows + 1
    if(iy(i, j) == 1) then !Y 方向格子辺が有効
      Nys = Nys + 1        !Y 方向格子辺数の加算
      numye(i, j) = Nys   !格子辺番号を設定
    end if
  end do
end do
```

5.6.3 スカラー評価点の接続リスト構築

スカラー評価点 P の差分方程式の計算に必要なステンシルは、東西南北の格子点 (E, N, W, S), 東西の X 方向格子辺 (e, w), 南北の Y 方向格子辺 (s, n) である (図 5.8)。

配列の変数名には前節と同じ記号を用いるが、以下において計算領域の全ての格子が対象であるため、水域を表す Wg が変数名につかない。

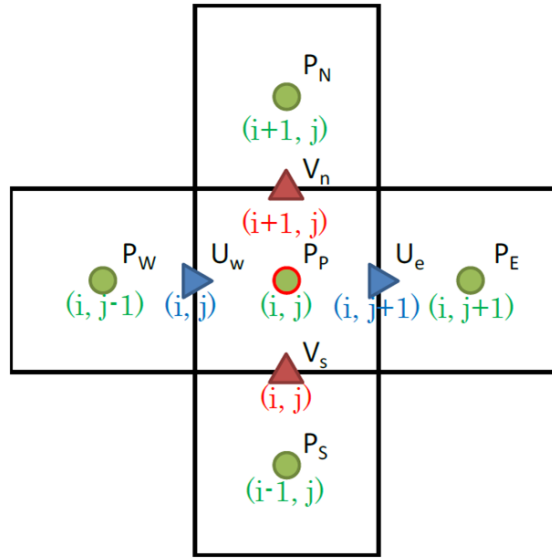


図 5.8 スカラー接続関係

- ステンシルの記号の定義 (inputdem.f90 より)

```
! scalar grid connectivity
!      +-----+           +-----+           +-----+
!      |       |           |       |           |       |
!      |   N   |           |   2   |           |  iNN  |
!      |       |           |       |           |       |
!      +-----+-----+ +-----+-----+ +-----+-----+
!      |       |       |       |       |       |       |       |
!      | W > P > E | | 3 3 P 1 1 | |iWW iWW P iEE iEE|
!      |       |       |       |       |       |       |       |
!      +-----+-----+ +-----+-----+ +-----+-----+
!      |       |           |       |           |       |
!      |   S   |           |   4   |           |  iSS  |
!      |       |           |       |           |       |
!      +-----+           +-----+           +-----+
! iEE = 1, iNN = 2, iWW = 3, iSS = 4
```

- アルゴリズム

```
do j = 1, ncols
do i = 1, nrows
  if(numg(i, j) > 0) then
    ig = numg(i, j)                !対象格子点 ig
    igc(iEE, ig) = numg(i, j+1)    !ig の東側格子点番号
    igc(iNN, ig) = numg(i+1, j)    !ig の北側格子点番号
    igc(iWW, ig) = numg(i, j-1)    !ig の西側格子点番号
    igc(iSS, ig) = numg(i-1, j)    !ig の南側格子点番号
    iec(iEE, ig) = numxe(i, j+1)   !ig の東側 X 方向格子辺番号
    iec(iNN, ig) = numye(i+1, j)   !ig の北側 Y 方向格子辺番号
    iec(iWW, ig) = numxe(i, j)     !ig の西側 X 方向格子辺番号
    iec(iSS, ig) = numye(i, j)     !ig の南側 Y 方向格子辺番号
    lonlat_g(:, ig) = lonlat(:, i, j) !ig の緯度経度
    xy_g(:, ig) = xy(:, i, j)      !ig の XY 座標
```

```

ezi_g(ig)      = ezi(i, j)      !ig の標高
hh0_g(ig)      = hh0(i, j)     !ig の水深
end if
end do
end do

```

5.6.4 ベクトル評価点 (X 方向) の接続リスト構築

X 方向ベクトル評価点 P における差分方程式の計算に必要なステンシルは、東西南北の X 方向格子辺 (E, N, W, S), 北東・北西・南東・南西の Y 方向格子辺 (ne, nw, se, sw), 東西の格子点 (e, w) である (図 5.9)。

配列の変数名には前節と同じ記号を用いるが、以下において計算領域の全ての格子が対象であるため、水域を表す Wg が変数名につかない。

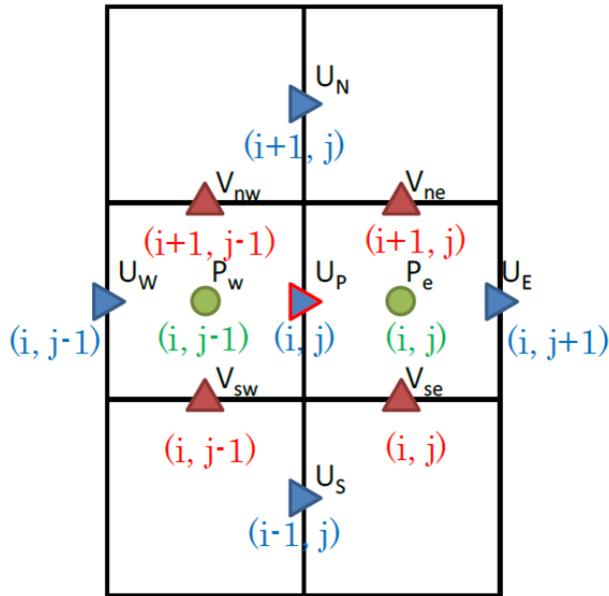


図 5.9 ベクトル x 接続関係

- ステンシルの記号の定義 (inputdem.f90 より)

```

! vector edge connectivity (x-direction)
!      +-----+-----+      +-----+-----+      +-----+-----+
!      |       |       |       |       |       |       |       |
!      |   >   |       |       |       |       |       |       |
!      |       |       |       |       |       |       |       |
!      +---^---+---^---+      +---2---+---1---+      +-iNW+-iNE-+
!      |       |       |       |       |       |       |       |
!      > W P E >      3 2 P 1 1 iWW iWg P iEg iEE
!      |       |       |       |       |       |       |       |
!      +---^---+---^---+      +---3---+---4---+      +-iSW+-iSE-+
!      |       |       |       |       |       |       |       |
!      |   >   |       |       |       |       |       |       |
!      |       |       |       |       |       |       |       |
!      +-----+-----+      +-----+-----+      +-----+-----+
! iEE = 1, iNN = 2, iWW = 3, iSS = 4
! iNE = 1, iNW = 2, iSW = 3, iSE = 4
! iEg = 1, iNg = 1, iWg = 2, iSg = 2

```

- アルゴリズム

```

do j = 1, ncols + 1
do i = 1, nrows
  if(ixe(i, j) == 1) then
    is = numxe(i, j)                !X方向の対象格子辺 is
    ixsrc(iEE, is) = numxe(i, j+1) !isの東側X方向格子辺番号
    ixsrc(iNN, is) = numxe(i+1, j) !isの北側X方向格子辺番号
    ixsrc(iWW, is) = numxe(i, j-1) !isの西側X方向格子辺番号
    ixsrc(iSS, is) = numxe(i-1, j) !isの南側X方向格子辺番号
    ixsrc(iNE, is) = numye(i+1, j) !isの北東側Y方向格子辺番号
    ixsrc(iNW, is) = numye(i+1, j-1) !isの北西側Y方向格子辺番号
    ixsrc(iSW, is) = numye(i, j-1) !isの南西側Y方向格子辺番号
    ixsrc(iSE, is) = numye(i, j) !isの南東側Y方向格子辺番号
    ixsgc(iEg, is) = numg(i, j) !isの東側格子点番号
    ixsgc(iWg, is) = numg(i, j-1) !isの西側格子点番号
  end if
end do
end do

```

5.6.5 ベクトル評価点 (Y 方向) の隣接リスト構築

Y 方向ベクトル評価点 P における差分式の計算に必要な情報は、東西南北の Y 方向ベクトル評価点 (E, W, S, N), 北東・北西・南東・南西の X 方向ベクトル評価点 (ne, nw, se, sw), 南北のスカラー評価点 (s, n) である (図 5.10)。

配列の変数名には前節と同じ記号を用いるが、以下において計算領域の全ての格子が対象であるため、水域を表す Wg が変数名につかない。

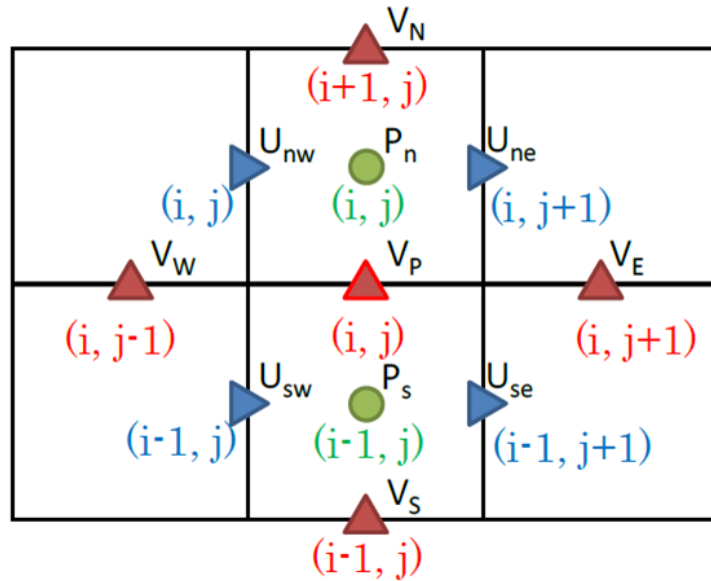


図 5.10 ベクトル y 接続関係

- ステンシルの記号の定義 (inputdem.f90 より)

```

! vector edge connectivity (y-direction)
! +-----+-----+-----+ +-----+-----+-----+ +-----+-----+-----+
! | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
! | > N > | | 2 1 1 | | iNW iNg iNE |
! | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
! +---+---+---+P+---+---+ +---3+---+P+---+1+---+ +---iEE+---+P+---+iEE+---+

```



```

! | | | | | | | | | | | | |
! | > S > | | 3 2 4 | | iSW iSg iSE |
! | | | | | | | | | | | | |
! +-----+-----+ +-----+4+-----+ +-----+iSS+-----+
! iEE = 1, iNN = 2, iWW = 3, iSS = 4
! iNE = 1, iNW = 2, iSW = 3, iSE = 4
! iEg = 1, iNg = 1, iWg = 2, iSg = 2

```

• アルゴリズム

```

do j = 1, ncols
  do i = 1, nrows + 1
    if(iye(i, j) == 1) then
      is = numye(i, j) !Y 方向格子辺 is
      iysyc(iEE, is) = numye(i, j+1) !is の東側 Y 方向格子辺番号
      iysyc(iNN, is) = numye(i+1, j) !is の北側 Y 方向格子辺番号
      iysyc(iWW, is) = numye(i, j-1) !is の西側 Y 方向格子辺番号
      iysyc(iSS, is) = numye(i-1, j) !is の南側 Y 方向格子辺番号
      iysxc(iNE, is) = numxe(i, j+1) !is の北東側 X 方向格子辺番号
      iysxc(iNW, is) = numxe(i, j) !is の北西側 X 方向格子辺番号
      iysxc(iSW, is) = numxe(i-1, j) !is の南西側 X 方向格子辺番号
      iysxc(iSE, is) = numxe(i-1, j+1) !is の南東側 X 方向格子辺番号
      iysgc(iNg, is) = numg(i, j) !is の北側格子点番号
      iysgc(iSg, is) = numg(i-1, j) !is の南側格子点番号
    end if
  end do
end do

```

5.6.6 流出境界の設定

boundary.dat として出力される流出境界のデータ構造について述べる。

計算領域縁は格子辺である。その点で流出境界を初期設定する。設定方法は3段階からなり、1) 流出境界の格子辺数の数え上げ、2) X 方向の流出境界の設定、3) Y 方向の流出境界の設定、である。

5.6.6.1 1) 流出境界の格子辺数の数え上げ

```

nxout = 0 !X 方向流出境界数の初期化
nyout = 0 !Y 方向流出境界数の初期化
do i = 1, Nxs
  if (ixsgc(iEg, i) == 0) then !i 番 X 方向格子辺の東側格子点が領域外
    nxout = nxout + 1 !X 方向流出境界数の加算
  end if
  if (ixsgc(iWg, i) == 0) then !i 番 X 方向格子辺の西側格子点が領域外
    nxout = nxout + 1 !X 方向流出境界数の加算
  end if
end do
do i = 1, Nys
  if (iysgc(iNg, i) == 0) then !i 番 Y 方向格子辺の北側格子点が領域外
    nyout = nyout + 1 !Y 方向流出境界数の加算
  end if
  if (iysgc(iSg, i) == 0) then !i 番 Y 方向格子辺の南側格子点が領域外
    nyout = nyout + 1 !Y 方向流出境界数の加算
  end if
end do

```

```
end do
```

5.6.6.2 2) X方向の流出境界の設定

```
if(nxout /= 0) then                                !流出境界番号, 流出方向
  allocate(ixout(nxout), ixdir(nxout))
  nxout = 0
  ixout(:) = 0
  ixdir(:) = ""
  do i = 1, Nxs
    if (ixsgc(iEg, i) == 0) then                  !i 番 X 方向格子辺の東側格子点が領域外
      nxout = nxout + 1                          !nxout 番目の流出境界番号の設定
      ixout(nxout) = i                          !流出の評価点番号を記憶
      ixdir(nxout) = "+"                        !流出方向を記憶
    end if
    if (ixsgc(iWg, i) == 0) then                  !i 番 X 方向格子辺の西側格子点が領域外
      nxout = nxout + 1                          !nxout 番目の流出境界番号の設定
      ixout(nxout) = i                          !流出の格子辺番号を記憶
      ixdir(nxout) = "-"                        !流出方向を記憶
    end if
  end do
end if
```

5.6.6.3 3) Y方向の流出境界の設定

```
if(nyout /= 0) then
  allocate(iyout(nyout), iydir(nyout)) !流出境界番号, 流出方向
  nyout = 0
  iyout(:) = 0
  iydir(:) = ""
  do i = 1, Nys
    if (iysgc(iNg, i) == 0) then                  !i 番 Y 方向格子辺の北側格子点が領域外
      nyout = nyout + 1                          !nyout 番目の流出境界番号の設定
      iyout(nyout) = i                          !流出の評価点番号を記憶
      iydir(nyout) = "+"                        !流出方向を記憶
    end if
    if (iysgc(iSg, i) == 0) then                  !i 番 Y 方向格子辺の南側格子点が領域外
      nyout = nyout + 1                          !nyout 番目の流出境界番号の設定
      iyout(nyout) = i                          !流出の格子辺番号を記憶
      iydir(nyout) = "-"                        !流出方向を記憶
    end if
  end do
end if
```

5.7 入力データの書式

本節では、入力データのファイル書式について述べる。

入力データの種類別に、テキストファイルの書式を解説した図、例、入力部分のソースコードの抜粋を順番に示す。ソースコードは正確な入力書式を把握できるようにするために示すものである。これらの項目のいくつかは、入力データの種類によっては省略する。

5.7.1 入出力ファイル名指定ファイル filename.dat の書式

このファイルでは、入力ファイル名や出力ファイル名を指定する。このファイルは、サブルーチン `flopen` で読み込まれる。filename.dat という名前は固定である（ソースコードを書き換えられない限り変更不可）。

5.7.1.1 書式図

1 行あたりの文字列の長さは 50 文字までである。書式を図 5.11 に示す。

行数	カラムNo.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
	コメント	open(10,file='filename.dat',status='old')																													← flopen.f90 1.7で定義された固定ファイル名															
1	データ (1)	a30																																												
	コメント	地形データファイル名 (11)																													← makeConnectiv.f90で自動生成															
2	データ (2)	a30																																												
	コメント	境界条件データファイル名 (20)																													← makeConnectiv.f90で自動生成															
3	データ (3)	a30																																												
	コメント	時間データファイル名 (22)																																												
4	データ (4)	a30																																												
	コメント	流量データファイル名 (12)																																												
5	データ (5)	a30																																												
	コメント	計算パラメータファイル名 (13)																																												
6	データ (6)	a30																																												
	コメント	開始時及び逐次計算時の出力csvファイル名 (31)																													← CSV形式で出力															
7	データ (7)	a30																													拡張子(.csv)前に繰返し回数が挿入															
	コメント	計算終了時の出力csvファイル名 (32)																													← CSV形式で出力															
8	データ (8)	a30																																												
	コメント	計算時間における最大値の出力csvファイル名 (33)																													← CSV形式で出力															
9	データ (9)	a30																																												
	コメント	入値の確認用のテキストファイル名 (34)																													← テキスト形式で出力															
10	データ (9)	a30																																												
	コメント	流入点から連続する砂礫が動き得る範囲を示すテキストファイル名 (99)																													← テキスト形式で出力															
11	データ (9)	a30																																												
	コメント	計算終了までに流域外に流出する総流量を記録するテキストファイル名 (80)																													← テキスト形式で出力															
12	データ (9)	a30																																												

図 5.11 filename.dat の書式

5.7.1.2 例

第 2 章計算事例における雲仙普賢岳計算事例 a で用いた filename.dat を例として以下に示す。

汎用解析プログラムの実行ファイルが存在するフォルダから相対パスでファイル名を記す。記載した入力ファイル名のファイルが所定の場所に存在しなければプログラム実行時にエラーとなる。出力ファイルはプログラム実行後に自動的に生成される。ただし、./output/のようなフォルダに保存したい場合、実行前にあらかじめフォルダを作っておかなければならない（パスの区切り文字は処理系に合わせて）。また、パスに日本語があると実行できない場合がある。

```
.\dem.dat
.\boundary.dat
.\time.dat
.\hydr.dat
.\param.dat
.\output\map_.csv
.\output\end.csv
.\output\max.csv
.\output\inputdata.dat
.\output\tau_strc.csv
.\output\outflow_volume.dat
```

5.7.2 地形データ (dem.dat) の書式

地形変換プログラム `makeConnectiv.f90` によって生成される地形データ (dem.dat) の書式について述べる。このファイルは、サブルーチン `inputdem` で読み込まれる。自動的に作成されるためテキストの中身の解説は省略し、地形データを読み込む Fortran ソースコードの該当部分のみ示す。

5.7.2.1 Fortran ソースコード

ファイルを読み込む Fortran ソースコード断片を示す。

```
read (11, '(3i10 ,2E25.15,i10)' )Ngrid,Nxs,Nys,dx,dy,idz
do i=0,Ngrid
  read(11,'(6(E25.15))')xy(1:2,i),lonlat(1:2,i),ezi(i),hn0(i)
end do
do i=0,Ngrid
  read(11,'(8i10)')igc(1:4,i),iec(1:4,i)
end do
do i=0,Nxs
  read(11,'(10i10)')ixsxc(1:4,i),ixsyc(1:4,i),ixsgc(1:2,i)
end do
do i=0,Nys
  read(11,'(10i10)')iysxc(1:4,i),iysyc(1:4,i),iysgc(1:2,i)
end do
```

5.7.3 境界データ (boundary.dat) の書式

地形変換プログラム makeConnevtiv.f90 によって生成される境界データ (boundary.dat) の書式について述べる。このファイルは、サブルーチン inputdem で読み込まれる。自動的に作成されるためテキストの中身の解説は省略し、境界データを読み込む Fortran ソースコードの該当部分のみ示す。

5.7.3.1 Fortran ソースコード

ファイルを読み込む Fortran ソースコード断片を示す。

```
read(20,'(2i10)') nxout, nyout
do i=1,nxout
  read(20,'(i10,a)')ixout(i),outDir(i)
end do
do i=1,nyout
  read(20,'(i10,a)') iyout(i), outDir(i)
end do
read(20,'(2i10)') nxwall, nywall
do i=1,nxwall
  read(20,'(i10)') ixwall(i)
end do
do i=1,nywall
  read(20,'(i10)')iywall(i)
end do
```

5.7.4 総時間・時間ステップ入力データ (time.dat) の書式

このファイルは、逐次計算の出力間隔 (iout)、区分毎の計算時間、時間ステップ Δt を設定するファイルである。サブルーチン inputhydr で読み込まれる。

5.7.4.1 書式図

総時間・時間ステップ入力データを図 5.12 に示す。

カラムNo.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38					
行数	時間入力データのフォーマット																																										
1	コ	t	i	m	e	-				2	0	0																															
	コ	任意の文字列(a5)					データ数mf (i5)					←データ数は他のファイル一致されることが必要																															
2	コ	継続時間 (s) (f10.0)										時間刻みdt1 (s) (f10.0)																															
3	コ	6 0 0 . 0										0 . 0										←11-20列目で与えるdtは 離散化式上のdt 連続式と運動方程式は2dtで一式解かれる																					
	コ	データ数分の行が並ぶ																																									
203		6 0 0 . 0										0 . 0																															

図 5.12 総時間・時間ステップ設定ファイルの書式

5.7.4.2 例

第2章 計算事例で示した雲仙普賢岳計算事例 a よりファイル内容例を示す。

```
time- 200
      0.0      0.1
      600.0    0.1
      600.0    0.1
      600.0    0.1
      (略)
      600.0    0.1
      600.0    0.1
      600.0    0.1
```

5.7.4.3 Fortran ソースコード

ファイルを読み込む Fortran ソースコード断片を示す。

```
read(22, '(5x, i5, i15)') mlf, iout
do i = 1, mlf
  read(22, '(4f10.0)') a0, a1, a2
  tdl1(i) = a0
  tdt1(i) = a1
end do
```

5.7.5 ハイドログラフ入力データ (hydr.dat) の書式

書式図, ファイル内容例, Fortran ソースコード断片およびハイドログラフの書式における数字の意味について述べる。このファイルは, サブルーチン inpuhydr で読み込む。

5.7.5.1 書式図

ハイドログラフ入力データの書式を図 5.13 に示す。

カラムNo.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	
行数	ハイドログラフ入力データのフォーマット																																									
1	コ	h	y	d	r	1				1																																
	コ	任意の文字列(a5)					ハイドロ数 (i5)					データ数 (i5)					流入点河床勾配sqin (f10.0)																									
2	コ	0 . 0										0 . 0										0 . 0										0 . 0										
3	コ	継続時間 (s) (f10.4)										流量 (m3/s) (f10.4)										粗粒分土砂濃度 (m3/s) (f10.4)										細粒分土砂濃度 (m3/s) (f10.4)										
	コ	データ数分の行が並ぶ																																								
203		6 0 0 . 0										0 . 0										0 . 0										0 . 0										
204	コ	* e n d *					← 流量パターン入力の終了を示す。 複数ハイドロの場合は最後のハイドロの末尾に記す。																																			
		(A5)																																								

図 5.13 ハイドログラフの入力データ書式

5.7.5.2 ファイル内容例

2章計算事例における雲仙普賢岳計算事例のケース a を基にファイル内容例を示す。末尾には*end*が必須であることに注意されたい。

3列目の粗粒分砂礫の土砂濃度、4列目の細粒分砂礫の土砂濃度を設定すると、流砂の計算を行うか判定する状態変数 noSediment が.false. となり、流砂計算が実行される。なお、細粒分砂礫の土砂濃度は、間隙流体の体積 $1 - C_L$ における体積分率であることに注意。

```
hydr1    1  200      0.13
  600.000    0.000    0.000   -1.000
  600.000    0.065    0.000   -1.000
  600.000    0.138    0.000   -1.000
(略)
  600.000    0.000    0.000   -1.000
  600.000    0.000    0.000   -1.000
  600.000    0.000    0.000   -1.000
```

end

なお、複数のハイドログラフ流入点においてハイドログラフを与える場合、以下の例における*end*の行に、hydr1 と同様の書式のヘッダーと 2 行目の書式のデータを挿入する。ヘッダーの名前 (hydr1 の部分) は異なる名称を与えなければならない。このとき、param.dat に 2 つ目のハイドログラフの名称と流入境界の格子番号を与える必要がある (図 5.16 参照)。2章 計算事例における雲仙普賢岳計算事例のケース b の例を示す。

```
hydr1    1  200      0.13
  600.000    0.000    0.000   -1.000
  600.000    0.065    0.000   -1.000
  600.000    0.138    0.000   -1.000
(略)
  600.000    0.000    0.000   -1.000
  600.000    0.000    0.000   -1.000
  600.000    0.000    0.000   -1.000
```

```
hydr2    2  200      0.17
  600.000    0.000    0.000   -1.000
  600.000    0.012    0.000   -1.000
  600.000    0.076    0.000   -1.000
(略)
  600.000    0.000    0.000   -1.000
  600.000    0.000    0.000   -1.000
  600.000    0.000    0.000   -1.000
```

end

5.7.5.3 Fortran ソースコード

ファイルを読み込む Fortran ソースコード断片を示す。

```
do
  read(12, '(a5, 2i5, f10.2)', iostat=is) mw, i0, i1, s0
  if(is < 0 .or. mw == '*end*') return
  ! check hydr no
  if(mpat /= i0) call errnum(' inputhydr-01')
  ! check data no in each hydr
  if(mlf /= i1) call errnum(' inputhydr-02')
  ! check slope of source point
  if(s0 <= 0.d0) call errnum(' inputhydr-03')
  do i = 1, i1
    read(12, '(5f10.4)') t0, a1, a2, a3
```

```

write(34, '(5f10.4)') t0, a1, a2, a3
qin(mpat, i) = a1
cli(mpat, i) = a2
! this is not used in this program
cfi(mpat, i) = a3
! this is not used in this program
end do
end do

```

ハイドログラフの書式における数字の意味 ハイドログラフファイルの書式は、2 行目以降に、任意の時間長（単位は秒）とその間に入力したい一定流量 m^3/s を記す。

以下に実際の観測データと入力書式の作り方のイメージを例示する。

【例】 時刻と流量のデータが以下のように得られているとする。

$(t, Q) = \{(0, 10), (100, 30), (150, 50), (300, 20), (500, 0)\}$

本プログラムのハイドログラフの書式は、次のように、時間長 $(t_{i+1} - t_i)$ と、その間の平均流量 $((Q_{i+1} + Q_i)/2)$ を与える。

dt	q
100	20
50	40
150	35
200	10

図 5.14 に、観測ハイドログラフ（青線）と入力書式（赤線）の違いを示す。ハイドログラフの波形変化がある場合は時間長さを細かく刻み、波形変化がなく一定値が入力される場合は時間長さを長くするとよい。また、ピーク流量が平滑化されるため、ピークをしっかりと入力して計算したい場合には、その部分の時間長を細かく刻むなどの工夫が必要となる。

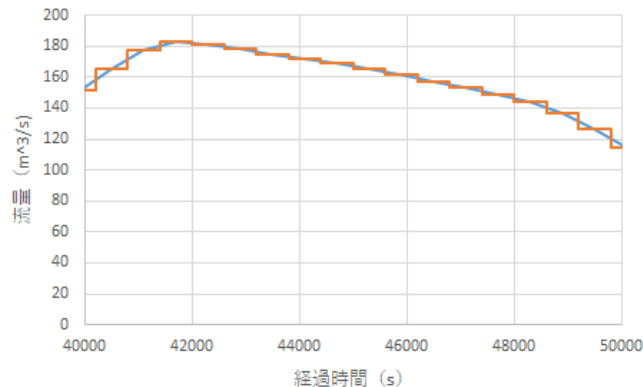


図 5.14 ハイドログラフの入力データにおける数字の意味

5.7.6 パラメータ入力ファイル (param.dat) の書式

パラメータ入力ファイル (param.dat) の書式について、書式図、ファイル内容例、Fortran ソースコードの要約を述べる。このファイルは、サブルーチン inputparam で読み込む。

5.7.6.1 書式図

パラメータ入力ファイルの書式図を図 5.15 に示す。18 行目の格子番号は makeConnvtiv.f90 の出力ファイル gridindex.asc を用いて探索する。同行における流下方向の方位は、x+y (東), x-y (西), x y+ (北), x y- (南), x+y+ (北東), x+y- (南東), x-y+ (北西), x-y- (南西) の 8 方位のいずれかを記入する。

行号	カラムNo.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43						
	コメント	パラメータ入力ファイルのフォーマット																																																
1	コメント	p	a	r	a	m	e	t	e	r																																								
	コメント	タイトル(任意の文字列可) (a10)																						← 名称																										
2	コメント	d	i	a	m	e	r	=	0.	0	1	0	0																																					
	コメント	任意の文字列(a10)										dm (f10.0)										← 粗粒土砂の代表粒径(m) dL																												
3	コメント	s	i	g	m	a	=	2	6	0	0	0																																						
	コメント	任意の文字列(a10)										sig (f10.0)										← 砂礫の密度(kg/m3) σ																												
4	コメント	r	h	o	=	1	0	0	0	0																																								
	コメント	任意の文字列(a10)										rho (f10.0)										← 水の密度(kg/m3) ρ																												
5	コメント	m	a	n	n	i	n	g	=	0.	0	4	0	0																																				
	コメント	任意の文字列(a10)										sodo (f10.0)										← マニング粗度係数 n																												
6	コメント	t	a	n	f	r	i	c	=	0.	7	0	0	0																																				
	コメント	任意の文字列(a10)										tanp (f10.0)										← 内部摩擦角 ϕ の正接($\phi=35^\circ$ で $\tan\phi=0.70$)																												
7	コメント	C	s	t	a	r	=	0.	6	0	0	0																																						
	コメント	任意の文字列(a10)										cst (f10.0)										← 河床堆積層の充填土砂濃度 C*																												
8	コメント	C	s	t	a	r	_	f	=	0.	2	0	0	0																																				
	コメント	任意の文字列(a10)										cstf (f10.0)										← 河床堆積層の細粒分砂礫充填土砂濃度 C*F																												
9	コメント	C	s	c	=	0.	0	2	0	0																																								
	コメント	任意の文字列(a10)										csc (f10.0)										← 掃流状集合流動と掃流砂流の形態境界の土砂濃度 Csc																												
10	コメント	t	a	u	c	r	i	t	=	0.	0	5	0	0																																				
	コメント	任意の文字列(a10)										tauc (f10.0)										← 無次元限界掃流力 τ^*																												
11	コメント	d	e	l	t	a	_	e	=	0.	0	0	0	0	7																																			
	コメント	任意の文字列(a10)										bit (f10.0)										← 侵食速度係数 δ_e																												
12	コメント	d	e	l	t	a	_	d	=	0.	0	0	1	0																																				
	コメント	任意の文字列(a10)										bit d (f10.0)										← 堆積速度係数 δ_d																												
13	コメント	e	p	s	i	l	i	o	n	=	0.	0	0	1	0																																			
	コメント	任意の文字列(a10)										th (f10.0)										← 水域と陸域の判定水深(m) threshold																												
13	コメント	b	e	t	a	=	1.	2	5	0	0																																							
	コメント	任意の文字列(a10)										th (f10.0)										← 土石流の運動量補正係数 β																												
14	コメント	f	i	g	_	r	e	b	e	d	=	0																																						
	コメント	任意の文字列(a10)										flg.rebed (i10)										← 粗粒分砂礫の連続式の過剰土砂を河床高に加える場合1																												
15	コメント	i	o	u	t	=	1																																											
	コメント	任意の文字列(a10)										iout (i10)										← 時系列データの出力をする場合には1																												
16	コメント	h	y	d	o	-	i	n																																										
	コメント	任意の文字列(a10)																																																
17	コメント	n	o	.	1																																													
	コメント	任意の文字列(a5)					nbc (i5)					← 流入地点数 Number of Boundary Condition																																						
18	コメント	h	y	d	r	1	1	5	5	6	x	y	-																																					
	コメント	(a5)					ibg (i10)					2x		a1 1x a1					← (a5)は「流量データファイル」の冒頭文字列と同一 以降の行は、18行目と同じ書式で、nbcの数だけ入力データを記入 ibgはグリッド番号、aiはX、Y方向の記号(+ or -)																															

図 5.15 パラメータファイルの書式

5.7.6.2 例

第 2 章 計算事例における雲仙普賢岳計算事例 a で使用した例を示す。

運動量補正係数 beta は、土石流の流砂形態時に移流項に適用され、1.25 か 1.0 を設定する。

```
parameter
diameter= 0.0100
  sigma= 2600.0
  rho= 1000.0
manning= 0.0400
tan_fric= 0.7000
  Cstar= 0.6000
  Cstar_f= 0.2000
  Csc= 0.0200
tau_crit= 0.0500
  delta= 0.0007
  delta_d= 0.0010
  epsilon= 0.0010
  beta= 1.2500
flg_rebed= 0
  iout= 0

hydo-in
no. 1
hydr1 112 x+y
```


5.7.6.3 Fortran ソースコードの要約

ファイルを読み込む Fortran ソースコード断片を示す。

```
read(13,'(a10,i5)') mj
read(13,'(a10,f10.0)') mw_10, dm
read(13,'(a10,f10.0)') mw_10, sig
read(13,'(a10,f10.0)') mw_10, rho
read(13,'(a10,f10.0)') mw_10, sodo
read(13,'(a10,f10.0)') mw_10, tanp
read(13,'(a10,f10.0)') mw_10, cst
read(13,'(a10,f10.0)') mw_10, cstf
read(13,'(a10,f10.0)') mw_10, csc
read(13,'(a10,f10.0)') mw_10, tauc
read(13,'(a10,f10.0)') mw_10, bit
read(13,'(a10,f10.0)') mw_10, bit_d
read(13,'(a10,f10.0)') mw_10, th
read(13,'(a10,f10.0)') mw_10, bm
read(13,'(a10,i10)') mw_10, flg_rebed
read(13,'(a10,i10)') mw_10, iout
read(13,'(a10)') mj
read(13,'(a5,i5)') mw, nbc
if (nbc > 0) then
  do i = 1, nbc
    read(13, '(a5,i10,2x,a1,1x,a1,f10.0)') mw, ibg(i), m1, m2, a0
    !c error-check
    if(ihyf(i) == 0) then
      write(*, *) &
        & ' 合致するハイドロ名がない: ', trim(mw)
      write(*, *) ' Cannot find the hydr name: ', trim(mw)
    end if
  end do
end if
```

5.7.7 入力ファイル間の整合

入力ファイル間には、図 5.16 に示すように、同じ数字や文字列にしなければエラーになり計算が実行されない部分がある。

1. 計算総時間と時間ステップの入力ファイル (time.dat) とハイドログラフの入力ファイル (hydr.dat) において、行数は同じにしなければならない。
2. hydr.dat と入力パラメータファイル (param.dat) におけるハイドロの名称 (5 文字) と入力箇所数

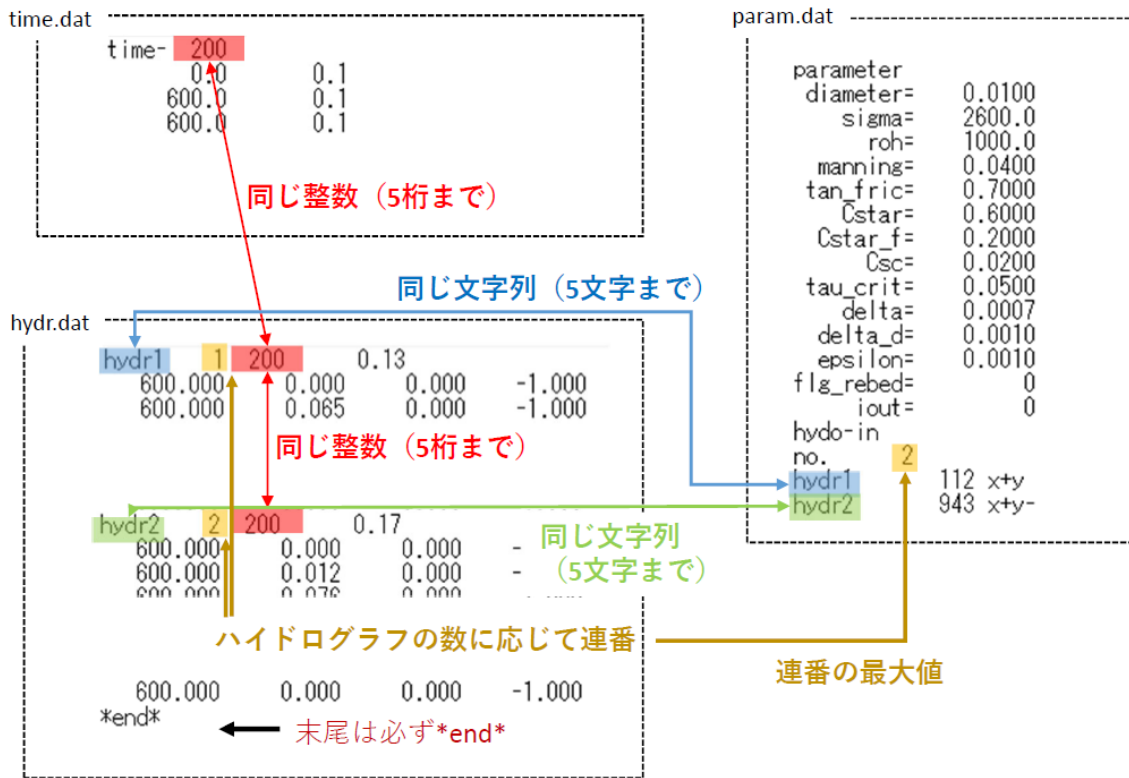


図 5.16 入力ファイル間の相互関係

5.8 コンパイル環境

筆者らは Intel Fortran compiler により並列化し、第 1 章の図 1.2 に示した計算速度を得た。

オープンソースや無償で個人利用可能なコンパイラ (Intel Fortran, GFortran) でコンパイルが可能である⁷⁾。並列化による実行速度の向上は Intel Fortran で確認した。

コンパイルは処理系によるが、`ifort *.f90 -o debrisflow.exe` や `gfortran *.f90 -o debrisflow.exe` で実行できる。並列化する場合は `ifort /Qopenmp *.f90 -o debrisflow.exe` や `gfortran -fopenmp *.f90 -o debrisflow.exe` などのコンパイラが備えているオプションを用いる。最適化の `-O3` オプションなども使うとよい。gfortran 用には `make all` でコンパイルできる Makefile (テキストファイル) を用意した。

5.9 処理例

本節では、2 章計算事例に示したケース 2 「古恵川」を例として、入力データ作成に関する前処理について述べる。

時間が経過すれば本節に示したコマンドラインプログラムは使用できなくなる可能性が高いが、処理の基本的な手順は参考となると考えられる。

5.9.1 処理の仕方

ここに示す処理は、コマンドプロンプト上でコマンドラインプログラムを用いて実行した。パス設定などの環境設定、各コマンドの使い方 (コマンドの意味、引数の意味など) は本資料では述べない。

5.9.2 実行環境

筆者の実行環境を示す。

- OS Microsoft Windows 10 Pro

7) 個人利用か商用利用かなどに応じて利用制限がかかる場合がある。利用に当たっては注意すること。

- Python 3.7.6 (default, Jan 8 2020, 20:23:39) [MSC v.1916 64 bit (AMD64)] :: Anaconda, Inc. on win32
- GDAL 3.1.4, released 2020/10/20
 - OSGeo4W のインストーラによるインストール, あるいは, Anaconda による `conda install gdal` あるいは `pip install gdal` などのパッケージ管理アプリを使ってコマンドライン上でインストールする。
 - `gdal_warp`, `ogrinfo` などがコマンドラインから利用できることを確認する。

5.9.3 準備

- 作業フォルダは, `./`で表すカレントディレクトリとする。
- ハイドログラフは, 降雨流出解析あるいは災害報告のハイドログラフなどを基に `./hydr.dat` の名称で保存する (書式は前述のとおり)。
- 総時間や時間刻みを設定する `./time.dat` は, `hydr.dat` をコピーして 2 列目を刻み時間に書き換えて作成する。
- 国土地理院基盤地図情報 (JPGIS 書式による XML ファイル) をダウンロードしておき, 脚注に示す `python script` など で Geotiff に変換・結合して `./dsm.tif` の名前で保存する⁸⁾。
- 地形データから計算領域を切り出すための KML をあらかじめ地理院地図で作成し `./clipper_11.kml` の名称で保存する。
- `dsm.tif` と `clipper_11.kml` の座標系は JGD2011 (EPSG 6668) (国土地理院による世界測地系 緯度経度法) とする。

5.9.4 入力データの地理座標系などメタデータの確認

`gdal*`はラスターデータ処理, `ogr*`はベクターデータ処理のプログラムである。`gdalinfo` はラスターデータのメタデータを, `ogrinfo` はベクターデータのメタデータを表示するコマンドである。

```
gdalinfo ./dsm.tif
```

```
ogrinfo ./clipper_11.kml
```

(結果は示さないが, 緯度経度法に同じ座標系に geocoding されていることを確認)

5.9.5 地理座標系の変換

- KML を平面直角座標系の地形データに合わせるものとする (計算格子幅をメートル単位で取得する必要があるため)

5.9.5.1 ファイル形式・座標変換

後述する切り出し処理で KML 形式を受付けないため, `ogr2ogr` によって, ファイル形式を `shapefile` に, また, 座標系を緯度経度から平面直角座標系: JGD2011 longlat (EPSG:6668) -> CS2 (EPSG:6670) に変換する。

```
ogr2ogr -f "ESRI Shapefile" ./clipper_cs.shp ./clipper_11.kml
```

```
-t_srs EPSG:6670 -s_srs EPSG:6668
```

5.9.5.2 layer 名を取得

```
ogrinfo ./clipper_cs.shp
```

→ `clipper_cs` を得る

5.9.5.3 layer の座標系などのメタデータの確認

```
ogrinfo ./10_output/clipper_cs.shp clipper_cs
```

8) 国土地理院基盤地図情報 (JPGIS 書式による XML ファイル) を用いる場合, Minoru Akagi 氏によるオープンソースソフトウェア `fgddem` により Geotiff 形式に変換できる。`fgddem` の URL (accessed on Feb. 2021): <https://github.com/minorua/fgddem.py>

5.9.6 地形データの切り出し

gdalwarp は、ラスターデータのフォーマット変換や地理座標変換を行うコマンドである。フラグの意味を簡潔に示す。

- -dstnodata は処理後の出力ラスターデータにおいて NoData に対応する数値
- -r は補間方法でここでは最近隣法
- -tr の2つの数字が処理後ラスターデータの X と Y 方向の格子幅で、ここでは 50m の正方形格子

```
gdalwarp -dstnodata -9999 -cutline ./clipper_cs.shp -r near
-crop_to_cutline -tr 50 50 ./dsm.tif
-of GTiff ./clipped_dsm.tif
```

5.9.6.1 (参考) 10m grid size の場合

```
gdalwarp -dstnodata -9999 -cutline clipper_cs.shp -r near
-crop_to_cutline -tr 10 10 dsm.tif -of AAIGrid dem.asc
```

5.9.7 JGD2011 longlat に座標系を付け替えて、AAIGrid 形式へ変換 (省略可)

gdal_translate はファイル形式を変換するコマンドである。

```
gdal_translate -a_nodata -9999 -r near ./10_output/clipped_dsm.tif
-of AAIGrid ./10_output/dem.asc
```

5.9.8 作成した地形データのメタデータ確認

```
gdalinfo ./clipped_dsm.tif
gdalinfo ./dem.asc
```

5.9.9 地形データの汎用解析入力形式への変換

```
makeConnectiv.exe
```

```
# input 2 by keyboard
```

- 次のような画面が表示される。

```
Converting 'dem.asc to 'dem.dat', 'boundary.dat' & 'gridindex.asc'
```

```
Select Coordinate System of input DEM.
```

- 1) Long-Lat CS
- 2) Japan Plane Rectangular CS
- 9) exit, no conversion

```
>>> 2
```

```
Bottom left:          -64930.21,          -31184.31
Center      :          -61455.21,          -28584.31
Top right   :          -57980.21,          -25984.31
```

```
Region size:    6950.00m x    5200.00m
```

```
Node:          6444    X-edge:          6576    Y-edge:          6596
```

```
dx:           50.00 m
```

```
dy:           50.00 m
```

```
Writing grid data...
Writing boundary conditions...
Writing grid index...
```

Conversion end.

- boundary.dat, dem.dat, gridindex.asc が得られる。

5.9.10 ハイドログラフ入力格子点番号の取得

5.9.10.1 準備

地理院地図を用いてハイドログラフ設定点の緯度経度を取得し、座標変換を行い平面直角座標 2 系における座標値を取得する⁹⁾¹⁰⁾。

付属ツールを使う場合

- 緯度経度（書式 ddd.ddd）から平面直角座標系への座標変換は本資料の projKawase2011.py を使う場合、以下のように行う¹¹⁾。
- 入力座標値が ddd.dd のフォーマットの場合（地理院地図などの利用を想定）

```
python projKawase2011.py ddd 2 131.1252777777772 32.92677500001945
```

```
11716.104554001844 -8113.156710001174
```

```
----summary
```

```
Input
```

```
lon = 1310731.000000; lat = 325536.390000
```

```
Output
```

```
no = 2
```

```
X_survey = -8113.156710; Y_survey = 11716.104554
```

```
Supplementary Info.
```

```
lambdaa = - 0.000000 deg 4.000000 min 5.148886
+0:4:5.148885673032964 azimuth from north clockwise
m = 0.999902
```

- 入力座標値が dddmmss.sss のフォーマットの場合

```
python projKawase2011.py dms 2 1310731 325536.39
```

```
11716.104554058838 -8113.1567121553235
```

```
----summary
```

```
Input
```

```
lon = 1310731.000000; lat = 325536.390000
```

```
Output
```

```
no = 2
```

```
X_survey = -8113.156712; Y_survey = 11716.104554
```

```
Supplementary Info.
```

```
lambdaa = - 0.000000 deg 4.000000 min 5.148886
+0:4:5.148885674043527 azimuth from north clockwise
```

9) 取得すべき座標値は GIS の座標系と一致させる必要がある。つまり、X 軸は東西、Y 軸は南北である。測量における座標系では、X 軸が南北、Y 軸が東西で結果が出力される場合がある。この時は、X と Y を逆に読みかえる必要がある。projKawase2011.py の出力表示では XY は測量座標系である。入力の XY は GIS と同じである。

10) 地理座標変換は、GIS のほか、国土地理院その他の Web サービスを利用できる。

11) python projKawase2011.py [inputform] [no] [x] [y] で使用できる。[inputform] に xy とすれば xy2ddd, ddd とすれば ddd2xy, dms とすれば dms2xy の変換となる。

```
m = 0.999902
```

- 変換の結果、平面直角座標系の座標値は、2系において X 11716.1046, Y -8113.1567 となる。

5.9.10.2 格子番号の取得

- ハイドログラフを設定するスカラー評価点における格子番号を取得する。
- 本資料に付属する `getIndex.py` を `getindex.asc` と同じフォルダにおき、次のように実行する（このスクリプトは `gdal` に依存）。

```
python getIndex.py 11716.1046 -8113.1567
```

- 標準出力の画面から `hydr input grid index` の数字をメモする

```
Input file size is 141, 239
0...10...20...30...40...50...60...70...80...90...100 - done.
Input (x, y) = (11716.1046, -8113.1567)
(xmin, ymax) = (8330.906735453958, -869.8567832317276)
(dx, dy) = (50.0, -50.0)
(ncols, nrows) = (141, 239)
(grid_x, grid_y) = (67, 144)
```

```
hydro input grid index = 14666
```

なお、`getIndex.py` は、`gridindex.asc` における行列数及び南西端の座標（画像座標系のため `xmin`, `ymax`）の座標）を読み、次のアルゴリズムで算出した X と Y 方向の格子番号から、領域全体通しの格子番号を抽出している。

```
grid_x = abs( int( (x - xmin)/dx ) )
grid_y = abs( int( (y - ymax)/dy ) )
"hydro input grid index = {0}".format(idx[grid_y][grid_x])
```

GISを使う場合 `gridindex.asc` は GIS のラスタ形式であるため、GIS で読み込み谷出口（流入境界点）の座標におけるラスタ値を取得すれば、格子番号を取得できる。

5.9.10.3 パラメータ入力ファイルへの書き込み

`param.dat` をテキストエディタで開き `hydr-in` の下の行（18行目）に、

- ハイドログラフ数を Fortran 書式 (5a,5i) (5列文字列, 5列整数) における i に相当する部分に記入（この例では 1）。
- 次の行に、`hydr.dat` の名称を Fortran 書式 (5a) (5列文字列) で記入し、書式 (i10) (15列整数) で格子点番号を記入。
- 続けて、ハイドログラフ設定点の方向を Fortran 書式 `1x,a1,1x,a1` で記入。x 軸正が東、y 軸正が北であるため、例えば、東の場合 `x+y`、北の場合 `x y+`、南西の場合 `x-y-` を記入（この例では北向きに設定）。

```
parameter
diameter=    0.0100
  sigma=     2600.0
  rho=       1000.0
mannings=    0.0400
tan_fric=    0.7000
  Cstar=     0.6000
Cstar_f=     0.2000
  Csc=       0.0200
tau_crit=    0.0500
  delta=     0.0007
```

```
delta_d=    0.0010
epsilon=    0.0010
flg_rebed=    0
iout=      0
hydo-in
no.        1
hydr1     14666 x y+
```

5.9.11 filename.dat の作成

以上から作成した入力ファイル名と出力ファイル名を `filename.dat` に記入する。以下の例では、出力フォルダ `./output/` に出力ファイルを書き出すように指定している。

```
.\dem.dat
.\boundary.dat
.\time.dat
.\hydr.dat
.\param.dat
.\output\map_.csv
.\output\end.csv
.\output\max.csv
.\output\inputdata.dat
.\output\tau_strc.csv
.\output\outflow_volume.dat
```

5.9.12 計算実行

- 出力フォルダの作成

`filename.dat` にパスを記載した場合、フォルダを事前に作成する。上の例では `.\output\` がフォルダの記載である。`debrisflow.exe` 実行時にフォルダがないとエラーとなるため、計算実行前に作成する。

```
mkdir output
```

- 入力ファイルの存在の確認

`filename.dat`, `hydr.dat`, `time.dat`, `dem.dat`, `boundary.dat`, `param.dat` の 6 種類がカレントディレクトリに存在することを確認する。

- 計算を実行する

```
debrisflow.exe
```

5.9.13 結果の可視化

計算結果は、QGIS などで可視化できる。「2章 計算事例」の計算結果は、QGIS (3.10.13) において、「レイヤ」>「レイヤの追加」>「CSV テキストレイヤの追加」によって `./output/tau_strc.csv` を選択し、X 属性を `longitude`、Y 属性を `latitude` として読み込み、レイヤプロパティのシンボロジにおいて「カテゴリ値による定義」の値を「1cm」としてフラグ 2 のみを表示させたものである。なお、サブルーチン `tau` で解説したように、「何 cm」の列はヘッダーの名称だけであり実態を表していない。どの列を選んでも `param.dat` で設定した限界掃流力以上であれば 1 が、流入境界条件から連続していれば 2 が保存されるはずである。

時系列データを出力した場合、計算結果が保存されたフォルダに `mkvtk.py` を保存して、`python mkvtk.py` を実行すれば、`map__00001.csv` などの時系列データから各項目を抽出した `vtk` (テキスト形式の Legacy VTK format) ファイルが作成される。ParaView などの可視化アプリケーションで `vtk` ファイルを開くと、点群 (Point Gaussian) として時系列データを可視化できる。ベクトル量 (総流砂量 `qx_qy`, 流速 `U_V`, 粗粒分砂礫の流砂量 `qblx_qbly`, 細粒分砂礫の流砂量 `qbfx_qbly`) については、Glyph で表示するとベクトル (矢印) の時系列変動を表示できる。

第6章 記号表

本章では、本資料における記号表を示す。同じ記号の重複もあるが文脈により判断できると考える。

基礎方程式・差分方程式に関する記号は、基本的に第3章および第4章の出現順に示す。ソースコードの変数に関する記号は、項目ごとに分類した6つの表に示す。

6.1 基礎方程式・差分方程式に関する記号表

表 6.1 基礎方程式・差分式に関する記号

記号	説明
x, y	空間座標系の X, Y 成分 [m]
t	時間 [s]
q, q_x, q_y	単位幅流量 (ベクトル, X, Y 成分) [m^2/s]
M, N	X, Y 成分の単位幅流量 [m^2/s]
u, u, v	水深平均流速 (ベクトル, X, Y 成分) [m/s]
β	運動量補正係数
g	重力加速度 [m/s^2]
h	流動深 (水と土砂の流れの水深) [m]
z_b	河床位 [m]
$\tau_b, \tau_{bx}, \tau_{by}$	河床せん断力 (ベクトル, X, Y 成分) [N/m^2]
ρ_T	流れ全体の質量密度 [kg/m^3]
ρ_m	間隙流体の質量密度 [kg/m^3]
σ	砂礫の質量密度 [kg/m^3]
ρ	水の質量密度 [kg/m^3]
$s = \sigma / \rho_m$	比重 [-]
$\gamma_s = s - 1$	水中重量 [-]
C_L	粗粒分砂礫の土砂濃度 [-]
C_F	間隙流体中の細粒分砂礫の土砂濃度 [-]
C_{*L}	河床堆積層における粗粒分砂礫の充填土砂濃度 [-]
C_{*F}	河床堆積層の間隙における細粒分砂礫の充填土砂濃度 [-]
C_{*DL}	土石流が新たに堆積する際の粗粒分砂礫の土砂濃度 [-]
C_∞	平衡土砂濃度 [-]
f_r	流動抵抗係数 (または河床の抵抗係数) [-]
i	侵食・堆積速度 [m/s]
δ_e	侵食速度係数 [-]
δ_d	堆積速度係数 [-]
ϕ	砂礫の内部摩擦角 [$rad/^\circ$]
$\bar{\alpha}^2$	停止限界掃流力と (移動) 限界掃流力に対する割合を示す係数 [-]
τ_*	無次元掃流力 $u_*^2 / \{(\sigma / \rho_T - 1)gd_L\}$ [-]
τ_{*c}	無次元限界掃流力 [-]
u_*	摩擦速度 $\sqrt{uh\bar{I}}$ [m/s]
$\tan \theta$	勾配: 河床あるいはエネルギー $\tau / \rho_T gh$ [-]
z_{elev}	初期河床位 [m]
z_v	河床変動量 [m]
z_{vL}	粗粒分砂礫の河床位 [m]

記号	説明
z_{vF}	細粒分砂礫の河床位 [m]
q	単位幅流量 [m^2/s]
Q	全流量 [m^3/s]
B	堰の幅 [m]
C	(堰の公式における) 流量係数 [$m^{1/2}/s$]
L	縦断方向の堰長 [m]
n	マニング粗度係数 [$s/m^{-1/3}$]
I	河床勾配 [-]
R, R'	抵抗係数 [-]
d_L	代表粒径 [m]
	ベクトルの大きさあるいはノルム
XDX, YDX	慣性項 $\partial(uM)/\partial x, \partial(vM)/\partial y$
YDX, YDY	慣性項 $\partial(uN)/\partial x, \partial(vN)/\partial y$
$\delta_x[A_{(P)}]$	A の評価点 P から X 成分に半格子幅離れた 2 点による差分演算子
$\delta_y[A_{(P)}]$	A の評価点 P から Y 成分に半格子幅離れた 2 点による差分演算子
$\delta_X[A_{(P)}]$	A の評価点 P から X 成分に 1 格子幅離れた 2 点による差分演算子
$\delta_Y[A_{(P)}]$	A の評価点 P から Y 成分に 1 格子幅離れた 2 点による差分演算子
$\overline{A_{(P)}}^x$	A の評価点 P から X 成分に半格子幅離れた 2 点による平均演算子
$\overline{A_{(P)}}^y$	A の評価点 P から Y 成分に半格子幅離れた 2 点による平均演算子
$\overline{A_{(P)}}^{xy}$	A の評価点 P から $\sqrt{2}$ 格子幅の距離にある周囲 4 点の平均演算子
$\overline{A_{(P)}}^t$	A の評価点 P と過去方向に 1 時間ステップ離れた同一点との平均演算子
$\overline{A_{(P)}}^{+t}$	A の評価点 P と未来方向に 1 時間ステップ離れた同一点との平均演算子
V_L	単位面積当たりの粗粒分砂礫の体積 (=粗粒分の流動深) [m]
V_F	単位面積当たりの細粒分砂礫の体積 (=細粒分の流動深) [m]
f_{rh}	限界流出土砂量を設定する係数 [-]

6.2 ソースコードの変数に関する記号表

「変数名」列における括弧内は配列の大きさである。「対応記号」列における上添字は時間ステップを表す。「説明」列において「時間ステップ $n+1$ の～」を「 $n+1$ の～」と省略する。また、第4章の差分式では物理量の記号に評価点位置を下添字 (P) で示したが省略する。変数の記号表は次のように分類して示す。

1. 計算領域・データ構造・標高に関する記号表 (表 6.2)
2. 時間入力データに関する記号表 (表 6.3)
3. ハイドログラフ入力データに関する記号表 (表 6.4)
4. 入力パラメータに関する記号表 (表 6.5)
5. 運動方程式・連続式の差分式に関する記号表 (表 6.6)
6. 出力に関する記号表 (表 6.7)

表 6.2 計算領域・データ構造・標高に関する記号表

変数名	対応記号	説明
Ngrid		格子総数
Nxs		X 成分格子辺数
Nys		Y 成分格子辺数
dx		X 成分格子幅
dy		Y 成分格子幅
iEE		東方向インデックス
iNN		北方向インデックス
iWW		西方向インデックス
iSS		南方向インデックス
iNE		北東方向インデックス
iNW		北西方向インデックス
iSW		南西方向インデックス
iSE		南東方向インデックス
iEg		東方向インデックス (格子辺から格子点)
iNg		北方向インデックス (格子辺から格子点)
iWg		西方向インデックス (格子辺から格子点)
iSg		南方向インデックス (格子辺から格子点)
Nwg		水域格子点総数
Nwxs		水域格子辺総数 (X 成分)
Nwys		水域格子辺総数 (Y 成分)
Nwg0		水域格子点総数 (前ステップ)
Nwxs0		水域格子辺総数 (X 成分) (前ステップ)
Nwys0		水域格子辺総数 (Y 成分) (前ステップ)
idxWg(Ngrid)		水域格子点番号の配列
idxWxs(Nxs)		水域格子辺番号 (X 成分) の配列
idxWys(Nys)		水域格子辺番号 (Y 成分) の配列
igc(4, Ngrid)		格子点の隣接格子点番号の接続リスト
iecc(4, Ngrid)		格子点の隣接格子辺番号の接続リスト
ixsxc(4, Nxs)		X 成分格子辺の隣接 X 成分格子辺番号の接続リスト
ixsync(4, Nxs)		X 成分格子辺の隣接 Y 成分格子辺番号の接続リスト
ixsgc(2, Nxs)		X 成分格子辺の隣接格子点番号の接続リスト
iysxc(4, Nys)		Y 成分格子辺の隣接 X 成分格子辺番号の接続リスト
iysyc(4, Nys)		Y 成分格子辺の隣接 Y 成分格子辺番号の接続リスト

変数名	対応記号	説明
iysgc(2, Nys)		Y 成分格子辺の隣接格子点番号の接続リスト
WDg(Ngrid)		格子点の水域/陸域フラグ
WDxs(Nxs)		格子辺の水域/陸域フラグ (X 成分)
WDys(Nys)		格子辺の水域/陸域フラグ (Y 成分)
nxout		流出境界格子辺 (X 成分)
nyout		流出境界格子辺 (Y 成分)
ixout(nxout)		流出境界格子辺番号の配列 (X 成分)
ixgout(nxout)		X 成分流出境界格子点番号と流出方向の配列
iyout(nyout)		流出境界格子辺番号の配列 (Y 成分)
iygout(nyout)		Y 成分流出境界格子点番号と流出方向の配列
nxwall		壁境界格子辺数 (X 成分)
nywall		壁境界格子辺数 (Y 成分)
ixwall(nxwall)		壁境界格子辺番号の配列 (X 成分)
iywall(nywall)		壁境界格子辺番号の配列 (Y 成分)

表 6.3 時間入力データに関する記号表

変数名	対応記号	説明
nt		計算総時間の最大データ分割数 (nt=777)
mlf		計算総時間のデータ分割数
tdlt(mlf)		計算総時間のうち 1 データ分割の計算時間
tdtt(mlf)		1 データ分割の時間刻み
dt	Δt	時間刻み
dt2	$2\Delta t$	時間刻みの 2 倍

表 6.4 ハイドログラフ入力データに関する記号表

変数名	対応記号	説明
mpat		ハイドログラフ波形の数
mlf		1 波形のハイドログラフのデータ刻み数=iout の出力間隔
mhy(mpat)		ハイドログラフ波形の名称
sqin(mpat)		ハイドログラフ波形の流入点河床勾配
qin(mpat, mlf)		ハイドログラフにおける流量
qmin(mpat, mlf)		ハイドログラフの X 成分単位幅流入流量
qnin(mpat, mlf)		ハイドログラフの Y 成分単位幅流入流量
hin(mpat, mlf)		ハイドログラフ流入点の水深 (等流水深)
uin(mpat, mlf)		ハイドログラフ流入点の流速 (X 成分)
vin(mpat, mlf)		ハイドログラフ流入点の流速 (Y 成分)
nbc		ハイドログラフ流入点数
nxbc		X 成分流入点数
nybc		Y 成分流入点数
ihyf(mt)		流入ハイドログラフ番号
ibg(mt)		流入点格子番号
ixbg(mt)		X 成分流入格子点番号
iybg(mt)		Y 成分流入格子点番号
ixhyf(mt)		X 成分ハイドログラフの格子辺番号
iyhyf(mt)		Y 成分ハイドログラフの格子辺番号

変数名	対応記号	説明
ixbd(mt)		ハイドログラフ流入方向 (X 成分)
iybd(mt)		ハイドログラフ流入方向 (Y 成分)
cli(mpat, mlf)		粗粒分砂礫の流入土砂濃度
cfi(mpat, mlf)		細粒分砂礫の流入土砂濃度

表 6.5 入力パラメータに関する記号表

変数名	対応記号	説明
dm	d_L	粗粒分砂礫の代表粒径
tanp	$\tan \phi$	静止内部摩擦角の正接
sig	σ	砂礫の質量密度
rho	ρ	水の質量密度
bit	δ_e	侵食速度係数
bit_d	δ_d	堆積速度係数
th	ε	最小流動深 (水域・陸域判定境界値)
sodo	n	マニング粗度係数
ts	τ_*	無次元掃流力
tauc	τ_{*c}	無次元限界掃流力
wsg	$\gamma_s = \sigma/\rho_m - 1$	砂礫の水中重量
gsr	$s = \sigma/\rho_m$	砂礫の比重
gr	g	重力加速度
bm	β	土石流の運動量補正係数 = 1.25 (固定値)
cst	C_*	河床堆積層の充填土砂濃度
cstf	C_{*F}	河床堆積層の間隙における初期細粒分土砂濃度
c90	$0.9C_*$	土石流における最大土砂濃度
cdl	C_{*DL}	土石流が新たに堆積する際の粗粒分砂礫の土砂濃度
cdc	$C_4 = 0.4C_{*DL}$	土石流の下限土砂濃度 C_4 , C_{dc}
csc	C_5	掃流状集合流動の下限濃度 C_5 , 0.02 等
flg_rebed		粗粒分砂礫による河床調節法のフラグ。1 で使用
iout		時系列の中間出力のフラグ。1 で出力
pi	π	円周率
r13		1/3
r43		3/4

表 6.6 運動方程式・連続式の差分方程式に関する記号表

変数名	対応記号	説明
rof(Ngrid)	ρ_m	間隙流体の質量密度
rot(Ngrid)	ρ_T	土石流の質量密度
qm0(Nxs)		n の X 成分単位幅流量
qm1(Nxs)	M^n	n+1 の X 成分単位幅流量
qm2(Nxs)	M^{n+2}	n+2 の X 成分単位幅流量
qn0(Nxs)		n の Y 成分単位幅流量
qn1(Nxs)	N^n	n+1 の Y 成分単位幅流量
qn2(Nxs)	N^{n+2}	n+2 の Y 成分単位幅流量
sei(Ngrid)	$\tan \theta, S_{ei}$	勾配 (河床, エネルギー, 水面)
us(Ngrid)	u_*	摩擦速度

変数名	対応記号	説明
smn(Ngrid)	n	粗度係数
hz(Ngrid)	$h + z_b$	水位
hn0(Ngrid)	h^{n+0}	n-1 の流動深
hn1(Ngrid)	h^{n+1}	n+1 の流動深
hn3(Ngrid)	h^{n+3}	n+3 の流動深
aia(Ngrid)	i	侵食・堆積速度式
ct1(Ngrid)		n+1 の流れ全体の土砂濃度 (全体)
ct3(Ngrid)		n+3 の流れ全体の土砂濃度 (全体)
cl1(Ngrid)	C_L^{n+1}	n+1 の流れ中の粗粒分砂礫の土砂濃度
cl3(Ngrid)	C_L^{n+3}	n+3 の流れ中の粗粒分砂礫の土砂濃度
cf1(Ngrid)	C_F^{n+1}	n+1 の間隙流体中の細粒分土砂濃度
cf3(Ngrid)	C_F^{n+3}	n+3 の間隙流体中の細粒分土砂濃度
clp(Ngrid)	C_∞	粗粒分砂礫の平衡土砂濃度
czl(Ngrid)	C_{*L}	河床堆積層の粗粒分砂礫土砂濃度
czf(Ngrid)	C_{*F}	河床堆積層の間隙における細粒分砂礫土砂濃度
qblx(Nxs)	$q_{bLx} = C_L M$	X 成分粗粒分流出土砂量
qbfx(Nxs)	$q_{bFx} = (1 - C_L) C_F M$	X 成分細粒分流出土砂量
qbly(Nys)	$q_{bLy} = C_L N$	Y 成分粗粒分流出土砂量
qbfy(Nys)	$q_{bFy} = (1 - C_L) C_F N$	Y 成分細粒分流出土砂量
ezi(Ngrid)	z_{elev}	初期標高 (河床位)
zl1(Ngrid)	z_b^{n+1}	n+1 の河床高
zl3(Ngrid)	z_b^{n+3}	n+3 の河床高
zdz(Ngrid)	z_v	Δt 間の河床変動量 (河床堆積層厚)
vv11(Ngrid)	V_L^{n+1}	n+1 の粗粒分砂礫河床堆積層厚
vv13(Ngrid)	V_L^{n+3}	n+3 の粗粒分砂礫河床堆積層厚
vvf1(Ngrid)	V_F^{n+1}	n+1 の細粒分砂礫河床堆積層厚
vvf3(Ngrid)	V_F^{n+3}	n+3 の細粒分砂礫河床堆積層厚

表 6.7 出力に関する記号表

変数名	対応記号	説明
qmx(Ngrid)		X 成分の単位幅流量
qny(Ngrid)		Y 成分の単位幅流量
qqa(Ngrid)		単位幅流量の大きさ
uux(Ngrid)		X 成分の流速
vy(Ngrid)		Y 成分の流速
uva(Ngrid)		流速の大きさ
ftx(Ngrid)		X 成分の流体力
fty(Ngrid)		Y 成分の流体力
fta(Ngrid)		流体力の大きさ
ipf(Ngrid)		スカラー評価点の計算フラグ
qblxm(Ngrid)		X 成分の粗粒分砂礫の流砂量
qblym(Ngrid)		Y 成分の粗粒分砂礫の流砂量
qbla(Ngrid)		粗粒分流出砂量の大きさ
qbfxm(Ngrid)		X 成分の細粒分砂礫の流砂量
qbfym(Ngrid)		Y 成分の細粒分砂礫の流砂量
qbfa(Ngrid)		細粒分砂礫の流砂量の大きさ

変数名	対応記号	説明
arvt(Ngrid)		土石流の到達時間
qqmax(Ngrid)		最大単位幅流量
qxmax(Ngrid)		最大単位幅流量の X 成分成分
qymax(Ngrid)		最大単位幅流量の Y 成分成分
tqmax(Ngrid)		最大単位幅流量発生時刻
uvmax(Ngrid)		最大流速
uul(Nxs)		X 成分格子辺における X 成分流速
vvl(Nys)		Y 成分格子辺における Y 成分流速
vxm(Ngrid)		最大流速の X 成分成分
vym(Ngrid)		最大流速の Y 成分成分
tvmax(Ngrid)		最大流速発生時刻
hmax(Ngrid)		最大流動深
thmax(Ngrid)		最大流動深発生時刻
hzmax(Ngrid)		最大流動深
thzmax(Ngrid)		最大流動深発生時刻
hahz(Ngrid)		最大流動深発生時の流動深
zahz(Ngrid)		最大流動深発生時の堆積深
zmax(Ngrid)		最大地盤変動高
tzmax(Ngrid)		最大地盤変動高発生時刻
zmin(Ngrid)		最小地盤変動高
tzmin(Ngrid)		最小地盤変動高発生時刻
fmax(Ngrid)		最大流体力
fmx(Ngrid)		最大流体力の X 成分成分
fmy(Ngrid)		最大流体力の Y 成分成分
tfmax(Ngrid)		最大流体力発生時刻
hafmax(Ngrid)		最大流体力発生時の流動深
zafmax(Ngrid)		最大流体力発生時の変動高
taus(Ngrid)	τ_*	限界掃流力
tapc(Ngrid)	$\max(\tau_{*c})$	最大無次元限界掃流力
meshflag(7, Ngrid)		土砂移動フラグ

付録 A 第 3 章の式導出

本章では、第 3 章に示した、各流砂形態の侵食・堆積速度式、平衡土砂濃度式、河床せん断応力をできるだけ体系的に導出する。土石流の項では、平衡土砂濃度などの導出に必要な集合流動の発生条件も述べる。最後の節では質量保存則と運動量保存則の表現として基礎方程式に用いた浅水流方程式を導出する。

A.1 準備

A.1.1 単一粒径と混合粒径

高橋の土石流理論における単一粒径の場合と混合粒径の扱いを簡潔に述べる。

単一粒径の土石流モデルは、平均粒径や 50%粒径など一つの要約統計量で実際の粒度分布を代表させ、土石流内部の土砂や岩などをその代表粒径による球（粒子、粒状体）で構成するモデルである。

混合粒径の土石流モデルは、本資料の範囲では、粗粒分砂礫と細粒分砂礫の二粒径の砂礫で構成されるモデルである。粗粒分砂礫は単一粒径モデルにおける砂礫に相当し、粒子間摩擦や粒子衝突応力を担う。細粒分砂礫は、沈降速度と乱れ速度の比によって決まり、間隙流体中を浮遊してあたかも間隙流体の密度が大きくなったかのような役割を担う¹⁾(高橋ほか, 1990)。考える空間における固体と液体の混合体について、ある瞬間の流れにおいて任意の位置にどの相が存在するかを検査する相関数を定義し、それを一流体モデルの連続式と運動方程式に乗じて体積平均すると、固液混相流におけるそれぞれの相を体積分率で表現することができる(例えば、日本流体力学会, 1991 や芦田ほか, 1994, 日本混相流学会, 2021)。このようにして、土石流中の粗粒分砂礫と細粒分砂礫の量は、単位体積あたりの割合（固相の体積分率）である土砂濃度によって表現できる²⁾。

河床堆積層における単一粒径や混合粒径の扱い方も基本的には同様である。ただし、細粒分砂礫が固相となること、堆積過程において新に土砂濃度が定義されることが異なる。

以上より、混合粒径の場合、流れの中では固相は粗粒分砂礫、液相は間隙に浮遊する細粒分砂礫と水となり(高橋ほか, 1990)、一方、河床堆積層では細粒分砂礫は静止するため、固相は粗粒分砂礫と細粒分砂礫、液相は水となる(図 3.10)。

A.1.2 記号の整理

本章及び次章の付録 B における記号は、高橋土石流理論の教科書(高橋 2004, 高橋 2006, Takahashi, 2014)や原著論文に基本的に従い、本編と概ね同じ記号を用いる(第 6 章記号表を参照)。ただし、式展開を統一的な記号で表現できるよう一部を変更した。

A.1.2.1 流砂形態

流砂形態を区別する場合、土石流、掃流状集合流動、掃流砂流のそれぞれに対し、上添字 (d) , (s) , (b) を用いる³⁾。ただし、流砂形態の記号は混乱が生じない場合は明示しない。

A.1.2.2 土砂濃度

土砂濃度は体積土砂濃度（固相の体積分率）とし、飽和を仮定して空隙だけの空間は考えない。

1) 高橋ほか (1990) は、混合粒径の土石流における間隙流体のせん断応力（全応力から粗粒分砂礫の衝突による分散応力を引いた応力）を摩擦速度の形式で表現し、Ruby 式から求めた細粒分砂礫の沈降速度との比を考慮して、間隙流体に微細粒子が浮遊する条件を実験的・理論的に検討した。高橋ほか (1990) における「微細粒子」を本資料では「細粒分砂礫」と呼称する。流れ場の状態による細粒分砂礫の挙動について、例えば Hotta and Miyamoto (2008) は既往の実験結果を整理した。

2) この定式化において混相流の密度や流速等の物理量を、乱流におけるレイノルズ応力の導出のように、平均部分と変動部分に分ける手続が示されることがあるが、このときは概ね各相が均質に分布している流れを仮定して平均化していると考えられる。掃流状集合流動になると下層に砂礫移動層が形成されるため、それぞれの層を密度流のように独立に扱う二層流モデルも提案されている(例えば、高濱ほか, 2002)。

3) 土石流は debris flow、掃流砂流は bed load である。掃流状集合流動の英語の呼称は様々であり(例えば水山・江頭, 2001)、Takahashi (1991) や Takahashi (2014) では immature debris flow と呼称を用いる。なぜ s の記号を使うのかは調べた範囲の文献では不明だったが、掃流状集合流動をローマ字表記にした souryuujyou... の s ではないかと考えられる。

混合粒径の場合、全砂礫・粗粒分・細粒分を区別するため下添字 T, L, F を付ける⁴⁾。細粒分土砂濃度 C_F は粗粒分砂礫の間隙 $1 - C_L$ における土砂濃度を表す。そのため、全砂礫・粗粒分・細粒分土砂濃度の間には $C_T = C_L + (1 - C_L)C_F$ の関係がある。つまり、 C_T が固相（粗粒・細粒含む）、 $1 - C_T$ が液相のうち水の体積分率である。単一粒径の場合は、細粒分砂礫がなく、粗粒と全砂礫が一致するため、 C_L または C で示す。

アスタリスク * は河床堆積物であることを示す⁵⁾。

下添字の無限大 ∞ は、土石流や掃流状集合流動によって河床との土砂のやり取りをしても河床標高の時間変動がない平衡状態における土砂濃度（平衡土砂濃度）を表す。平衡土砂濃度は単位体積あたりで定義する「体積土砂濃度」と、検査体積の表面を通過する単位面積あたりの全流量に対する粗粒分または細粒分の流砂量の割合で定義する「輸送土砂濃度」があり、区別して認識することは重要である（江頭ほか, 2014）ため、輸送土砂濃度は \hat{C}_∞ のようにハットを付けて表現することにした。堆積過程では土石流から土砂が離脱し河床に新たに土砂を堆積させる。新しい河床堆積層の土砂濃度を C_{*DL} と記す。

いくつかの例を示す。

- C_L, C_F, C_T : それぞれ粗粒分砂礫・細粒分砂礫・全砂礫の土砂濃度。単一粒径の場合 $C = C_L = C_T$
- $C_{L\infty}^{(d)}$: 土石流の流動形態における粗粒分砂礫の平衡土砂濃度（単一粒径の時は C_∞ ）
- $C_{T\infty}^{(d)}$: 土石流の流動形態における全砂礫の平衡土砂濃度

A.1.2.3 質量密度

質量密度は単位体積あたりの質量である。単に密度ともいう。本文と同様であるが下添字 T が土砂濃度と意味が異なるので整理する。砂礫の密度は組成等によらず σ とする。水の密度は ρ とする。本資料の範囲では、付録 B における不飽和堆積層の侵食速度式の議論を除き、土石流や河床堆積物は飽和状態を仮定するため空隙は存在しない。

単一粒径の場合、流れ全体のみかけの密度⁶⁾は、砂礫と水の平均値として $\rho_T = C\sigma + (1 - C)\rho$ とする。

混合粒径の場合、流れ全体のみかけの密度は、砂礫と間隙流体の平均値として $\rho_T = C_L\sigma + (1 - C_L)\rho_m$ とし、その間隙流体（細粒分砂礫が浮遊）のみかけの密度は $\rho_m = C_F\sigma + (1 - C_F)\rho$ とする⁷⁾。

A.1.3 状況設定

次節以降の式の誘導では、いずれも一定勾配 θ の無限長斜面上に、非粘着性材料（砂礫）による一定の厚さ D の河床堆積層が存在する状況を設定し、その上の様々な流れを考えて議論する⁸⁾。

河床堆積層は、（力を無視できる）浸透流が斜面と平行に流れ飽和している状態を仮定する。座標系は斜面下流方向を X 軸、それに直交する上向きを Z 軸とし、堆積層の最下面は、侵食されない基盤岩石（基岩ともいう）のような固定床または問題とする状況で侵食され得ない深さとし、この面の高さを Z 軸の原点とする。 Y 軸方向（奥行方向）は単位幅とする。

土石流と掃流状集合流動の総称を集合流動ということにして、河床堆積物の集合流動の条件を議論するため、堆積層表面から深さ $a (= D - z)$ (a は堆積層表面で深さ 0 で z 方向の負の向きに増加) において検査面を設定する。その検査面に作用する応力で議論するため、検査面上部の集合流動や土砂を検討対象とする。

次節以降、河床堆積物表面上の流れを次のように設定して、様々な関係式を求める。

- 一定水深 h_0 の清水の流れを考え、土石流（集合流動）の発生条件を分析する（図 A.1 (a)）。
- 一定水深 h 、一定土砂濃度 C の土石流を考え、侵食・堆積過程を議論し、平衡土砂濃度を導く（図 A.1 (b)）。
- 同じ状況で、土石流を混合粒径に変更した場合を考え、平衡土砂濃度を導く（図 A.1 (b)）。
- 単一粒径および混合粒径の侵食速度・堆積速度式を導く（図 A.1 (b)）。

4) T は total, L は large, F は fine の略と考えられる。Takahashi (1991) や Takahashi (2014) では、粗粒に対応する英語は本文では coarse を用いるが、数式における記号としては L を用いている。また、後出する C_{*DL} の D は堆積 deposition と考えられる。

5) 下添字の * は本資料の範囲であっても添えられる量によって意味が異なる。 C_* , τ_* , u_* はそれぞれ河床堆積層、無次元掃流力、摩擦速度を表す。2 番目は無次元化を表す記号、その他はある量を表す記号である。

6) 「みかけ」は全体として平均した場合の、という意味であり、「実質」は、その物質だけの、という意味である。後者については、例えば、砂礫の連続式では実質の土砂のやり取りによって方程式を立てる。

7) 細粒分砂礫を含む間隙流体のように平均化された密度の流体として扱うことは、二相流を一相流として扱うことに相当する。このように相を減らした流体を、擬均質流と呼ぶ場合がある（日本混相流学会, 2021, p.77）

8) 非粘着性材料という設定は、クーロン摩擦力を考える際に、粘着力 $c = 0$ とすることを意味する。そこで、本資料では土砂を「砂礫」と呼称する。

- 単一粒径に戻し、土石流の内部だけに注目し、流速分布、平均流速、抵抗則、河床せん断応力を検討する (図 A.1 (b))。
- 下層に土石流、上層に清水が流れる二層流を考え、掃流状集合流動について議論する (図 A.1 (c))。

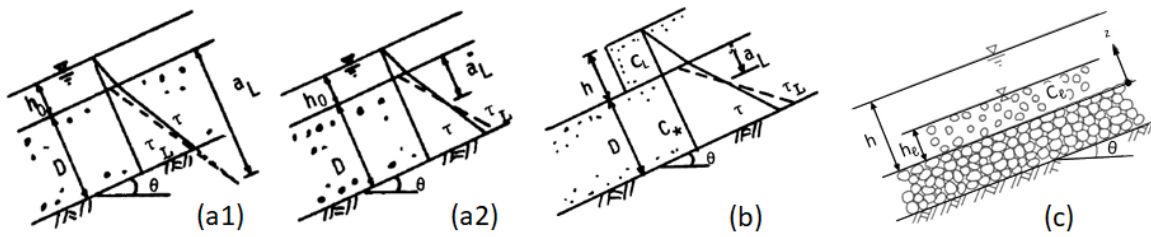


図 A.1 理論解析のための状況設定：a) 発生 (高橋, 1977), b) 土石流 (高橋, 1977), c) 掃流状集合流動 (高橋, 1982)。記号の一部を本文に合わせて編集

A.2 土石流

本節では次の事項を述べる。

- 斜面上に表面流が流れる状況下 (図 A.1 (a)) における高橋 (1977) に基づく集合流動の発生条件を示す。
- 土石流が流下する状況として、水深 h 、土砂濃度 C_L (単一粒径なら C) の土石流が等流で流れている状況を考え、堆積物内での力のつり合う深さ (移動堆積層厚 a_L) を計算すると、平衡土砂濃度、侵食・堆積過程が判定できることを示す (高橋, 1977)。
- 移動堆積層厚がある時間 (遅れ時間 T_e) で土石流に取り込まれると考えると侵食速度式を、また、土石流からある時間 (堆積に係る遅れ時間 T_d) で一定土砂が離脱すると考えて堆積速度式を求める。
- 同じ考えを混合粒径に拡張し、第 3 章で用いた侵食速度式と堆積速度式を導く (高橋・匡, 1986)。
- バグノルドによる粒子流モデル (Bagnold, 1954) を適用し、土石流の河床せん断応力を求める (高橋, 1977)。

A.2.1 発生条件と移動堆積層厚

谷 (1968) は、土石流 (集合流動) の発生形態を、1) 崩壊がそのまま直進し土石流に発展するもの、2) 崩壊土砂が一度溪床に堆積して天然ダムを形成しこれが欠壊して土石流となるもの、3) 溪床の堆積土砂移動し土石流に発展するもの、の 3 つに分類した。

3 番目の河床堆積物侵食型の土石流発生を対象として集合流動が発生する条件を議論する。河床堆積物の土砂濃度は図 3.7 に示す C_3 以上だから砂礫が常に接触しクーロン摩擦⁹⁾が生じる。その抵抗力 τ_L と、重力による河床堆積物が斜面下方に動こうとするせん断応力 τ とのつり合い¹⁰⁾から、河床堆積物が動き得る深さ a_L (移動堆積層厚という。) を計算できる。これを傾斜角について解くことで、3 章の図 3.8 で示した「土石流・崩壊」、「土石流」、「掃流状集合流動」の発生下限勾配が示される (高橋 1977; 芦田ほか 1978)。

高橋 (1977) は、河床堆積物が集合流動する条件として、次の 2 パターンを考えた¹¹⁾ (高橋, 1977)。

9) 土石流発生時の河床堆積層や集合流動内で粒子が常に接触する場合、粒子間にはクーロン摩擦が作用する (土石流流体中の応力は本資料の範囲では砂礫の衝突応力でありクーロン摩擦ではない。土石流中の砂礫が接触しクーロン摩擦を考慮する理論もある)。河床堆積物のある深さの検査面より上に存在する物体の垂直応力に、その検査面に作用する摩擦力は比例する (クーロン摩擦の法則)。傾斜を少しずつ大きくすると、垂直応力は減少しせん断応力は大きくなり、やがて土砂が動き出す。動き出す瞬間に土砂に作用するせん断応力はクーロンの破壊基準を満たす大きさである。クーロンの破壊基準を満たす摩擦力をクーロン摩擦 (力) という。有効応力 (全垂直応力から間隙水圧を引いたもの) を $\sigma' = \bar{\sigma} - \bar{u}$ とすれば、クーロン摩擦は $\tau_c = \sigma' \tan \phi + c$ と書ける。ここに $\bar{\sigma}$ は全応力、 \bar{u} は間隙水圧、 $\tan \phi$ は内部摩擦角の正接、 c は粘着力である。本資料においては、非粘着性材料を対象とするため $c = 0$ とする。なお、土のクーロンの破壊基準は金属の降伏条件 (弾性限界の強度) に相当する条件である。また、破壊時に限らず垂直応力に比例するせん断応力をクーロン摩擦と呼ぶこともある。

10) 力がつり合っているギリギリで保っている状態 (極限状態) で分析する。次の瞬間にバランスが崩れると崩壊が発生する。このような分析方法を極限つり合い法などという。

11) 高橋 (1977) は詳細にせん断力と抵抗を分類して分析しているが、ここでは後の議論に必要な 2 つの場合だけを示す。なお、集合流動の発生限界を概ね同じ設定で研究し、類似した条件式を導いた最初の研究は大岡 (1973) ではないかと考えられる。

- 1.河床堆積層の全層流動 (図 A.1 (a1)) : 基盤岩石の上面 $a = D$ において抵抗力よりせん断応力が大きい, あるいは, 基盤岩石上では力がつり合い, せん断力の深さ方向の変化率より抵抗の変化率が大きい, 河床堆積層内ではせん断力が常に大きい。数式では, $\tau(D) > \tau_L(D)$ あるいは, $\tau(D) = \tau_L(D)$ かつ $\partial\tau(a)/\partial a < \partial\tau_L(a)/\partial a$ と表現できる。
- 2.河床堆積層の一部流動 (図 A.1 (a2)) : 移動堆積層厚 a_L が地表面付近の代表粒径 d_L 程度の深さから堆積層厚 D の間に存在する場合であり $d_L < a_L < D$ である。 $\partial\tau(a)/\partial a < \partial\tau_L(a)/\partial a$ は同様である。

なお, 高橋(1980)は, 図 3.8 の「土石流・崩壊」の右の線(土砂移動が発生しない領域の下限勾配)は, 砂の不安定条件と同様に内部摩擦角 ϕ と同じ勾配とし

$$\tan \theta = \tan \phi \quad (\text{A.1})$$

と表現した。

A.2.1.1 準備

集合流動発生 of 2つのパターンを検討する前に, 準備事項を述べる。

斜面方向の力のつり合いを考える。せん断応力は深さの関数 $\tau = \tau(a)$ であり, 抵抗力 $\tau_L = \tau_L(a)$ は砂礫によるクーロン摩擦力であるとすれば, それぞれ次式のように書ける。

$$\tau(a) = g \sin \theta \{ \sigma a C_* + \rho a (1 - C_*) + \rho h_0 \} = g \sin \theta \{ C_* (\sigma - \rho) a + \rho (a + h_0) \} \quad (\text{A.2})$$

$$\tau_L(a) = g \cos \theta \{ C_* (\sigma - \rho) a \} \tan \phi \quad (\text{A.3})$$

ここに, 単一粒径の場合, 深さ a に設定した検査面上において, 河床堆積物の砂礫質量 $\sigma a C_*$, その内部に含まれる清水質量 $\rho a (1 - C_*)$, さらに表面流による清水質量 ρh_0 が存在することに注意する。 $\tau = \tau_L$ となる堆積層の深さ a を移動堆積層厚 a_L として, これについて解くと,

$$a_L = \frac{\rho \tan \theta}{C_* (\sigma - \rho) (\tan \phi - \tan \theta) - \rho \tan \theta} h_0 \quad (\text{A.4})$$

となり, さらに $\tan \theta$ について解くと

$$\tan \theta = \frac{C_* (\sigma - \rho)}{C_* (\sigma - \rho) + \rho (1 + h_0/a_L)} \tan \phi \quad (\text{A.5})$$

を得る。

次に, 深さ方向のせん断応力や抵抗力の変化率が変わらない条件は, 式 A.2 と 式 A.3 をそれぞれ a で微分した値が等しい, から $\partial\tau(a)/\partial a = g \sin \theta C_* \{ (\sigma - \rho) + \rho \}$ および $\partial\tau_L(a)/\partial a = g \cos \theta C_* (\sigma - \rho) \tan \phi$ より $\partial\tau(a)/\partial a = \partial\tau_L(a)/\partial a$ を求めると,

$$\tan \theta = \frac{C_* (\sigma - \rho)}{C_* (\sigma - \rho) + \rho} \tan \phi \quad (= \tan \theta_2) \quad (\text{A.6})$$

である。右辺は後述する議論から崩壊発生下限勾配であることが分かり, しばしば $\tan \theta_2$ と記される(図 3.8)。

A.2.1.2 堆積層全層流動する場合

基盤岩石面 $a = D$ において抵抗力よりせん断応力が大きい $\tau(D) > \tau_L(D)$ の条件は, 式 A.5 を参考にして,

$$\tan \theta > \frac{C_* (\sigma - \rho)}{C_* (\sigma - \rho) + \rho (1 + h_0/D)} \tan \phi \quad (\text{A.7})$$

である。

次に, 図 A.1 (a1) に示される状況を考える。底面では力がつり合い $\tau(D) = \tau_L(D)$ であるが, せん断力の深さ方向の変化率より抵抗の変化率が大きく $(\partial\tau/\partial a_L)_{a_L=D} < (\partial\tau_L/\partial a_L)_{a_L=D}$ の時, 河床堆積層内ではせん断力が抵抗力より常に大きい。このとき, 式 A.6 より

$$\tan \theta < \frac{C_* (\sigma - \rho)}{C_* (\sigma - \rho) + \rho} \tan \phi \quad (\text{A.8})$$

である。以上から, 全層流動する条件は式 A.7 と式 A.8 から,

$$\frac{C_* (\sigma - \rho)}{C_* (\sigma - \rho) + \rho (1 + h_0/D)} \tan \phi < \tan \theta < \frac{C_* (\sigma - \rho)}{C_* (\sigma - \rho) + \rho} \tan \phi \quad (\text{A.9})$$

と求められる。

A.2.1.3 移動堆積層厚より上部が集合流動する場合

移動堆積層深 a_L は、代表粒径程度より厚く堆積層厚より小さいと考えられ、 $d_L < a_L < D$ の範囲である。よって、式 A.5 より

$$\frac{C_*(\sigma - \rho)}{C_*(\sigma - \rho) + \rho(1 + h_0/d)} \tan \phi < \tan \theta < \frac{C_*(\sigma - \rho)}{C_*(\sigma - \rho) + \rho(1 + h_0/D)} \tan \phi \quad (\text{A.10})$$

である。勾配が左辺より小さいと集合流動とならず掃流砂流となる。そのため、左辺が掃流状集合流動発生下限勾配である¹²⁾。この左辺の式は、図 3.8 の掃流状集合流動と掃流砂流の境界線を表す。

また、 a_L を範囲ではなく $a_L = nd$ (ただし、 $1 \leq n$ の適当な実数で、 $\max(a_L) = D$) とすれば、

$$\tan \theta = \frac{C_*(\sigma - \rho)}{C_*(\sigma - \rho) + \rho(1 + h_0/nd)} \tan \phi \quad (\text{A.11})$$

と書ける。

芦田ほか (1978) や Takahashi (1978) は、物理的な考察によって、表面流が土砂を取り込んで集合流動として流れる条件を、移動堆積層厚 a_L が表面流の水深と同程度、と考えた¹³⁾。1 に近い適当な定数 $\kappa^{-1} = h_0/a_L = h_0/(nd)$ を導入して式 A.11 に代入すると、

$$\tan \theta = \frac{C_*(\sigma - \rho)}{C_*(\sigma - \rho) + \rho(1 + \kappa^{-1})} \tan \phi \quad (= \tan \theta_1) \quad (\text{A.12})$$

となる。 κ を実験で求める適当な定数と考えれば、この勾配は相対水深に依存しない表現となる。現場や実験のデータによる土石流の発生下限勾配が約 15 度であることと調和的である。この勾配を土石流発生下限勾配といい、しばしば $\tan \theta_1$ と記され、図 3.8 の「土石流」の注記の左の線を表す ($\tan \theta_2 > \tan \theta_1$ であることに注意)。

以上から、非粘着性材料で構成される無限長斜面における理想的な集合流動 (土石流と掃流状集合流動) の発生条件は、式 A.9 と式 A.10 の共通部分

$$\frac{C_*(\sigma - \rho)}{C_*(\sigma - \rho) + \rho(1 + h_0/d)} \tan \phi < \tan \theta < \frac{C_*(\sigma - \rho)}{C_*(\sigma - \rho) + \rho} \tan \phi \quad (= \tan \theta_2) \quad (\text{A.13})$$

となる。

A.2.1.4 数値例

κ の値をいくつか示す。高橋 (1977) で示される $h_0/d = 1.4$ に相当する値は $\kappa = 0.714$ である。高橋 (1980) では $\kappa \approx 1$ または実験定数として 0.7、高橋 (1982) では $\kappa = 1 \sim 0.7$ 、高橋・匡 (1986) では 1.0、高橋 (2006, p.171) では 0.5 である。高橋 (2004, p.121) は $\kappa = 1$ とした方が実際の現象と矛盾しないかもしれない、と述べている。土木学会水工学委員会 (2019, p.256) の土石流発生下限勾配は、 $\kappa = 0.7$ とした $\tan \theta = C_*(s - 1)/(C_*(s - 1) + 2.4) \tan \phi$ が記載されている。ここに s は比重で σ/ρ である。

A.2.2 平衡土砂濃度と侵食・堆積過程

前節では表面流 h_0 は清水であったが、今度は無限長斜面の河床堆積層面上を一様な流動深 h の一様な土砂濃度 C_L の土石流が等流で流下する状況を考える (図 A.1 (b))。土石流発生勾配の導出過程と同様の考え方によって、堆積層内に深さ a の検査面を考えて、力が釣り合う移動堆積物の深さ (移動堆積層厚 a_L) を求める。これが河床面と一致する時、侵食も堆積もない平衡状態を表す条件が導かれる。この状態における土石流の土砂濃度を平衡土砂濃度 C_∞ という。平衡土砂濃度を用いると、流れの土砂濃度と平衡土砂濃度の差によって、次の瞬間に侵食過程となるか堆積過程となるかを判断できる。また、この導出によって、河床堆積層の土砂濃度 C_* が理論的な土砂濃度の最大値であることが分かる。この節では、ここに示した事項を単一粒径で求めた後、それを拡張して第 3 章で用いた混合粒径の平衡土砂濃度を導出する。なお、移動堆積層厚の土砂が、その後ある時間を掛けて流れに取り込まれると考えれば (つまり、流れに土砂が連行されると)、次節に示す侵食速度式が導ける。

12) 下限勾配の具体的な数値を示すと、高橋 (1977) では、 $C_* = 0.7$ 、 $\sigma = 2,650[\text{kg}/\text{m}^3]$ 、 $\rho = 1,000[\text{kg}/\text{m}^3]$ 、 $h_0/d = 1.4$ 、 $\tan \phi = 0.8$ とすると、 $\tan \theta < 14.3^\circ$ となることから、現場や実験による土石流発生下限勾配である約 15 度とほぼ一致する、と議論された。高橋 (1977) 当時は、掃流状集合流動という流砂形態が文献に現れ始めた時期であったはずであり、同論文ではこの流砂形態について説明した上で「Type I」という名称で与え、土石流の実験データと区別している。そのため、どの流砂形態の発生下限勾配が明確に区分されておらず、高橋 (1977) は、掃流砂流との境界勾配という意味の「水成勾配」と呼ぶべきものと述べている。その翌年、物理的考察を踏まえ、 $h_0/d = \kappa^{-1}$ とおき、土石流発生下限勾配が新たに示された。

13) つまり、すべての土砂を取り込むと土砂濃度 0.5 程度になるのであれば、土石流のような集合流動が発生する、という物理的解釈に基づく仮定を設けたことになると考えられる。

A.2.2.1 単一粒径

高橋 (1977) に基づいて、単一粒径の平衡土砂濃度、侵食・堆積過程の判定について述べる。

土石流発生下限勾配の導出と同様に、せん断応力と抵抗力（摩擦力）がつり合う移動堆積層厚 $a_L = a|_{\tau=\tau_L}$ を求める。せん断応力は、深さ a の面より上方における河床堆積層と流れ部分（浮力を考慮）の土砂の重量と水の重量の斜面方向成分であり、単一粒径のため、土砂濃度を C で表せば

$$\tau = g \sin \theta \{ (Ch + C_*a)(\sigma - \rho) + (h + a)\rho \} \quad (\text{A.14})$$

である（式 A.2 とは土石流に由来する Ch が加わる点が異なることに注意）。一方、抵抗力は、極限つり合い状態では、間隙水圧を考慮した有効応力によるクーロン摩擦力と等しいと考えられるため、土石流と堆積層内の砂礫を考慮すれば

$$\tau_L = g \cos \theta \{ (Ch + C_*a)(\sigma - \rho) \} \tan \phi \quad (\text{A.15})$$

となる。

従って、 $\tau = \tau_L$ の時、 $a = a_L$ と書けば、

$$a_L = \frac{\rho \tan \theta - C(\sigma - \rho)(\tan \phi - \tan \theta)}{C_*(\sigma - \rho)(\tan \phi - \tan \theta) - \rho \tan \theta} h = \frac{\frac{\rho \tan \theta}{(\sigma - \rho)(\tan \phi - \tan \theta)} - C}{C_* - \frac{\rho \tan \theta}{(\sigma - \rho)(\tan \phi - \tan \theta)}} h \quad (\text{A.16})$$

となる。 $a_L = 0$ とすれば初期河床堆積層の上面と一致することを意味し、侵食も堆積も生じない平衡状態であると解釈できる。上の式を C について解き、この土砂濃度を平衡土砂濃度 $C_\infty = C|_{a_L=0}$ と定義すると、

$$C_\infty = C|_{a_L=0} = \frac{\rho \tan \theta}{(\sigma - \rho)(\tan \phi - \tan \theta)} \quad (\text{A.17})$$

を得る¹⁴⁾¹⁵⁾。

これを式 A.16 に代入すれば、

$$a_L = \frac{C_\infty - C}{C_* - C_\infty} h \quad (\text{A.18})$$

を得る。分母は正であること¹⁶⁾および a の正の向き（Z 軸の負の方向が正）に注意すれば、 $a_L > 0$ であれば河床が低下する侵食過程（流れが河床の土砂を取り込む）であり、 $a_L < 0$ であれば河床が上昇する堆積過程（流れから土砂が離脱する）を表すことが分かる。

従って、流れの土砂濃度と平衡土砂濃度の大小関係によって堆積過程・侵食過程・平衡状態が分類でき、次のようにまとめられる。

- $C = C_\infty (a_L = 0)$ であれば、平衡状態
- $C < C_\infty (a_L > 0)$ であれば、侵食過程
- $C > C_\infty (a_L < 0)$ であれば、堆積過程

なお、高橋 (1977) においては、土石流段波が到達したとき、 $a_L > 0$ であればせん断力が摩擦力を上回り河床堆積層を流動化させて土石流が成長し、 $a_L = 0$ では河床表層の各個運搬のみ発生し、 $a_L < 0$ では段波の流動性がつよくなければ堆積する、と説明される。

14) 土石流の平衡土砂濃度の分母における $\tan \phi$ は、高橋 (1977) が、土石流の流れ内部の一樣濃度が向きによらず一致する条件として平衡土砂濃度を導出する際には砂礫の衝突角 $\tan \alpha$ で表現される（河床せん断応力の項で後述）。Bagnold (1966) の実験やそれを開水路で評価した高橋 (1977) 及び高橋ほか (1978) の実験により、衝突角が内部摩擦角とほぼ同一の値と考えてよいことが明らかになった（例えば、高橋, 2004, p.57-58）。なお、この 2 つの角度の関係について、移動床上の土砂濃度や流速分布を考慮すると、線濃度 $\lambda > 3$ の範囲で、 $\tan \alpha = (1 + \lambda^{-1}) \tan \phi = (C_*/C)^{1/3} \tan \phi$ の関係がある（高橋, 2004, p.61）。この $\tan \alpha$ の第 2 式は Takahashi (1991)、第 3 式は Savage and Sayed (1984) の実験に基づき高橋 (1992) で示された実験式である。この関係式は、Takahashi (1991) で提案される石礫型土石流の第 2 近似理論を導くために用いられる。なお、高橋 (1980) によれば、衝突角は動摩擦角に相当し、内部摩擦角を静摩擦角であり、一般に $\alpha < \phi$ である。つまり $\tan \alpha$ の方が平衡土砂濃度が大きく、流動性が高く表現される原因は粒子間の流れのせん断力の乱れ成分であるとする。しかし、粒子間距離が小さく、実際はほとんど粒子を取り込めないため、より小さな平衡土砂濃度となる $\tan \phi$ で表せる、とされる。高橋 (1977) には乱れ成分には言及していないが同様の議論がみられる。

15) 式 A.17 を $\tan \theta$ について解くと

$$\tan \theta_e = \frac{(\sigma - \rho)C_\infty}{(\sigma - \rho)C_\infty + \rho} \tan \phi = \frac{(s - 1)C_\infty}{(s - 1)C_\infty + 1} \tan \phi$$

となり、この勾配を平衡勾配 θ_e という。土石流に関する文献でしばしば出現する勾配の表現である。例えば、高橋ほか (1991) では、慣性の効果を考慮した飽和堆積層の侵食速度式の定義において、この式の C_∞ を C_L とした式を平衡勾配としている。

16) $C_* - C_\infty$ は、掃流状集合流動発生下限勾配以上の勾配（式 A.10 の範囲）において $\sigma - \rho > 0$ と $\tan \phi - \tan \theta > 0$ を共に満たせば常に正である。

A.2.2.2 混合粒径

高橋・匡 (1986) に従って、流れの土砂濃度が粗粒分砂礫 C_L と細粒分砂礫 C_F に分割される場合を考えて、上述の単一粒径の場合と同様の手続で混合粒径の平衡土砂濃度の導出を行う。ただし、河床堆積物における細粒分砂礫は考えず、粗粒分砂礫の土砂濃度 C_* のみを考える¹⁷⁾。

まず、単一粒径の時と同様に、せん断応力とせん断抵抗力が等しくなる移動堆積層厚 $a_L = a|_{\tau=\tau_L}$ を求める。せん断応力は

$$\tau = g \sin \theta [(C_T h + C_* a)(\sigma - \rho) + (h + a)\rho] \quad (\text{A.19})$$

である¹⁸⁾。 $g \sin \theta C_T h(\sigma - \rho)$ は土石流の重力の斜面流下方向の成分¹⁹⁾、 $g \sin \theta C_* a(\sigma - \rho)$ は河床堆積面下の深さ a までの流動している河床堆積層の重力の斜面方向下流成分、 $g \sin \theta (h + a)\rho$ は深さ a から水面までの水の重力の斜面流下方向成分である。

抵抗力は、クーロン摩擦力であって、土石流中の粗粒分砂礫が抵抗すること、土石流の間隙には細粒分土砂が浮遊していること、河床堆積物の間隙は水だけであることに注意すると、

$$\tau_L = g \cos \theta [C_L h(\sigma - \rho_m) + C_* a(\sigma - \rho)] \tan \phi \quad (\text{A.20})$$

である。

$\tau = \tau_L$ による深さ a である移動堆積層厚 a_L を示した後、平衡土砂濃度を求める。式 A.19 と式 A.20 を等置して、 a_L に関する項を左辺にまとめると、

$$\{\tan \theta [C_* (\sigma - \rho) + \rho] - C_* (\sigma - \rho) \tan \phi\} a_L = C_L (\sigma - \rho_m) h \tan \phi - [C_T h (\sigma - \rho) - h \rho] \tan \theta \quad (\text{A.21})$$

となる。さらに簡潔にするために、全粒分の平衡土砂濃度

$$C_{T\infty} = \frac{\rho \tan \theta}{(\sigma - \rho)(\tan \phi - \tan \theta)} \quad (\text{A.22})$$

を定義して、これを適用することを念頭に式変形すると、

$$- [C_* (\sigma - \rho)(\tan \phi - \tan \theta) - \rho \tan \theta] a_L = -h \left[\rho \tan \theta - (\sigma - \rho_m) C_L \left\{ \tan \phi - \frac{\sigma - \rho}{\sigma - \rho_m} \frac{C_T}{C_L} \tan \theta \right\} \right] \quad (\text{A.23})$$

となり、両辺を $(\sigma - \rho)(\tan \phi - \tan \theta)$ で割ると

$$[C_* - C_{T\infty}] a_L = \left[C_{T\infty} - C_L \frac{(\sigma - \rho_m) \left\{ \tan \phi - \frac{\sigma - \rho}{\sigma - \rho_m} \frac{C_T}{C_L} \tan \theta \right\}}{(\sigma - \rho)(\tan \phi - \tan \theta)} \right] h \quad (\text{A.24})$$

である。これより、混合粒径の移動堆積層 (高橋・匡, 1986, 式 (13))

$$a_L = \frac{C_{T\infty} - C_L \frac{(\sigma - \rho_m) \left\{ \tan \phi - \frac{\sigma - \rho}{\sigma - \rho_m} \frac{C_T}{C_L} \tan \theta \right\}}{(\sigma - \rho)(\tan \phi - \tan \theta)}}{C_* - C_{T\infty}} h \quad (\text{A.25})$$

を得る。

次に、単一粒径の場合と同様に、 $a_L = 0$ における C_L を混合粒径土石流の粗粒分砂礫の平衡土砂濃度 $C_{L\infty}$ と定義し、その導出を行う。まず式 A.25 右辺の分子を変形する。全砂礫の平衡土砂濃度 $C_{T\infty}$ に定義 (式 A.22) を代入すると、

$$\frac{1}{(\sigma - \rho)(\tan \phi - \tan \theta)} \cdot \left[\rho \tan \theta - C_L (\sigma - \rho_m) \tan \phi + C_T (\sigma - \rho) \tan \theta \right] \quad (\text{A.26})$$

となり、これに全土砂濃度の定義²⁰⁾

$$C_T = C_L + (1 - C_L) C_F = C_L (1 - C_F) + C_F \quad (\text{A.27})$$

17) 河床堆積層の充填土砂濃度は全粒分の土砂濃度 C_{*T} と書くべきかもしれないが、ここでは原著論文に記号は従う。

18) 一様粒径と同様に各相の重力成分を全て書き出してみる。土石流における粗粒分砂礫と粗粒分砂礫の固体に作用する重力成分 $g \sin \theta \sigma [C_L + (1 - C_L) C_F] h$ 、間隙流体 $g \sin \theta \rho [(1 - C_L)(1 - C_F)] h$ 、河床堆積層における固相 $g \sin \theta \sigma C_* a$ 、間隙地下水 $g \sin \theta \rho (1 - C_*) a$ となる。これらの項を全て足し合わせ、 $C_T = C_L + (1 - C_L) C_F$ の関係を用いて整理しても同様の式が得られる。河床堆積層では地下水で飽和していても細粒分砂礫が浮遊していないと仮定していることが明確にわかる。つまり、河床堆積層においては細粒分砂礫は固相として扱う。

19) ここでは形式上細粒分砂礫も流れにおける分散相 (固相) として扱われるようである。

20) C_F は粗粒分砂礫に対する間隙における細粒分砂礫の占める体積である。よって、単位体積あたりの細粒分砂礫の体積の割合 (体積分率) は $(1 - C_L) C_F$ であり、水の部分 (飽和を仮定) は、 $(1 - C_L)(1 - C_F)$ であることに注意

を代入して整理すれば,

$$\frac{1}{(\sigma - \rho)(\tan \phi - \tan \theta)} \left[\{C_F \sigma + (1 - C_F) \rho\} \tan \theta - C_L (\sigma - \rho_m) \tan \phi + C_L (1 - C_F) (\sigma - \rho) \tan \theta \right] \quad (\text{A.28})$$

となる。ここで、 $1 = (\sigma - \rho_m) / (\sigma - \rho_m)$ および間隙流体密度の定義 $C_F \sigma + (1 - C_F) \rho = \rho_m$, $(1 - C_F) (\sigma - \rho) = (1 - C_F) \sigma + (1 - C_F) \rho = \sigma - \rho_m$ の関係に注意して変形すると,

$$\frac{(\sigma - \rho_m)}{(\sigma - \rho)(\sigma - \rho_m)(\tan \phi - \tan \theta)} \left[\rho_m \tan \theta - C_L (\sigma - \rho_m) (\tan \phi - \tan \theta) \right] \quad (\text{A.29})$$

となり, 展開すれば

$$\frac{(\sigma - \rho_m)}{(\sigma - \rho)} \left[\frac{\rho_m \tan \theta}{(\sigma - \rho_m)(\tan \phi - \tan \theta)} - C_L \right] \quad (\text{A.30})$$

となる。したがって, 式 A.25 の分子にこれを代入すると

$$a_L = \frac{(\sigma - \rho_m)}{(\sigma - \rho)} \left[\frac{\rho_m \tan \theta}{(\sigma - \rho_m)(\tan \phi - \tan \theta)} - C_L \right] h \quad (\text{A.31})$$

を得る。

$a_L = 0$ として C_L について解いて, 混合粒径の平衡土砂濃度 $C_{L\infty}$ を求めると

$$C_{L\infty} = C_L |_{a_L=0} = \frac{\rho_m \tan \theta}{(\sigma - \rho_m)(\tan \phi - \tan \theta)} \quad (\text{A.32})$$

を得る (高橋・匡, 1986)。この式が第3章に示した式 3.12 である。粗粒分砂礫の平衡土砂濃度は, 単一粒径の式 A.17 において, 間隙流体密度を ρ を ρ_m と置き換えた式と同じ表現となった。

この平衡土砂濃度式を式 A.31 に代入すると,

$$a_L = \frac{(\sigma - \rho_m)}{(\sigma - \rho)} \frac{C_{L\infty} - C_L}{C_* - C_{T\infty}} h \quad (\text{A.33})$$

が得られる。従って, 単一粒径と同様に, 流れの粗粒分砂礫の土砂濃度と粗粒分砂礫の平衡土砂濃度の大小により堆積過程, 侵食過程, 平衡状態が次のように表現できる。

- $C_L = C_{L\infty} (a_L = 0)$ であれば, 平衡状態
- $C_L < C_{L\infty} (a_L > 0)$ であれば, 侵食過程
- $C_L > C_{L\infty} (a_L < 0)$ であれば, 堆積過程

なお, このとき理論的に $\sigma - \rho_m > 0$ (間隙流体密度の最大値を規定) および $C_* - C_{T\infty} > 0$ (流れの固体成分の最大濃度の規定) が設けられることになる。

以上の計算過程で示したように, 混合粒径の式は単一粒径とは異なる様々な条件を考慮して導出されている。平衡土砂濃度では, 混合粒径の式は単一粒径の式において間隙流体密度を ρ を ρ_m と単純に置き換えられたが, 一般に混合粒径の式は単一粒径の式と同じ式にはならない。 a_L の式についてみれば, 単一粒径 (式 A.18) と混合粒径 (式 A.33) で数式の形が異なる。このあたりの事情が, 飽和堆積層における混合粒径の侵食速度式の様々な表現を生み出す一因となる。詳細は侵食速度式の項および付録 B を参照。

A.2.3 侵食速度式

高橋と共同研究者による侵食速度式はいくつかの形式で提案されている。高橋 (1977) では微分方程式による侵食の考え方は提示されているが, その後, 連行や剥離による侵食速度式が提案された。ここでは, 飽和河床堆積層の侵食を対象としてまず単一粒径の式を求め, 次に第3章に示した混合粒径の侵食速度式を求める (その他の式は付録 B 参照)。

飽和河床堆積層に対する侵食速度式の基本的な考え方は, δ_e を侵食速度係数, u を流下方向の流れの流速とすると, 前節までに示した移動堆積層厚 a_L に相当する土砂量が, 次元解析的に (代表長/代表速度) で定義される遅れ時間

$$T_e = \frac{1}{\delta_e} \frac{d_L}{u} \quad (\text{A.34})$$

の時間をかけて侵食されると考える。このとき, 単位時間あたりの侵食量である侵食速度式は

$$i = \frac{a_L}{T_e} = \delta_e \alpha \frac{hu}{d_L} = \delta_e \alpha \frac{q}{d_L} \quad (\text{A.35})$$

と書ける(高橋・匡, 1986)。ここで, 移動堆積層厚 a_L は, 式 A.18 や式 A.33 に示されるように, 水深に比例するため係数を α と表記した。侵食速度係数は実験で決まる定数である。この式は, 流れに直交する河床から物質が取り込まれる速度を表し, 連行 (entrainment) の表現である。

A.2.3.1 単一粒径の侵食速度式

高橋ほか (1999) は, 単一粒径の侵食速度式

$$i = \delta_e \frac{C_\infty - C}{C_* - C_\infty} \frac{q}{d_L} \quad (\text{A.36})$$

を示した。この式は, 高橋 (2004, p.128) に基づくと, 式 A.35 に示したように, 式 A.18 に示される単一粒径の移動堆積層厚 a_L (高橋, 1977) を, 遅れ時間 T_e (式 A.34) で除したものである²¹⁾。

A.2.3.2 混合粒径の侵食速度式

混合粒径の場合, 本来前述した移動堆積層厚 a_L (式 A.33) で見たように, 単一粒径とは式の形が大きく異なる。

しかし, 本資料第3章で用いた式 3.10 は, 高橋 (2004, p.128) が, 上述の単一粒径の侵食速度式において, 議論を簡単にするために河床堆積層における ρ を ρ_m , 流れの土砂濃度 C を C_L と近似的に置き換えて,

$$i = \delta_e \frac{C_\infty - C_L}{C_* - C_\infty} \frac{q}{d_L} \quad (\text{A.37})$$

と表現した式において, 速度に比例した

$$i = \delta_e \frac{C_\infty - C_L}{C_* - C_\infty} u \quad (\text{A.38})$$

と表現しなおしたものである。流速に比例する侵食速度式は, 単一粒径について川池ほか (2000) が示している。侵食速度が流速に比例するには, 次項に示す堆積速度式の評価と同様に, 遅れ時間の代表長さとして代表粒径 d_L ではなく水深 h を用いたことを意味する。

A.2.4 堆積速度式

高橋と共同研究者による堆積速度式はいくつかの形式で提案されている。ここでは, 本資料3章に示した式を導出する (他の式は付録Bに示す)。

堆積速度式の基本的な考え方は, 飽和時における侵食速度式と類似しているが, 流れからの離脱土砂量を移動堆積層厚のように決められない点が異なる。そこで, 流れの土砂濃度が平衡土砂濃度より大きいときにその余剰土砂濃度と体積 (単位面積のため水深と一致) の積に相当する土砂量が堆積すると仮定する。堆積速度式は, その土砂量が瞬時に堆積せずに遅れ時間をかけて堆積すると考えて導かれる (高橋・匡, 1986)。

流れから河床へ粗粒分砂礫の堆積量

$$V_d = h(C_\infty - C_L) \quad (\text{A.39})$$

が, 次元解析的に (代表長さ/代表速度) で定義される遅れ時間

$$T_d = \frac{1}{\delta_d} \frac{d_L}{u} \quad (\text{A.40})$$

で堆積すると考える。ここに, δ_d は堆積速度係数である。このとき, 単位時間あたりに堆積する実質の土砂量 (堆積速度式) は V_d/T_d となる。一方, その堆積土砂量と等しい土砂量が土石流の流れから離脱すると考えて, 土石流から離脱する堆積速度 i を, 侵食速度式と同様に, 間隙水を含む「みかけの単位時間・単位面積あたりの量」と定義する。堆積土砂量と等しい土砂量は, i に土砂濃度をかけた実質土砂量である。この土砂濃度は, 土石流の流体から静止した河床堆積土砂に変化する際の土砂濃度であって, 流れ中の土砂濃度や河床堆積層の土砂濃度とは本来異なるため, 土石流が新たに堆積する際の粗粒分砂礫の土砂濃度 C_{*DL} を新たに定義する。以上より, $iC_{*DL} = V_d/T_d$ が成立, 式 A.39, 式 A.40 および $q = hu$ より, 堆積速度式は,

$$i = \delta_d \frac{C_\infty - C_L}{C_{*DL}} \frac{q}{d_L} \quad (\text{A.41})$$

21) 正確には高橋, 1977 の a_L は式 A.16 であり, 平衡土砂濃度で表現していない。ここに示した侵食速度式の表現が明示されたのは高橋 (2004, p.128) と思われる。

となる (高橋・匡, 1986)。ここで, C_{*DL} は, 不明であるため, 便宜的に, 河床堆積層における全砂礫の充填土砂濃度の定義 $C_* = C_{*L} + (1 - C_{*L})C_{*F}$ における粗粒分砂礫濃度 C_{*L} と等しいとして

$$C_{*DL}(= C_{*L}) = \frac{C_* - C_{*F}}{1 - C_{*F}} \quad (\text{A.42})$$

と定義する²²⁾(中川ほか, 1996)。

中川ほか (2002) は, 堆積速度式の評価の際には, 粒径よりも流れの土砂濃度に関係する量が重要であることから, 高橋ほか (1991) で考え方が示された遅れ時間²³⁾

$$T_d = \frac{1}{\delta'_d} \frac{h}{u} \quad (\text{A.43})$$

を採用し, $q = hu$ の関係式を考慮して

$$i = \delta'_d \frac{C_\infty - C_L}{C_{*DL}} u \quad (\text{A.44})$$

を導いた。ここに, δ_d 代表長さに水深を用いたときの堆積速度係数である。堆積速度式が流速に比例する表現となる。これが第3章における式 3.11 である。

ところで, 侵食速度式の遅れ時間 T_e と堆積速度式の遅れ時間 T_d について, 高橋 (2004, p.129) は, 侵食では粒径が影響すると想定されるが, 堆積では粒径より流動深の影響と考えることが適当, とする (高橋ほか, 1991)。ここで示した式 A.41 と式 A.44 では, 遅れ時間の定義が異なるため, 堆積速度係数 δ_d の値も異なる²⁴⁾。

A.2.5 河床せん断応力

ここまでの土石流の話題は, 単一・混合粒径で土砂濃度 (固相の体積分率) の扱いが異なるものの, 無限長斜面における飽和堆積層上の等流を考えて, 極限つり合い法により移動堆積層厚 a_L を求め, 勾配や連行速度 (侵食・堆積速度) などを議論するという点で論理展開は類似した。本節では異なる考えから出発し, 土石流の力学に関して世界的にも最も初期の研究成果である高橋 (1977) に従って, 土石流内部の応力を示し, 土石流の中に設定した検査面における応力のつり合いを通して流速分布, 断面 (または水深) 平均流速と抵抗則を示した後, 河床せん断応力を求める²⁵⁾。

抵抗則とは, 流れが壁面から受ける抵抗との関係を示す式のことであり, 本資料では流速係数 u/u_* で評価する²⁶⁾。流速係数は, 水深平均流速 u の大きさと摩擦速度 ($u_* = \sqrt{ghI}$) の比 u/u_* である。ここに, g は重力加速度, h は流動深, I は勾配である。河床勾配とすれば $I = \sin \theta$ である。

土石流や掃流状集合流動の集合流動では, 高橋 (1992) や高橋 (2004, p.92) に示される図 A.2 のように, 流速係数と相対水深 (代表粒径と流動深の比 h/d_L) の間に線形関係が成り立つことが実験的にも理論的にも示され,

$$\frac{u}{u_*} = R \frac{h}{d_L} \quad (\text{A.45})$$

と書ける²⁷⁾。ここに, R を抵抗係数という (例えば, 江頭・宮本, 2003)。

河床せん断応力は, 一様勾配 θ の水路の河床面に作用するときには,

$$\tau_b = \rho_T g h \sin \theta = \rho_T u_*^2 \quad (\text{A.46})$$

22) 参考情報として, 中川ほか (1996) における河床堆積層の土砂濃度と C_{*DL} の数値例を示すと, $C_* = 0.65$, $C_{*F} = 0.2$, $C_{*L} = C_{*DL} = 0.56$ である。

23) 高橋ほか (1991) において「堆積に必要な時間 (著者注: 本資料における堆積に関する遅れ時間 T_d のこと) を粒径と流速を用いて表すなど, 少し異なった形で与えてきたが, ここで提案しているような遅れ時間を水深と流速を用いて表すことがよいことに初めて言及した。その後高橋 (2004, p.129) では, 遅れ時間の代表長さに代表粒径より水深を用いることが適当と記述する。

24) 代表長さの違いに応じた遅れ時間はそれぞれ, $T_d = (1/\delta_d)(d_L/U)$, $T'_d = (1/\delta'_d)(h/U)$ である。堆積速度 i が同じであれば, この2式には $\delta'_d = \delta_d(h/d_L)$ の関係がある (高橋, 2006, p.184)。

25) ここに示すのは高橋 (1977) が提案し, 後に石礫型土石流の第一近似理論と呼ばれるものである (高橋, 2004, p.56)。その後, 第二近似理論 (Takahashi, 1991) や慣性域での流動測 (高橋ほか, 1996), 石礫型及び乱流型土石流の一般理論 (高橋・里深, 2002) などが研究された。また, Bagnold (1954) の中立粒子の仮定を克服するために高橋・辻本 (1997) により斜面における粒子流理論が提案された。これらの成果は高橋 (2004) や高橋 (2006), Takahashi (2009) に要点がまとめられる。また, 例えば伊藤 (2001) や高濱ほか (2000b) により構成則が比較・検討される。本資料では割愛する。

26) 流速係数は, Φ, ϕ, ψ の記号で表現する文献も多い。流動抵抗係数 f_r や抵抗係数 R で表現する場合もある。

27) 掃流砂流の場合, 流速係数は対数則となる (例えば, 日野, 1983)。ただし, マニング則を用いることで式 A.45 や式 A.46 の類似表現にまとめることができる。なお, 図 A.2 の「清水乱流」は対数則による流速係数であり, 本資料のマニング型の表現ではないことに注意。

と書ける。ここに、 ρ_T は流れ全体のみかけの密度、 $u_* = \sqrt{gh \sin \theta}$ は摩擦速度（せん断応力を速度の次元で表した指標）である。

河床せん断応力（第3章と表現を合わせる都合上、密度で除して表現する）は、式 A.45 を摩擦速度について解いて、式 A.46 に代入すると

$$\frac{\tau_b}{\rho_T} = u_*^2 = R^{-2} \left(\frac{h}{d_L} \right)^{-2} u^2 = f_r u^2 \quad (\text{A.47})$$

となる。この第4式は、河床せん断応力を、流速の大きさの二乗と比例定数である流動抵抗係数 f_r で表現した式である（例えば、宮本・山下, 2003）。

式 A.47 の第3式と第4式から、流動抵抗係数と抵抗係数には

$$f_r = R^{-2} \left(\frac{h}{d_L} \right)^{-2} \quad (\text{A.48})$$

の関係がある²⁸⁾。

以上から、単位密度あたりの河床せん断応力 τ_b/ρ_T を求める手続きの一つは、抵抗則を流速係数と相対水深で表し、その比例係数である抵抗係数 R を求め、流動抵抗係数 f_r を計算し、 u^2 を乗ずればよいことがわかる。

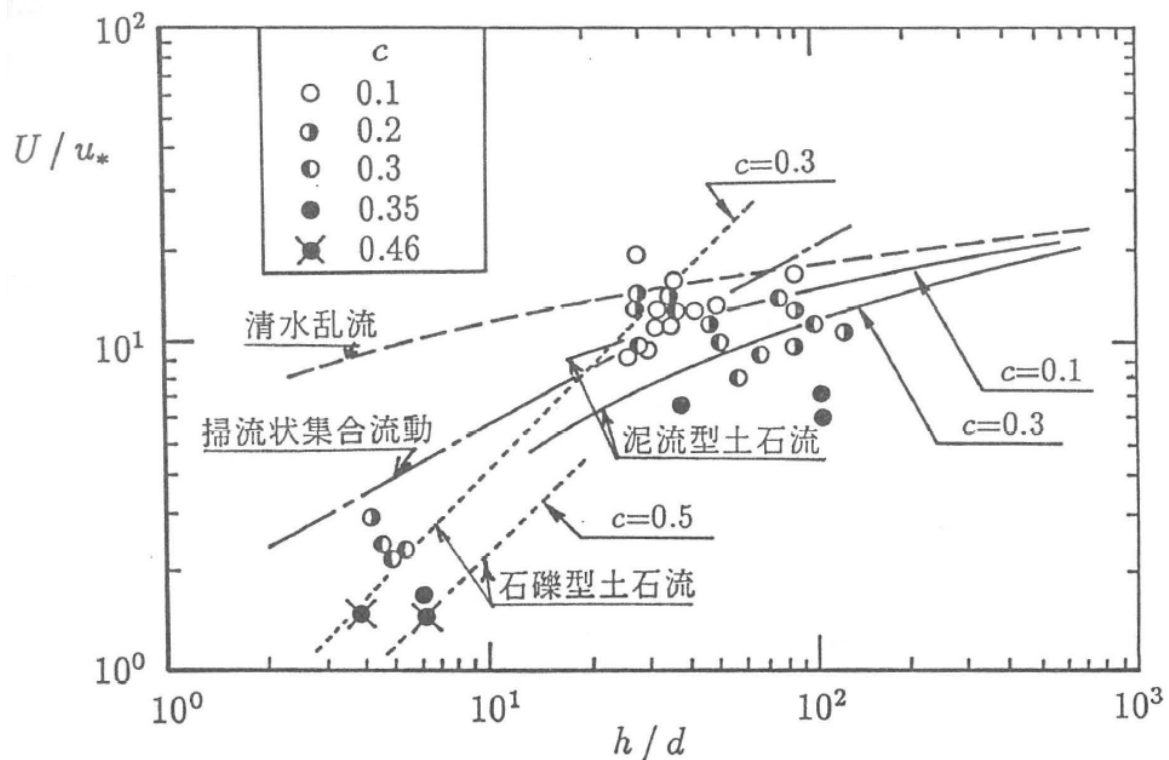


図 A.2 流砂形態の抵抗則（高橋 1992）

A.2.5.1 石礫型土石流の応力モデルと流速分布

高橋 (1977) は、土石流中に固体が分散するのは粒子の衝突と考え、衝突応力にバグノルド応力を適用し石礫型土石流のダイラタント流体モデルを提案した。この研究は、Takahashi (1978) としても発表され、その後の世界の土石流研究の先駆けとなった²⁹⁾(高橋, 2004, p.56)。

Bagnold (1954) は、重力の影響を除くため液中に同密度の粒子を用いた粒子流³⁰⁾の実験によって運動の

28) 同じことであるが $\tau_b = \rho_T f_r u_*^2 = \rho_T u_*^2$ であるため、 $f_r = (u/u_*)^{-2}$ と表現できる。抵抗則を f_r で表現することもある。

29) 同時期に高橋 (1977) と同じように、Bagnold (1954) の粒子流の衝突応力を用いて土石流のダイラタント流体モデルを導いた研究として大同 (1974) がある（この研究に先行して大同 1973 があるが、この論文の中で修正した記載している）。このことについて、一般連続体力学（金谷 (1979) の極性流体、Leigh (1968) に要約される Truesdell ほかによる有理連続体力学や古典場の連続体力学、非線形場の連続体力学など）の理論的枠組みを用いて土石流を研究した宮本 (1985) は、「おそらく、大同および高橋による一連の研究は、土石流の流動機構について初めて力学的なメスを入れたものであり、土石流の流動機構についての力学的な研究方向・指針を指示したものである」と記す。その後、日本の土石流力学の理論研究では、本資料に示す高橋と共同研究者による土石流理論、大同の考え方や表現を発展させた宮本 (1985) や江頭ほか (1997)、伊藤 (2001)、高濱ほか (2000a) 等による土石流理論が大きな流れとして存在する。なお、九州大学の榎ほか (1982) や橋本・平野 (1996) などによる理論研究など、国内だけでも多くの研究者が土石流を研究している。

30) 間隙が空気である粒状体の流れは乾燥粒子流という。間隙流体に限らず重力を考慮するときは斜面流で考える。

激しい慣性領域の流れはダイラタント流体モデル（応力が速度勾配の二乗に比例する非ニュートン流体の構成則）で表現されることを示し、一次元流れの内部垂直応力が $P = \sigma a_i \cos \alpha \lambda^2 d_L^2 (du/dz)^2$ 、せん断応力が $T = P \tan \alpha$ と表現した。ここに、 a_i は係数、 α は衝突角、 λ は線濃度であり、土砂濃度との関係は、 $\lambda^{-1} = (C_*/C)^{1/3} - 1$ である³¹⁾。

高橋 (1977) は、以下に示すように開水路の定常等流にバグノルドのモデルを適用し石礫型土石流の流れを分析した。河床面の上方 z の位置に存在する土石流内部に検査面を設定する。 z の関数である仮想的な検査面を上下に動かして、検査面より上部における応力のつり合いを考える。土石流内の砂礫がそれぞれの衝突によって水面まで分散すると考えると、その応力はバグノルドによる粒子の衝突応力で記述できると考えて、これが検査面上部における重力によるせん断応力とつり合う。このとき、垂直方向 (Z 軸) と流下方向 (X 軸) の応力のつり合いはそれぞれ

$$\sigma a_i \cos \alpha \lambda^2 d_L^2 \left(\frac{du}{dz} \right)^2 = C(\sigma - \rho)g(h - z) \cos \theta \quad (\text{A.49})$$

$$\sigma a_i \sin \alpha \lambda^2 d_L^2 \left(\frac{du}{dz} \right)^2 = \{C(\sigma - \rho) + \rho\}g(h - z) \sin \theta \quad (\text{A.50})$$

と示される。ここで、砂礫が衝突しているため、式 A.50 の右辺におけるせん断応力は砂礫が接触しながら動く時に生じるクーロン摩擦ではなく内部摩擦角がかからないことに注意が必要である。

高橋 (1977) に基づき、 $z = 0$ の河床において流速 $u = 0$ （すべりなし条件³²⁾）の条件で、後者の式 A.50 を変形すると

$$\frac{du(z)}{dz} = \sqrt{K}(h - z)^{1/2}, \quad K = \frac{g \sin \theta}{a_i \sin \alpha} \left[\frac{C\sigma + (1 - C)\rho}{\sigma} \right] \lambda^{-2} d_L^{-2} \quad (\text{A.51})$$

となる。ここに、 K は見やすくするために次に計算する積分変数以外をまとめたものである。角括弧の部分は $(C\sigma + (1 - C)\rho)/\sigma = \rho_T/\sigma$ とすればより簡潔に書ける。 $z = 0 \sim h$ （河床から水面）の範囲で z で積分すると³³⁾

$$u(z) = \frac{2}{3} \sqrt{K} [h^{3/2} - (h - z)^{3/2}] \quad (\text{A.52})$$

であり、3分の2乗則に従う流速分布（ある水深における流速）が得られる。

A.2.5.2 石礫型土石流の断面平均流速と抵抗則

上で求めた流速分布（式 A.52）に、水深平均流速の定義

$$\bar{u} = \frac{1}{h} \int_0^h u(z) dz \quad (\text{A.53})$$

を用いて、積分を計算して平均流速を求めると³⁴⁾

$$\bar{u} = \frac{2}{3} \sqrt{K} \frac{3}{5} h^{3/2} = \frac{2}{5} \left\{ \frac{g \sin \theta}{a_i \sin \alpha} \left[\frac{C\sigma + (1 - C)\rho}{\sigma} \right] \right\}^{1/2} \lambda^{-1} \frac{h^{3/2}}{d_L} \quad (\text{A.54})$$

となる。ここで、右辺において、 $(h^{3/2}/d_L = \sqrt{h}(h/d_L))$ であり、波括弧内の $\sqrt{g \sin \theta}$ と合せて考えれば、摩擦速度 $u_* = \sqrt{gh \sin \theta}$ となるから、両辺を摩擦速度で割ると（平均流速を改めて u と表示し直す）

$$\frac{u}{u_*} = \frac{2}{5} \left\{ \frac{1}{a_i \sin \alpha} \left[\frac{C\sigma + (1 - C)\rho}{\sigma} \right] \right\}^{1/2} \lambda^{-1} \frac{h}{d_L} \quad (\text{A.55})$$

が得られる。左辺は流速係数であるから、この式は石礫型土石流の抵抗則を表し、土石流の流速係数は、式 A.45 に示すように、相対水深（粒径に対する水深の比） h/d_L に比例することが理論的に導けた³⁵⁾。この

31) 線濃度 λ の定義を示す。線濃度は、Bagnold (1954) が提案したダイラタンシーを表現するために導入された指標である。2次元的に等間隔で円形の粒子が配置され、粒径 d_L 、粒子の中心距離を bd_L 、粒子間の間隙長さを s とすると、 $bd_L = s + d_L$ だから $b = (s/d_L) + 1 = \lambda^{-1} + 1$ の関係が得られる。ここで、すき間が大きくなると濃度が下がるように線濃度 $\lambda = s/d_L$ は定義される。粒子の体積濃度（土砂濃度） C 、 $s = 0$ または $\lambda = \infty$ のときの濃度を粒子充填濃度 C_* とすると、 b は粒径に対する一方のすき間の大きさだから、 $C = C_*/b^3 = C_*/(\lambda^{-1} + 1)^3$ の関係から、 $\lambda^{-1} = (C_*/C)^{1/3} - 1$ が得られる（高橋, 2004, p.54）。

32) すべりなし条件とは、河床などで固体と流体の速度差がない移動あるいは静止する状況を表す条件である。

33) $\int_0^z (h - z)^{1/2} dz = -\frac{2}{3} [(h - z)^{3/2}]_0^z = \frac{2}{3} [h^{3/2} - (h - z)^{3/2}]$

34) $\bar{u} = \frac{1}{h} \int_0^h [h^{3/2} - (h - z)^{3/2}] dz = \frac{1}{h} \{ h^{3/2} [z]_0^h + \frac{2}{5} [(h - z)^{5/2}]_0^h \} = \frac{3}{5} h^{3/2}$

35) X 方向の応力のつり合い式から同様の計算を行ない流速分布を求めると、 $u(z) = \frac{2}{3} \sqrt{K} [h^{3/2} - (h - z)^{3/2}]$ 、 $K = \frac{g \cos \theta}{a_i \cos \alpha} [C(1 - \sigma/\rho)] \lambda^{-2} d_L^{-2}$ となり、垂直方向の応力のつり合いから求めた流速分布と異なる。これは不合理であり、土砂濃度が一樣であることに起因する。この不合理を解消する条件 $K = \hat{K}$ を計算すると、土石流の平衡土砂濃度 $C_\infty = \rho \tan \theta / (\sigma - \rho) (\tan \alpha - \tan \theta)$ が得られ、材料物性を一定とすれば土砂濃度が勾配により一意に定まることが示される（高橋, 1977）。なお、江頭・宮本 (2003) は、水深方向に応力や土砂濃度に変化するより一般的な状況において、移動床の河床条件として土砂濃度が C_* 、衝突応力が零だから速度勾配も零などの条件を適用すると、断面平均濃度として平衡土砂濃度と同じ式を導出されることを示した。このことから、平衡土砂濃度式は、移動床における理論河床 $z = 0$ を定める条件であると分かる。

関係は、高橋 (1977) により実験でも確かめられた。

ところで、Bagnold (1966) によれば衝突角 α が内部摩擦角 ϕ とほぼ等しく、Takahashi (1978) は開水路の移動床における土石流の実験により、高橋・吉田 (1979) は固定床の実験によりこれを確認した。

従って、上の式において $\sin \alpha = \sin \phi$ とすれば、

$$\frac{u}{u_*} = \frac{2}{5} \left\{ \frac{1}{a_i \sin \phi} \left[\frac{C\sigma + (1-C)\rho}{\sigma} \right] \right\}^{1/2} \lambda^{-1} \frac{h}{d_L} = R \frac{h}{d_L} \quad (\text{A.56})$$

と書き換えられる。第3式は相対水深の比例係数を抵抗係数 R で表現したもので、

$$R = \frac{2}{5} \left\{ \frac{1}{a_i \sin \phi} \left[\frac{C\sigma + (1-C)\rho}{\sigma} \right] \right\}^{1/2} \lambda^{-1} \quad (\text{A.57})$$

である。

A.2.5.3 河床せん断応力

抵抗係数 R から流動抵抗係数 f_r を求める。式 A.48 に式 A.57 と $\rho_T = C\sigma + (1-C)\rho$ を代入すれば

$$f_r = R^{-2} \left(\frac{h}{d_L} \right)^{-2} = \frac{25}{4} a_i \sin \phi \left[\frac{\rho_T}{\sigma} \right]^{-1} \lambda^2 \left(\frac{h}{d_L} \right)^{-2} \quad (\text{A.58})$$

であるから、土石流の河床せん断応力は、式 A.47 より

$$\frac{\tau_b^{(d)}}{\rho_T} = f_r u^2 = \frac{25}{4} a_i \sin \phi \left[\frac{\rho_T}{\sigma} \right]^{-1} \lambda^2 \left(\frac{h}{d_L} \right)^{-2} u^2 \quad (\text{A.59})$$

と求められる。ここで、線濃度 $\lambda = [(C_*/C)^{1/3} - 1]^{-1}$ を代入すると、

$$\frac{\tau_b^{(d)}}{\rho_T} = \frac{25}{4} a_i \sin \phi \left[\frac{\rho_T}{\sigma} \right]^{-1} \left[\left(\frac{C_*}{C} \right)^{1/3} - 1 \right]^{-2} \left(\frac{h}{d_L} \right)^{-2} u^2 \quad (\text{A.60})$$

を得る。これが第3章で示した土石流の河床せん断応力の式 3.6 である。ただし、第3章では流速 u^2 はベクトル表記 $\|\mathbf{u}\|u$ 、河床堆積物の土砂濃度 C_* は土石流からの離脱後の堆積土砂濃度 C_{*DL} である³⁶⁾(高橋・匡, 1986)。

また、第3章の式における係数 $1/8$ は、 $(25/4)a_i \sin \phi$ の値であり、高橋・中川 (1992) に基づくものである。 $1/8$ になるのは、 $a_i \sin \phi = 0.02$ の時の値である。逆算すると、 $a_i = 0.04$ (高橋, 2004, p.61) であれば、 $\phi = 30^\circ$ 、 $a_i = 0.042$ (Bagnold, 1954) であれば、 $\phi = 28^\circ$ である³⁷⁾。

A.3 掃流状集合流動

掃流状集合流動は、勾配が減少して砂礫が全層に分散できるほど土砂濃度が高くない場合に、上層に水流層、下層に石礫型土石流と同様の集合流動を有する二層流である (図 A.1 (c))。二層流の特徴を考慮した輸送土砂濃度としての平衡土砂濃度を示す (下層の集合流動層だけであれば土石流と同様の土砂濃度)。河床せん断応力は実験に基づく抵抗則を用いて導出する。

A.3.1 平衡土砂濃度

掃流状集合流動の平衡土砂濃度式は、本資料の第3章では、Takahashi (1991, p.47) で提案された

$$\hat{C}_\infty^{(s)} = 6.7(C_\infty^{(d)})^2 \quad (\text{A.61})$$

36) 線濃度における C_* は、線濃度が提案された Bagnold (1954) では、記号 C_0 を使い最大可能濃度 (the maximum possible concentration) と定義され、その後検討を加えた Bagnold (1966) では、記号 C_* を使い最密充填時の最大体積濃度 (the maximum volume concentration at closest possible packing) と定義される。

37) a_i は Bagnold (1954) が反発力の垂直成分に掛かる実験定数として導入し、パラフィン性の中立粒子 (無重力状態で一樣に球が分散させた状態) による慣性領域の実験では 0.042 とした。Bagnold (1966) では、線濃度 λ に応じた領域によって値が代わり、粒子間に明瞭な間隔が見えニュートン流体と振舞う領域 (粒子間に生じる応力より間隙流体間のせん断応力が卓越する領域; 粘性域) で一定値 0.043 で、線濃度の上昇とともに値が大きくなり粒子が常に接触する非ニュートン流体として振舞う領域 (粒子間応力が卓越する領域: 慣性域) では 0.24 まで上昇とした。高橋 (1977) は、この Bagnold の実験結果を参考に、土砂濃度に応じて $a_i = 0.042(C_L < 0.81C_*)$, $a_i = 0.066[1/\lambda - 14] + 0.042(C_L \geq 0.81C_*)$ と整理したが、Bagnold の実験と異なり粒度分布を持つ砂礫の平均粒径で当てはまるかは、この時点では未知とした。一方、粒子間の (動的な) 衝突角 α について、Bagnold (1954) は、パラフィン性の中立粒子 (無重力状態における粒子) の慣性域の実験では平均値として $\tan \alpha = 0.32$ (粘性域では 0.32 から 0.75) とし、Bagnold (1966) は、その後の検討を進め、粒状体の衝突角は土質力学でいうところの安息角で置き換えてよく、また、安息角は、内部摩擦角より少し小さいものの、実験のばらつきを考慮すると内部摩擦角とほぼ等しい、とした。高橋・吉田 (1979) は固定床の実験でこれを確認した。以上から、例えば、 $a_i = 0.042$ 、内部摩擦角 $\phi = 36^\circ$ とすると、 $(25/4)a_i \sin \phi \approx 0.154 \approx 1/6.5$ となる。

を用いた（輸送土砂濃度にはハットを付けた）。高橋ほか（1991）において、 $\hat{C}_\infty^{(s)} \leq C_\infty^{(d)}$ を適用範囲として、「従来の実験結果による」実験式としての平衡土砂濃度として導入される。Takahashi（1991）をみると、いくつか式と実験結果から導かれているが、導出の詳細は記載されていない。Takahashi（1991）に表れる式は元々高橋（1982）に示されたものと考えられる³⁸⁾。

A.3.2 河床せん断応力

実験によれば、掃流状集合流動の抵抗則を流速係数で評価すると、土石流と同様に相対水深に比例すると近似できる（高橋，1982）。この比例係数を抵抗係数 R とすると

$$\frac{u}{u_*} = R \frac{h}{d_L} \quad (\text{A.62})$$

が成り立つ（図 A.2 参照）。従って、みかけの密度あたりの河床せん断応力と抵抗係数 R ，流動抵抗係数 f_r の間には，

$$\frac{\tau_b^{(s)}}{\rho_T} = u_*^2 = R^{-2} \left(\frac{h}{d_L} \right)^{-2} u^2 = f_r u^2 \quad (\text{A.63})$$

が成り立つ（土石流の項と同様）。以下，この関係を踏まえ，第 3 章で示した掃流状集合流動の河床せん断応力を導く。

高橋（1982）は，平衡状態の水流層と粒子流動層（本資料では，砂礫移動層ともいう。）の二層流を考え，下層の粒子流動層については底面摩擦と土砂濃度は土石流と同様（集合流動）とし，上層の水流層については移動床の流れと同様に混合距離理論によるレイノルズ応力を考えて底面（水流層と粒子流動層の界面）にある粒径の限界掃流力が作用するとして，両層における流速分布，抵抗則，流砂量式を導いた。それらの理論式を実験結果で近似して流れ全体の抵抗則として次の式を示した。

$$\frac{u}{u_*} = (0.23 + 0.65 \tan \theta) \frac{h}{d_L} \quad (\text{A.64})$$

ここに，実験によって確認されたこの式の適用範囲は相対水深 h/d_L 5 から 50 程度，勾配 4, 6, 8, 10° である。このとき，抵抗係数は

$$R = 0.23 + 0.65 \tan \theta \quad (\text{A.65})$$

である³⁹⁾。

その後，高橋・藤井（1988）は，同じ実験データの平均的な傾向から， $0.02 \leq C_L < 0.3$ の範囲における実験式

$$\frac{u}{u_*} = 0.8 \frac{\rho_m}{\sigma - \rho_m} \frac{h}{d_L} \quad (\text{A.66})$$

を示した⁴⁰⁾。

高橋（2004, p.88）は，勾配が急になるほど流速係数が小さくなる傾向があるとしつつも，高橋（1982）と伊藤（2001）の実験データと比較すると実験データにはばらつきがあるため，粒子浮遊条件が問題とならない $h/d_L < 20$ の範囲において，勾配に依存しない実験式

$$\frac{u}{u_*} = (0.4 - 0.7) \frac{h}{d_L} \quad (\text{A.67})$$

が成立するものとした。このとき，抵抗係数は

$$R = 0.4 - 0.7 \quad (\text{A.68})$$

である。なお，Takahashi（1991, p.46）では $u/u_* = 1.5(h/d_L)^{0.56}$ を簡易にした式として $u/u_* = 0.4h/d_L$ が示される。

抵抗係数 $R = 0.7$ のとき，河床せん断応力を計算すると，

$$\frac{\tau_b}{\rho_T} = \frac{1}{0.49} \left(\frac{h}{d_L} \right)^{-2} u^2 \quad (\text{A.69})$$

である。これが第 3 章で用いた河床せん断応力（式 3.7）である⁴¹⁾⁴²⁾。

38) 付録 B で式 A.61 の導出を試みた。

39) 勾配 4, 10° の時，抵抗係数 $R = 0.23 + 0.65 \tan \theta$ は，それぞれ約 0.28, 0.34 である。式 A.67 の最小値程度である。

40) 論文中に数値はないため，仮に $\sigma = 2,600 \text{ kg/m}^3$, $\rho_f = 1,200 \text{ kg/m}^3$ としてみると，抵抗係数 $R = 0.8 \frac{\rho_m}{\sigma - \rho_m} = 0.69$ である。

41) 式 A.69 の形式から，流動抵抗係数は $f_r = \frac{1}{0.49} \left(\frac{h}{d_L} \right)^{-2}$ である。

42) この河床せん断応力の式は，高橋ほか（1988a）および同時期の論文において，「従来の実験結果から一つの経験則を誘導」して得

A.4 掃流砂流

山地河川における急勾配に適用可能な掃流砂量式と Manning 則に基づく流量の比を考え、輸送土砂濃度としての平衡土砂濃度を導く。河床せん断応力は Manning 平均流速公式から求める。

A.4.1 平衡土砂濃度

記述が煩雑になるのを防ぐため、結果を示した後、導き方を示す。ただし、掃流砂量式と移動限界掃流力の導き方の詳細は、本文に示す引用文献あるいは高橋 (2006)、中川・辻本 (1986) などを参照されたい。

第 3 章で示した掃流砂流の平衡土砂濃度 (式 3.14)

$$C_{\infty}^{(b)} = \frac{(1 + 5 \tan \theta) \rho_m \tan \theta}{\sigma - \rho_m} \left(1.0 - \bar{\alpha}^2 \frac{\tau_{*c}}{\tau_*}\right) \left(1.0 - \bar{\alpha} \sqrt{\frac{\tau_{*c}}{\tau_*}}\right) \quad (\text{A.70})$$

を中川ほか (1996) に従って導く。中川ほか (1996) には考え方や導き方が示されているが、式 A.70 自体は示されていない。数式として論文に明記したのは高橋ほか (1999) と考えられる。

輸送土砂濃度は、掃流砂量 $q^{(b)}$ 、流量 q とすると、 $\hat{C} = q^{(b)}/q$ と定義される。これを、掃流砂流の平衡土砂濃度 $\hat{C}_{\infty}^{(b)} = \hat{C}$ として同一視できると考える。

急勾配河川に適用可能な高橋の掃流砂量式 (Takahashi 1987) は、

$$q_*^{(b)} = \frac{q^{(b)}}{\sqrt{(\frac{\sigma}{\rho} - 1)gd^3}} = \frac{(1 + 5 \tan \theta)}{\cos \theta} \sqrt{\frac{8}{f}} \tau_*^{3/2} \left(1 - \alpha^2 \frac{\tau_{*c}}{\tau_*}\right) \left(1 - \alpha \sqrt{\frac{\tau_{*c}}{\tau_*}}\right) \quad (\text{A.71})$$

である⁴³⁾⁴⁴⁾。一方、Manning 平均流速は

$$v = \frac{1}{n} h^{2/3} I^{1/2} \quad (\text{A.72})$$

であるから、これに基づく流量は、 $q = hv$ に注意すると

$$q = \frac{1}{n} h^{5/3} \sin^{1/2} \theta \quad (\text{A.73})$$

である⁴⁵⁾。

従って、この 2 式を $\hat{C}_{\infty} = q^{(b)}/q$ に代入すればよい。その際、流砂量と無次元流砂量の関係 $q^{(b)} = \sqrt{(\sigma/\rho - 1)gd^3} q_*^{(b)}$ に注意して、無次元掃流力 $\tau_* = u_*^2/[(\sigma/\rho - 1)gd] = ghI/[(\sigma/\rho - 1)gd]$ および、開水路における Darcy-Weisbach の抵抗則に基づく $\sqrt{8/f} = h^{1/6}/[n\sqrt{g}]$ (導出は後述式 A.77)、河床勾配 $I = \sin \theta$ に注意する。式 A.71 の末尾の () 2 つの部分は計算に影響しないため、計算過程を見やすくするため左辺に移項すると (ただし、 ρ は混合粒径の場合 ρ_m とし、比重を $s = \sigma/\rho_m$ とする)、

$$\begin{aligned} \frac{\hat{C}_{\infty}^{(b)}}{(1 - \alpha^2 \frac{\tau_{*c}}{\tau_*})(1 - \alpha \sqrt{\frac{\tau_{*c}}{\tau_*}})} &= \frac{q^{(s)} q^{-1}}{(1 - \alpha^2 \frac{\tau_{*c}}{\tau_*})(1 - \alpha \sqrt{\frac{\tau_{*c}}{\tau_*}})} \\ &= \{(s - 1)gd^3\}^{1/2} \frac{(1 + 5 \tan \theta)}{\cos \theta} \frac{h^{1/6}}{n\sqrt{g}} \left(\frac{gh \sin \theta}{(s - 1)gd}\right)^{3/2} \cdot \left[\frac{1}{n} h^{5/2} \sin^{1/2} \theta\right]^{-1} \quad (\text{A.74}) \\ &= \frac{(1 + 5 \tan \theta) \tan \theta}{s - 1} \end{aligned}$$

となり、左辺の分母を右辺に移項すれば、上に示した掃流砂流の平衡土砂濃度式である式 A.70 が得られる。

ここから、上の式変形の背景となる開水路の抵抗則と急勾配河川に適用可能な掃流砂量式について簡単に解説する。

$\sqrt{8/f}$ について、等流の開水路流における摩擦損失係数 f' と壁面せん断応力の Darcy-Weisbach の表示 $\tau = \rho f' u^2/2$ (例えば、日野, 1983, p.146-148) に対して、ムーディ (Moody) 図等で値がわかる管路の摩擦損

られる式として示され、その後の論文でも 1/0.49 を係数として用いられる。この数字の根拠は論文中には明示はされていないが、高橋 (1982) や橋本ほか (1986) による理論的研究や実験的検討を参考に高橋・匡 (1986) における簡略化した抵抗係数 (ただし水深の 1.5 乗ずれている) もしくは高橋・藤井 (1988) の式 A.66 を頭頭に $R = 0.7$ を設定したものと推定される。この値は高橋 (2004) の実験式である式 A.67 でも確かめられた。

43) Takahashi (1987) に、式 A.71 の導き方は示されておらず、近似方法やその考え方の詳細は不明であった。

44) 中川ほか (1996, 式 (15)) では、式 A.71 における $\tau_*^{3/2}$ が抜けているため注意。

45) 式 A.73 では、中川ほか (1996) における径深 R を流動深 h とした。この近似は本来広幅水路を仮定した場合の表現である。径深はある断面における流体で濡れた周長 (潤辺) で断面積を割った長さである。壁と流体が接する濡れ長において抵抗力が作用するため、径深の表現が一般的である。

失係数 $f = 4f'$ および $\tau = \rho u_*^2$ の関係式を代入し、流速係数について解くと

$$\frac{u}{u_*} = \sqrt{\frac{8}{f}} \quad (\text{A.75})$$

の関係がある。一方、流速係数の定義式 u/u_* において、分母に摩擦速度の定義 $u_* = \sqrt{ghI}$ を、分子にはマニング平均流速 式 A.72 をそれぞれ代入すると

$$\frac{u}{u_*} = \frac{n^{-1}h^{2/3}I^{1/2}}{(ghI)^{1/2}} = \frac{h^{1/6}}{n\sqrt{g}} \quad (\text{A.76})$$

となるから、式 A.75 と式 A.76 の 2 式を等値すると

$$\sqrt{\frac{8}{f}} = \frac{h^{1/6}}{n\sqrt{g}} \quad (\text{A.77})$$

が得られる。なお、抵抗の指標として $f = 8u_*^2/u^2$ という表示は土砂水理学の文献でよく使われる（例えば、芦田ほか, 1973）。

次に、高橋 (2006, p.217-236) に基づいて、Takahashi (1987) の掃流砂量式の背景を簡潔に述べる。

芦田・道上 (1972) は、平坦勾配における流砂量式

$$q_*^{(b)} = \frac{q^{(b)}}{\sqrt{(s-1)gd^3}} = 17\tau_*^{3/2} \left(1 - \frac{\tau_{*c}}{\tau_*}\right) \left(1 - \frac{u_{*c}}{u_*}\right) \quad (\text{A.78})$$

を提案した（混合粒径を体積分率ではなく粒度分布と考えた混合粒径の式も提案しているが、単一粒径の式を示す；芦田・道上式という）。無次元掃流力 $\tau_* = \rho u_*^2$ （無次元限界掃流力 τ_{*c} も値が異なるだけで式は同一）の関係から $u_{*c}/u_* = \sqrt{\tau_{*c}/\tau_*}$ と表現できる。芦田ほか (1978) は、停止限界掃流力（流れの中の砂粒が停止する掃流力）は、移動限界掃流力（止まっている砂粒を動かす最小掃流力であり、通常これを限界掃流力という。）と異なり、粒径に関わらずほぼ一定となることを見出した。そして、粒径分布が広い山地河川では停止限界掃流力の方が適当であるため、先の式を急勾配河川に適用できるように拡張し

$$q_*^{(b)} = \frac{q^{(b)}}{\sqrt{(s-1)gd^3}} = \frac{12 - 24\sqrt{I}}{\cos\theta} \tau_*^{3/2 - \sqrt{I}} \left(1 - \alpha^2 \frac{\tau_{*c}}{\tau_*}\right) \left(1 - \alpha \frac{u_{*c}}{u_*}\right) \quad (\text{A.79})$$

を提案した（上と同様に単一粒径の場合のみを示す；芦田・高橋・水山式という）。 α^2 は、同じ芦田ほか (1978) による移動限界掃流力の検討で示された。流水中の礫は河床の静止砂礫より移動しやすいと考えられる。そのため、停止限界掃流力は移動限界掃流力より小さいはずである。そこで、停止限界掃流力を掃流力の下限に等しいと仮定する。砂礫が移動するときの流速と停止するときの流速の関係を粒子頂点の流速で評価して、

$$u_{*cs}^2 = \frac{2(\mu_f - \frac{s}{s-1} \tan\theta)}{1 - \frac{s}{s-1} \tan\theta} u_{*c}^2 \quad (\text{A.80})$$

を得た。ここに、 u_{*cs} は停止限界摩擦速度、 u_{*c} は剥離粒子によって定義される移動限界摩擦速度、 μ_f は砂礫の動摩擦係数、 $s = \sigma/\rho$ は砂礫の比重である。両辺を $(s-1)gd$ で割り、 $\tau_* = u_*^2/((s-1)gd)$ の関係を用いて、係数を α^2 で表すと、

$$\tau_{*cs} = \alpha^2 \tau_{*c} \quad (\text{A.81})$$

と変形でき、停止限界掃流力と移動限界掃流力の関係が得られる。ここで、 μ_f は、Bagnold (1956) の実験によると 0.32 から 0.5、大きな粒径では $\mu = 0.5$ であることを参考に、芦田ほか (1978) の実験結果から 0.425 を仮定する。記号を第 3 章の式 3.15 に合せてバーをつければ、

$$\bar{\alpha}^2 = 2 \frac{0.425 - \frac{\sigma}{\rho_m} \frac{\tan\theta}{\sigma/\rho_m - 1}}{1 - \frac{\sigma}{\rho_m} \frac{\tan\theta}{\sigma/\rho_m - 1}} \quad (\text{A.82})$$

を得る。

本項の冒頭に示した Takahashi (1987) の急勾配河川の掃流砂量式（式 A.71）は、高橋 (2006, p.233) によれば、芦田・高橋・水山式（式 A.79）が実験結果より少し過大な流砂量を与える点を修正した経験的な掃流砂量式である。

A.4.2 河床せん断応力

I を河床勾配 $\sin \theta$ とし、マニング平均流速を変形して、 u_*^2 を求められれば、河床せん断応力は $\tau_b/\rho_T = u_*^2$ として得られる。

そこで、 $u_*^2 = ghI$ を作るように式を変形することを考えて、マニング平均流速（式 A.72）の両辺を二乗すると

$$u^2 = n^{-2} ghI \frac{h^{1/3}}{g} \quad (\text{A.83})$$

となり、 ghI について解くと

$$ghI (= u_*^2) = \frac{gn^2}{h^{1/3}} u^2 \quad (\text{A.84})$$

となる。よって、河床せん断応力は

$$\frac{\tau_b}{\rho_T} = u_*^2 = \frac{gn^2}{h^{1/3}} u^2 \quad (\text{A.85})$$

である。流速の自乗の係数が、流動抵抗係数 f_r である。なお、式 A.84 を流速係数が相対水深に比例する形式に変形すると、

$$\frac{u}{u_*} = \frac{d_L h^{-5/6}}{n\sqrt{g}} \frac{h}{d_L} \quad (\text{A.86})$$

となり、相対水深の係数として抵抗係数 R が求まる。

A.5 基礎方程式

A.5.1 3次元の連続式、運動方程式（ナビエ・ストークス方程式）

連続体力学（変形する流体や固体の力学）の問題を解くには、一般に、質量保存則、運動量保存則、角運動量保存則、エネルギー保存則を満たす方程式を立て、材料に応じた構成則などの未知変数間の関係式を複数立て、未知変数の数と同数の方程式を連立して閉じる。本資料の範囲では、応力テンソルが対称テンソルであることで角運動量保存則が満たされ、熱交換などを考えないためエネルギー保存則は用いない。

流体を考えると、流速ベクトル $\mathbf{u} = (u, v, w)^T$ (T は転置) の流速場⁴⁶⁾に、任意の閉空間 V （その表面は S ）を検査体積として設定すれば、保存則は一般に、「検査体積内部に貯留する単位体積あたりの物理量 Y の時間変化率 = 検査体積表面を流れるその量の収支 + 検査体積におけるその量の単位時間、単位体積あたりの湧き出し量」と表現できる⁴⁷⁾。検査体積の表面 S を微小面積要素（面素という） dS に分割し、面素のそれぞれの外向き法線を \mathbf{n} とする。

検査体積内部に貯留する物理量 Y の時間変化率は、 $\int_V (\partial Y / \partial t) dV$ と書ける。

次に検査体積表面を流れる Y の収支を考える。表面を通過する対象とする物理量の収支は、面素 dS を単位時間、単位面積あたりに通過する物理量 $F(Y)$ （これを流束あるいはフラックスという）を足し合わせればよい。 $F(Y)$ の \mathbf{n} 方向へ射影した符号付きの量は内積で求められる。つまり、一つの面素を通過する量は $F(Y) \cdot \mathbf{n} dS$ となり、内積の符号により流出か流入か自動的に表現されるため、検査体積表面のすべての微小な面素を足し合わせる操作（積分）により $\int_S F(Y) \cdot \mathbf{n} dS$ で収支を表現できる⁴⁸⁾。外向き法線が正の向きであるため、この積分値が正のとき検査体積から物理量 Y が流出とする。ガウスの定理⁴⁹⁾を用いれば、面積積分を体積積分に換算できる。

最後に、湧き出し量は、単位時間、単位体積あたりの量を Φ とすれば、 $\int_V \Phi dV$ となる。

46) 位置に物理量が付属する空間を場という。流速場は、任意の位置を指定すると流速の数値が得られる空間のことをいう。

47) 本資料では導出においてもオイラー的表現を採用し固定した閉空間における物理量の収支を考える。流れの中に任意の閉空間を考え、その閉空間の運動をラグランジュ的に表現した後レイノルズ輸送の定理を用いて等価なオイラー表現に書き換える、という手続きもある。

48) フラックスは面素を単位面積・単位時間あたりに通過する物理量のことである。計算の詳細は、ファイマンほか (1986, 2章) や奈佐原 (2020, 15章) などベクトル解析の解説を参照。ここで \cdot は $F(Y)$ がスカラーの場合、ベクトルの場合、テンソルの場合で異なる演算子である。

49) ガウスの定理は、 $\int_S \mathbf{f} \cdot \mathbf{n} dS = \int_V \nabla \cdot \mathbf{f} dV$ である。ここに $\nabla = (\partial/\partial x, \partial/\partial y, \partial/\partial z)^T = \partial/\partial x_j$ ($j = 1, 2, 3$) である。 \cdot は、 \mathbf{f} がスカラーの場合、 $\nabla \cdot \mathbf{f} = \partial f/\partial x_j = \text{grad} f$ 、ベクトルの場合、 $\nabla \cdot \mathbf{f} = \partial f_i/\partial x_i = \text{div} \mathbf{f}$ 、テンソルの場合は演算順序が重要で、 $\nabla \cdot \mathbf{f}$ は正確には $\mathbf{f} \cdot \nabla$ であるから $\mathbf{f} \cdot \nabla = \text{div} \mathbf{f} = \partial f_{ij}/\partial x_k (\mathbf{e}_i \otimes \mathbf{e}_j) \cdot \mathbf{e}_k = (\partial f_{ij}/\partial x_k) \delta_{jk} \mathbf{e}_i = (\partial f_{ij}/\partial x_j) \mathbf{e}_i$ となる。ただし、本文で示す応力テンソル \mathbf{T} は \mathbf{f}^T に相当し、 $\int_S \mathbf{f} \cdot \mathbf{n} dS = \int_V \nabla \cdot \mathbf{f}^T dV$ であり、 $\nabla \cdot \mathbf{f}^T = (\partial f_{ji}/\partial x_k) \mathbf{e}_k \cdot (\mathbf{e}_j \otimes \mathbf{e}_i) = (\partial f_{ji}/\partial x_j) \mathbf{e}_i$ である。ベクトル解析はファイマンほか (1986, 3章) や奈佐原 (2020, 16章) など、テンソル解析は久田 (1992, p.33-35) などを参照されたい。

以上より、上述の言葉を数式で表すと

$$\int_V \frac{\partial Y}{\partial t} dV = - \int_V [\nabla \cdot F(Y) + \Phi] dV \quad (\text{A.87})$$

となり、これが任意の形状・大きさの検査体積で成立するから、恒等的に

$$\frac{\partial Y}{\partial t} = -\nabla \cdot F(Y) + \Phi \quad (\text{A.88})$$

が成り立つ。式 A.87 より、式 A.88 の各項は体積積分するとその物理量になり、その次元は単位時間、単位体積あたりのその物理量の次元となることがわかる。従って、 Y は単位体積あたりの量で「XX 密度」、 ∇ は 1/長さの次元のため $F(Y)$ は単位時間、単位面積あたりの量で「XX 流束（あるいは XX フラックス）」、 Φ は単位時間、単位体積あたりの量で「XX 速度」などの名称で呼ばれることがある。

A.5.1.1 質量保存則

質量保存則を表す式を導く。 Y は単位体積あたりの質量（質量密度、あるいは単に密度） ρ とし、質量流束 $F(Y) = \rho \mathbf{u}$ 、質量の湧き出し量 ϕ とすれば、表面から流入する時は面素の法線ベクトルの外向き（正は外向きだから流出方向）と逆向きであることに注意して、

$$\int_V \frac{\partial \rho}{\partial t} dV = - \int_S \rho \mathbf{u} \cdot \mathbf{n} dS + \int_V \phi dV \quad (\text{A.89})$$

である。ガウスの定理を用いて、面積積分を体積積分に変換すると、

$$\int_V \frac{\partial \rho}{\partial t} dV = - \int_V \nabla \cdot (\rho \mathbf{u}) dV + \int_V \phi dV \quad (\text{A.90})$$

となり、これが任意の形状の検査体積で成立するから、

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = \phi \quad (\text{A.91})$$

となる。質量保存則を表すこの偏微分方程式を連続式という。

湧き出し量を零として、非圧縮性流体の定義式 $D\rho/Dt = 0$ を、式 A.91 に適用すると

$$\nabla \cdot \mathbf{u} = 0 \quad (\text{A.92})$$

を得る⁵⁰⁾。これが非圧縮性流体の連続式である。

A.5.1.2 運動量保存則

運動量保存則を表す運動方程式を導く。 Y は単位体積あたりの運動量（運動量密度） $\rho \mathbf{u}$ とする（運動量は ρ と \mathbf{u} の積でひとつの物理量と捉える）。ところで、流れに外部から力が作用することは流れが外部から運動量を吸収することと等価である（今井, 2003）。連続体では有限な検査体積の変形を考えるため、流れに作用する力は表面力と体積力に区分される。体積力は、質量が存在するだけで供給される重力等がこれにあたり、運動量源（湧き出し）とみなすことができ、 $\rho \mathbf{K}$ で記す。一方、表面力は、流体内部に任意に設けた検査体積の表面に作用する力のことで、単位面積・単位時間あたりに流出する運動量を表す（通常、正の向きは面素の法線ベクトルと一致させ、引張りを正とする）。曲面上の位置で定義するため、その位置における面の向き \mathbf{n} に対して定まる応力テンソル $\mathbf{T} = \tau_{ji}$ (j は面の法線向き、 i はその面における方向を表す添字) によって、表面力 \mathbf{t} は、 $\mathbf{t} = \mathbf{T} \mathbf{n}$ (テンソル表記⁵¹⁾) では $t_i = \tau_{ji} n_j$) と書ける（コーシー関係）。また、流体が流れると質量も流れる（運動量が移流する）ことから、検査体積の表面の面素を通過する運動量として、 $(\rho \mathbf{u}) \mathbf{u}$ ⁵²⁾ も存在する⁵³⁾。

50) D/Dt は物質微分（あるいはラグランジュ微分）であり、 $D/Dt = \partial/\partial t + u\partial/\partial x + v\partial/\partial y + w\partial/\partial z = \partial/\partial t + \mathbf{u} \cdot \nabla$ である。

式 A.91 は $\phi = 0$ の時、 $\frac{\partial \rho}{\partial t} + \mathbf{u} \cdot \nabla \rho + \rho \nabla \cdot \mathbf{u} = 0$ であるから、非圧縮性の条件 $D\rho/Dt = 0$ を適用すると式 A.92 が得られる。

51) ここでは、直交行列を添字で簡潔に表現したもの、という程度で了承されたい。 i, j はそれぞれ 1~3 の範囲であり、1 は X 方向、2 は Y 方向、3 は Z 方向を表す。添字が一つの記号は、ベクトルの成分を表す。 τ_{ij} のように添字が 2 つの記号は、法線ベクトル i 方向の面に作用する j 方向の軸の応力のように、3 行 3 列の行列を表す。ひとつの項に同じ添字があれば、総和規約を適用する。

52) テンソルであるため $\rho \mathbf{u} \otimes \mathbf{u}$ と書かれる量であるが、簡略化した表記を用いた。

53) 林 (1996) は、移流によって運搬される運動量 $(\rho \mathbf{u}) \mathbf{u}$ を「流体に携行されて 1 つの断面をとって単位時間あたりに運ばれる運動量」であるから「携行運動量流束」と呼び (p.76)、一方、応力テンソル（直交成分が圧力）による運動量は実際の物質の移動を伴わないため、「流体が静止していても同じ断面を通じて伝えられる単位時間あたりの運動量の値を示しており、いわば”伝導的な”運動量流束」であるとして「伝導運動量流束」と呼んだ (p.81)。日野 (1983) は、それぞれの項を「運動量束」、本文における F に相当する量を運動量の流れあるいはフローフォースと呼称している。

以上より、運動量保存則は

$$\int_V \frac{\partial \rho \mathbf{u}}{\partial t} dV = - \int_S (\rho \mathbf{u}) \mathbf{u} \cdot \mathbf{n} dS + \int_S \mathbf{T} \cdot \mathbf{n} dS + \int_V \rho \mathbf{K} dV \quad (\text{A.93})$$

と書けるから、ガウスの定理により体積積分に変換すると

$$\int_V \left[\frac{\partial(\rho \mathbf{u})}{\partial t} + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) \right] dV = \int_V \left[\nabla \cdot \mathbf{T} + \rho \mathbf{K} \right] dV \quad (\text{A.94})$$

となり、先と同様に任意の検査体積で成立するとき

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) = \nabla \cdot \mathbf{T} + \rho \mathbf{K} \quad (\text{A.95})$$

となる。この運動量保存則を表現する偏微分方程式を運動方程式という。

非圧縮性流体におけるニュートン流体の構成則は、行列の成分表記（テンソル表記）で書くと、

$$\mathbf{T} = \tau_{ji} = -p \delta_{ij} + \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \quad (\text{A.96})$$

となる。ここに、 p は圧力、 μ は粘性係数、 δ_{ij} はクロネッカーデルタで、 $i = j$ なら 1、 $i \neq j$ なら 0 である⁵⁴⁾。X 方向 ($i = 1$) の場合、ニュートン流体の構成則を運動方程式に代入すると左辺第 3 項は、

$$(\nabla \cdot \mathbf{T})_{i=1} = \left(\frac{\tau_{ji}}{\partial x_j} \right)_{i=1} = \frac{\partial}{\partial x_j} \left[-p \delta_{1j} + \mu \left(\frac{\partial u_1}{\partial x_j} + \frac{\partial u_j}{\partial x_1} \right) \right] = -\frac{\partial p}{\partial x_1} + \mu \frac{\partial^2 u_1}{\partial x_1^2} + \mu \frac{\partial}{\partial x_1} \left(\frac{\partial u_j}{\partial x_j} \right) \quad (\text{A.97})$$

となり、右辺第 3 項は連続式 $\nabla \cdot \mathbf{u} = \partial u_j / \partial x_j = 0$ より零となる。他の方向も ($i = 2, 3$) 同様に計算できるから、運動方程式

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) = -\nabla p + \mu \nabla^2 \mathbf{u} + \rho \mathbf{K} \quad (\text{A.98})$$

を得る。非圧縮により $\rho = \text{const.}$ 、動粘性係数 $\nu = \mu / \rho = \text{const.}$ とすれば、単位質量あたり（比体積）での表現

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u} \mathbf{u}) = -\frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u} + \mathbf{K} \quad (\text{A.99})$$

が得られる。このニュートン流体の運動方程式をナビエ・ストークス方程式という。

A.5.2 浅水方程式の導出

本資料の基礎方程式は、流れの水深方向の変化に対して平面方向の変化が大きいため平面 2 次元の基礎方程式を用いる。これを導くためには、上述の 3 次元の運動方程式と連続式を水深積分して平均化操作を行う。これを浅水流近似という。

ここでは、砂防・地すべり技術センター砂防技術研究所 (2000) にならって、本資料で用いた基礎方程式を導く。運動方程式は一流体モデルとして扱うため、その一流体モデルを浅水流化する。一方、連続式においては、河床を不連続面と捉え、原点から水面まで（流れから河床堆積層までを含む）の検査体積を考える。河床面のような不連続面を含んでも成立する保存則は式 A.90 に示したような積分型の表現である。それを水深積分して連続式を浅水流化する。この際、不連続面とそれをはさむ連続体をそれぞれ別に考える。流体が不連続面を通過する条件は、不連続面に直交する前後の流速を u_1, u_2 としたときに、質量流束が変わらない条件 $\rho(u_1 - u_2) = 0$ であることが示される⁵⁵⁾。この u_1, u_2 は侵食・堆積速度に相当することになる。

以下本項では、数学の公式を示した後、連続式と運動量保存則の浅水流化を行う。浅水流化の詳細は、高橋ほか (1986) や細田 (2004)、池田 (1999)、関根 (2005)、後藤 (2022, 9 章) などを、不連続面を含む連続体の保存則については例えば Leigh (1968) やベッカー・ビュルガー (1983)、芦田ほか (1994)、Hutter *et al.* (1994) 等を確認されたい。

A.5.2.1 数学の公式

微分積分学の基本定理や積の微分公式から導かれる次のライプニッツの積分公式

$$\begin{aligned} \frac{\partial}{\partial x} \int_{b(z)}^{t(z)} f(x, y, z) dz &= \int_{b(z)}^{t(z)} \frac{\partial f(x, y, z)}{\partial x} dz \\ &+ f(x, y, t) \frac{\partial t(z)}{\partial x} - f(x, y, b) \frac{\partial b(z)}{\partial x} \end{aligned} \quad (\text{A.100})$$

54) δ_{ij} は行列表記なら単位行列である。

55) 跳び条件または跳躍条件 (jump condition) という。圧縮性流体における、衝撃波前後のランキン・ユゴニオの関係式がこれに当たる。

を用いる（例えば、関根 2005；後藤 2022, 9 章参照）。また、次の平均操作

$$\bar{f}(z) = \frac{1}{b-a} \int_a^b f(z) dz \quad (\text{A.101})$$

を用いる。

A.5.2.2 連続式の浅水流方程式

流れの浅水流化 まず、数学の公式の使い方を示す意味で、高橋ほか（1986）を参考に流れの部分の浅水流化を行う。式 A.92 を河床底面 $z_b = z_b(t, x, y)$ から水面 $z_s = z_s(t, x, y)$ まで積分する。

第 1 項にライプニッツの公式を適用すると、

$$\int_{z_b}^{z_s} \frac{\partial u}{\partial x} dz = \frac{\partial}{\partial x} \int_{z_b}^{z_s} u(z) dz - u(z_s) \frac{\partial z_s}{\partial x} + u(z_b) \frac{\partial z_b}{\partial x} \quad (\text{A.102})$$

ここで、右辺第 1 項に平均操作を行う。水深 h と河床、水面との間に $h = z_s - z_b$ の関係があることに注意し、断面平均流速を U とすれば、式 A.101 より

$$U = \frac{1}{z_s - z_b} \int_{z_b}^{z_s} u(z) dz \quad (\text{A.103})$$

であるから、

$$\int_{z_b}^{z_s} u(z) dz = hU \quad (\text{A.104})$$

である。従って、式 A.102 は

$$\int_{z_b}^{z_s} \frac{\partial u}{\partial x} dz = \frac{\partial(hU)}{\partial x} - u(z_s) \frac{\partial z_s}{\partial x} + u(z_b) \frac{\partial z_b}{\partial x} \quad (\text{A.105})$$

である。第 2 項は同様である。第 3 項は、

$$\int_{z_b}^{z_s} \frac{\partial w}{\partial z} dz = w(z_s) - w(z_b) \quad (\text{A.106})$$

である。

以上より、式 A.92 を積分すると、式 A.105 や式 A.106 を代入して各項の和をとると

$$\frac{\partial(hU)}{\partial x} + \frac{\partial(hV)}{\partial y} - \left[u(z_s) \frac{\partial z_s}{\partial x} + v(z_s) \frac{\partial z_s}{\partial y} - w(z_s) \right] + \left[u(z_b) \frac{\partial z_b}{\partial x} + v(z_b) \frac{\partial z_b}{\partial y} - w(z_b) \right] = 0 \quad (\text{A.107})$$

となる。ここで、左辺の角括弧を変形するために運動学的境界条件を適用する。つまり、河床面における流体の Z 方向の流速と河床面形状 $z_b(t, x, y)$ の Z 方向の時間変化（速度）が等しいという境界条件が成立するものとする。このとき、河床面形状の時間微分と連鎖則を用いると、

$$\begin{aligned} w(z_b) &= \left(\frac{dz}{dt} \right)_{z_b} = \left(\frac{\partial z}{\partial t} + \frac{\partial z}{\partial x} \frac{\partial x}{\partial t} + \frac{\partial z}{\partial y} \frac{\partial y}{\partial t} \right)_{z_b} \\ &= \frac{\partial z_b}{\partial t} + u(z_b) \frac{\partial z_b}{\partial x} + v(z_b) \frac{\partial z_b}{\partial y} \end{aligned} \quad (\text{A.108})$$

である。水面の境界条件も同様だから、水面形状 $z_s(t, x, y)$ を時間微分して

$$w(z_s) = \left(\frac{dz}{dt} \right)_{z_s} = \frac{\partial z_s}{\partial t} + u(z_s) \frac{\partial z_s}{\partial x} + v(z_s) \frac{\partial z_s}{\partial y} \quad (\text{A.109})$$

である。以上より、式 A.107 に式 A.108 と式 A.109 を代入し、 $\partial(z_s - z_b)/\partial t = \partial h/\partial t$ の関係より

$$\frac{\partial h}{\partial t} + \frac{\partial hU}{\partial x} + \frac{\partial hV}{\partial y} = 0 \quad (\text{A.110})$$

を得る。これが浅水流化した連続式であり、第 3 章で用いた流れ全体の連続式である。

河床を含む連続式の浅水流化 先は簡単のため河床より上方にのみ検査体積を設定したが、河床変動を考慮した連続式を導くため、今度は河床より下方の原点（基盤岩石など適当な原点を $z = 0$ とする）から水面までの範囲に検査体積を設定する。この時、 $z = z_b$ に河床（不連続面）が存在することになる。式 A.92 の積分範囲を河床 $z = z_b$ で 2 つに分割し、

$$\int_{z_b}^{z_s} \nabla \cdot \mathbf{u}_1 dz + \int_0^{z_b} \nabla \cdot \mathbf{u}_2 dz = 0 \quad (\text{A.111})$$

とする。ここに、左辺第1項が流体（流速の添字1）、第2項が河床堆積物（流速の添字2）である。

まず、式 A.111 の左辺第1項、つまり流体部分の連続式の計算結果は前項で導いた式 A.110 と同様である⁵⁶⁾。

次に、河床面より下、式 A.111 の左辺第2項について考える。原点に基盤岩石など固定床を想定すると原点における境界条件として流速を $u(0) = v(0) = w(0) = 0$ と考えることができ、式 A.108 を参考にすれば運動学的境界条件は $z = 0$ では $w(0) = 0$ となる。また、河床においては、流体と河床の接線成分の相対速度が零、つまり流体が固体とともに動く滑りなし条件で近似できると考えるとすれば、境界条件は $u(z_b) = 0, v(z_b) = 0$ となる。したがって、運動学的境界条件は、式 A.109 を参考に

$$\begin{aligned} w(z_b) &= \left(\frac{dz}{dt} \right)_{z_b} = \frac{\partial z_b}{\partial t} + 0 \frac{\partial z_b}{\partial x} + 0 \frac{\partial z_b}{\partial y} \\ &= \frac{\partial z_b}{\partial t} \end{aligned} \quad (\text{A.112})$$

となる。さて、この境界条件と河床堆積層内の断面平均流速 $U = V = 0$ という静止の条件を考えて、式 A.111 左辺第2項にライプニッツの公式を適用すると、式 A.107 を参考にすれば、

$$\int_0^{z_b} \nabla \cdot \mathbf{u}_2 dz = \frac{\partial z_b}{\partial t} \quad (\text{A.113})$$

となる。

以上より、式 A.111 は、式 A.110 と式 A.113 より

$$\frac{\partial h}{\partial t} + \frac{\partial hU}{\partial x} + \frac{\partial hV}{\partial y} + \frac{\partial z_b}{\partial t} = 0 \quad (\text{A.114})$$

を得る。左辺の最後の項は、河床標高の時間変動である。この変動量は侵食・堆積速度 i と等しいはずだから、上式は次のように2つの式に分解できる。

$$\begin{aligned} \frac{\partial h}{\partial t} + \frac{\partial hU}{\partial x} + \frac{\partial hV}{\partial y} &= i \\ \frac{\partial z_b}{\partial t} &= -i \end{aligned} \quad (\text{A.115})$$

運動量を $M = q_x = hU, N = q_y = hV$ のようにひとつの物理量として捉えれば、第3章における流れ全体の連続式（式 3.3）と河床位方程式（式 3.20）となる。この式は、形式的には、河床面を界して検査体積を流れと河床堆積層の2つに分割し、それぞれの検査体積において連続式が成立し、界面での相互作用として侵食・堆積速度式を導入した形と解釈できる。

次に、砂礫の連続式を立てる。式 A.111 において、検査体積の空間を分割して、固相の部分だけに着目して連続式を考えればよい。左辺第1項は流れ、左辺第2項は河床堆積層だから、それぞれ C_L, C_{*L} を乗じれば

$$\int_{z_b}^{z_s} C_L \nabla \cdot \mathbf{u}_1 dz + \int_0^{z_b} C_{*L} \nabla \cdot \mathbf{u}_2 dz = 0 \quad (\text{A.116})$$

となる。同じように計算すると

$$\frac{\partial C_L h}{\partial t} + \frac{\partial C_L hU}{\partial x} + \frac{\partial C_L hV}{\partial y} + C_{*L} \frac{\partial z_b}{\partial t} = 0 \quad (\text{A.117})$$

となる。ただし、 C_{*L} は定数であるために微分演算子の外に配置した。先と同様に、侵食・堆積速度式を用いると、

$$\begin{aligned} \frac{\partial C_L h}{\partial t} + \frac{\partial C_L hU}{\partial x} + \frac{\partial C_L hV}{\partial y} &= C_{*L} i \\ \frac{\partial C_{*L} z_b}{\partial t} &= -C_{*L} i \end{aligned} \quad (\text{A.118})$$

となり、第3章の粗粒分砂礫の連続式（式 3.4）と河床位方程式（式 3.21）を得る。なお、ここでは、 i は侵食速度の場合の表示である。堆積過程では物理的な考察から $C_{*L} \rightarrow C_{*DL}$ とする。

検査体積を分割して細粒分砂礫の空間だけ考えれば、同じ要領で、 $C_L, C_{*L} \rightarrow (1 - C_L)C_F, (1 - C_{*L})C_{*F}$ と置き換えれば細粒分砂礫の連続式（式 3.5）と河床位方程式（式 3.23）を得る。

56) 後述するように河床の境界条件が異なるが、前項で $u(z_b) = v(z_b) = 0$ の下で計算すれば同じ式が得られることを確かめられる。

A.5.2.3 運動方程式の浅水流方程式

式 A.99 の浅水流方程式を導く。流れだけに着目して河床から水深まで積分すると、

$$\int_{z_b}^{z_s} \left[\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u}\mathbf{u}) \right] dz = \int_{z_b}^{z_s} \left[-\frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u} + \mathbf{K} \right] dz \quad (\text{A.119})$$

である。ここに、体積力は重力であり一様勾配 θ の斜面において斜面方向の X 軸とそれに直交する Z 軸方向に分解すると $\mathbf{K} = (g \sin \theta, 0, -g \cos \theta)^T$ である。

まず、式 A.119 の Z 方向成分を考えて、圧力が静水圧近似されることを示す。浅水流で近似できるとすれば、水深方向の慣性力（左辺）と粘性力（右辺第 2 項）が、圧力や重力に比べて十分に小さいと仮定でき、運動方程式の Z 成分は

$$0 = -\frac{1}{\rho} \frac{\partial p}{\partial z} - g \cos \theta \quad (\text{A.120})$$

となる。 ρ は非圧縮性流体だから一定として、式 A.120 を任意の水深 z から水面 z_s まで積分すると、

$$\int_z^{z_s} \frac{\partial p}{\partial z} dz = -\rho g \cos \theta \int_z^{z_s} dz \quad (\text{A.121})$$

$$p(z_s) - p(z) = -\rho g \cos \theta (z_s - z)$$

である。水面 $z = z_s$ における圧力は大気圧だから、大気圧を基準とした圧力（ゲージ圧）で考えれば $p(z_s) = 0$ である。これを境界条件として与えれば、

$$p(z) = \rho g \cos \theta (z_s - z) \quad (\text{A.122})$$

を得る。圧力を静水圧に近似できることが示された。

次に、式 A.119 の X 方向成分を考える。この時、圧力は式 A.122、せん断応力は $\tau_{jx} (= \mu \partial u / \partial x_j)$ とすれば、

$$\int_{z_b}^{z_s} \left[\frac{\partial u}{\partial t} + \frac{\partial uu}{\partial x} + \frac{\partial vu}{\partial y} + \frac{\partial wu}{\partial z} \right] dz = \int_{z_b}^{z_s} \left[-g \cos \theta \frac{\partial z_s}{\partial x} + \frac{1}{\rho} \left(\frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} \right) + g \sin \theta \right] dz \quad (\text{A.123})$$

と書ける⁵⁷⁾。左辺、右辺の順番に各項を浅水流化する。

左辺は、連続式と同様に計算すればよい。左辺第 1 項はライプニッツの公式と平均操作により

$$\int_{z_b}^{z_s} \frac{\partial u}{\partial t} dz = \frac{\partial}{\partial t} \int_{z_b}^{z_s} u dz - u_s \frac{\partial z_s}{\partial t} + u_b \frac{\partial z_b}{\partial t} \quad (\text{A.124})$$

である。ここに、 $u_s = u(z_s)$ 、 $u_b = u(z_b)$ である。第 2 項はライプニッツの公式と平均操作により

$$\int_{z_b}^{z_s} \frac{\partial uu}{\partial x} dz = \frac{\partial}{\partial x} \int_{z_b}^{z_s} u u dz - u u \frac{\partial z_s}{\partial x} + u u \frac{\partial z_b}{\partial x} = \frac{\partial (\beta_{xx} h U U)}{\partial x} - u u \frac{\partial z_s}{\partial x} + u u \frac{\partial z_b}{\partial x} \quad (\text{A.125})$$

となる。第 3 項も同様に計算できる。第 4 項は

$$\int_{z_b}^{z_s} \frac{\partial wu}{\partial z} dz = u_s w_s - u_b w_b \quad (\text{A.126})$$

である。第 1 項から第 4 項まで和をとって、積分記号のない項を u で括り、連続式の場合と同様に運動学的境界条件（式 A.109 と式 A.108）を適用すれば、

$$\frac{\partial h U}{\partial t} + \frac{\partial}{\partial x} (\beta_{xx} h U U) + \frac{\partial}{\partial y} (\beta_{yx} h U V) \quad (\text{A.127})$$

を得る。ここに、 β_{xx} 、 β_{yx} は運動量補正係数であり、実際の流速分布と水深平均した流速との違いを補正するために平均操作の際にこの係数をかける⁵⁸⁾。

次に右辺の各項について考える。右辺第 1 項（圧力項）は、ライプニッツの公式を用いて

$$\begin{aligned} -g \cos \theta \int_{z_b}^{z_s} \frac{\partial z_s}{\partial x} dz &= -g \cos \theta \left[\frac{\partial}{\partial x} \int_{z_b}^{z_s} z_s dz - z_s \frac{\partial z_s}{\partial x} + z_b \frac{\partial z_s}{\partial x} \right] \\ &= -g \cos \theta \left[\frac{\partial}{\partial x} (z_s^2 - z_s z_b) - z_s \frac{\partial z_s}{\partial x} + z_b \frac{\partial z_s}{\partial x} \right] \\ &= -g \cos \theta (z_s - z_b) \frac{\partial z_s}{\partial x} \\ &= -gh \cos \theta \frac{\partial z_s}{\partial x} \end{aligned} \quad (\text{A.128})$$

57) $z_s = z_s(t, x, y)$ であるが、 z は x の関数ではないため、 $\partial_x(z_s - z) = \partial_x z_s$ となる。ただし、 $\partial_z = \partial / \partial z$ とする。

58) ここでは、乱流モデルを導入せずに議論を展開している。多くの文献では、例えば高橋ほか (1986) のように、乱流モデルを導入し、流速を平均成分と変動成分にわけて、Boussinesq 近似や Prandtl の混合距離モデルを用いてレイノルズ応力を導入する。この時、運動量補正係数には流速の変動速度成分が含まれることになる。乱流モデルの導入しても議論の展開の筋は同様である。運動量補正係数の式は高橋ほか (1986) を参照。

である。この計算は後藤 (2022, p.329) を参考とした。 z_s は z の関数ではないから $-g \cos \theta \frac{\partial z_s}{\partial x} \int_{z_b}^{z_s} dz = -g \cos \theta \frac{\partial z_s}{\partial x} [z]_{z_b}^{z_s} = -gh \cos \theta \frac{\partial z_s}{\partial x}$ でよいと考えられる。

右辺第 2 項の応力テンソルを積分するため、後藤 (2022) および細田 (2004) を参考に、まず計算に必要な準備を行う。表面力 (ベクトル) \mathbf{t} は、コーシー関係より、応力テンソル τ_{ji} と表面力が作用する面の法線ベクトル $\mathbf{n}(=n_j)$ で記述できる。すなわち、 $t_i = \tau_{ji}n_j$ である。X 方向成分だけ考えると

$$t_x = \tau_{jx}n_j = \tau_{xx}n_x + \tau_{yx}n_y + \tau_{zx}n_z \quad (\text{A.129})$$

である。ところで、ある曲面 $f(x, y, z)$ の法線ベクトルは、曲面上の微小要素 dx に直交するベクトルである。全微分を零とした $\nabla f \cdot dx = 0$ の関係より ∇f が dx に直交するため、 ∇f を正規化したベクトルが法線ベクトルである (例えば、金谷, 2005)。従って、河床は $f_b(t, x, y, z) = z - z_b = 0$ と書けることを考えれば、河床の法線ベクトルは

$$\mathbf{n} = \frac{\nabla f_b}{\|\nabla f_b\|} = \frac{-\frac{\partial z_b}{\partial x} \mathbf{e}_1 - \frac{\partial z_b}{\partial y} \mathbf{e}_2 + \mathbf{e}_3}{\sqrt{1 + \left(\frac{\partial z_b}{\partial x}\right)^2 + \left(\frac{\partial z_b}{\partial y}\right)^2}} \quad (\text{A.130})$$

と書ける。ここに、 $\|\cdot\|$ はノルム、 $\mathbf{e}_k, (k=1, 2, 3)$ は添字と同じ成分が 1、それ以外が 0 の正規直交基底ベクトルである。従って、式 A.129 と式 A.130 より

$$(t_x)_b = \frac{-(\tau_{xx})_b \frac{\partial z_s}{\partial x} - (\tau_{yx})_b \frac{\partial z_s}{\partial y} + (\tau_{zx})_b}{\sqrt{1 + \left(\frac{\partial z_s}{\partial x}\right)^2 + \left(\frac{\partial z_s}{\partial y}\right)^2}} \quad (\text{A.131})$$

となる。河床面が急勾配でない場合、x 方向の河床勾配 $\frac{\partial z_b}{\partial x}$ と y 方向の河床勾配はきわめて小さいため分母 ≈ 1 と近似できる。ここでもそのように近似すると

$$(t_x)_b = -(\tau_{xx})_b \frac{\partial z_s}{\partial x} - (\tau_{yx})_b \frac{\partial z_s}{\partial y} + (\tau_{zx})_b \quad (\text{A.132})$$

となる。 $(t_x)_b$ は河床に作用する表面力、つまり河床せん断応力である。以上の関係は自由表面でも同様であるため (添え字 b を s に変更すればよい)、 $(t_x)_b$ を τ_{bx} と改めると、

$$\tau_{bx} = -(\tau_{xx})_b \frac{\partial z_b}{\partial x} - (\tau_{yx})_b \frac{\partial z_b}{\partial y} + (\tau_{zx})_b \quad (\text{A.133})$$

および

$$\tau_{sx} = -(\tau_{xx})_s \frac{\partial z_s}{\partial x} - (\tau_{yx})_s \frac{\partial z_s}{\partial y} + (\tau_{zx})_s \quad (\text{A.134})$$

を得る。

以上の準備をして、式 A.123 の右辺におけるせん断応力項の積分する。ライプニッツの公式 (式 A.100)、式 A.133、式 A.134、平均操作 (式 A.101) を用いると、

$$\begin{aligned} \int_{z_b}^{z_s} \frac{\partial}{\partial x_j} (\tau_{jx}) dz &= \int_{z_b}^{z_s} \left[\frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} \right] dz \\ &= \frac{\partial}{\partial x} \int_{z_b}^{z_s} \tau_{xx} dz + \frac{\partial}{\partial y} \int_{z_b}^{z_s} \tau_{yx} dz + (\tau_{zx})_s - (\tau_{zx})_b \\ &\quad - \left[(\tau_{xx})_s \frac{\partial z_s}{\partial x} + (\tau_{yx})_s \frac{\partial z_s}{\partial y} \right] + \left[(\tau_{xx})_b \frac{\partial z_b}{\partial x} + (\tau_{yx})_b \frac{\partial z_b}{\partial y} \right] \\ &= \frac{\partial}{\partial x} \int_{z_b}^{z_s} \tau_{xx} dz + \frac{\partial}{\partial y} \int_{z_b}^{z_s} \tau_{yx} dz + (\tau_x)_s - (\tau_x)_b \\ &= \frac{\partial}{\partial x} (h\bar{\tau}_{xx}) + \frac{\partial}{\partial y} (h\bar{\tau}_{yx}) + (\tau_x)_s - (\tau_x)_b \end{aligned} \quad (\text{A.135})$$

ここに、1 行目の左辺は同じ添え字の和をとるアインシュタイン総和規約を用いて略記し、最後の行のバーは平均を表す演算子である。高橋ほか (1986) に示されるように水平粘性を無視すると、式 A.135 の最終行における右辺第 1 項と 2 項は零となる⁵⁹⁾。また、吹送流 (風によって生じる流れ) 等を考えなければ、第 3 項の水面に作用する表面力 $(\tau_x)_s = 0$ となる。以上から、式 A.135 を書き直すと、

$$\int_{z_b}^{z_s} \frac{\partial}{\partial x_j} (\tau_{jx}) dz = -(\tau_x)_b \quad (\text{A.136})$$

59) ニュートン粘性の法則 (あるいはニュートン流体の構成則 (式 A.96)) より、せん断力は $\bar{\tau}_{xx} = \mu \partial u / \partial x$ 、 $\bar{\tau}_{yx} = \mu \partial u / \partial y$ であり、ともに水深方向 (河床との摩擦による速度変化) と比較して速度勾配が小さいため零に近似したものと、考えられる。

となる⁶⁰⁾。

最後に、式 A.123 の右辺における重力項を積分すると、

$$\int_{z_b}^{z_s} g \sin \theta dz = g \sin \theta (z_s - z_b) = gh \sin \theta \quad (\text{A.137})$$

となる。

以上から、ナビエ・ストークス方程式の式 A.123 の X 成分は、式 A.127 および式 A.128、式 A.136、式 A.137 より

$$\frac{\partial hU}{\partial t} + \frac{\partial}{\partial x}(\beta_x hUU) + \frac{\partial}{\partial y}(\beta_y hUV) = -gh \cos \theta \frac{\partial z_s}{\partial x} - \frac{1}{\rho}(\tau_x)_b + gh \sin \theta \quad (\text{A.138})$$

と浅水流化される。河床せん断応力を求める際に河床勾配 θ が急でないとは定したため、 $\sin \theta = \theta \approx 0$ 、 $\cos \theta = 1$ ⁶¹⁾とすれば、右辺の最後の項を無視して

$$\frac{\partial hU}{\partial t} + \frac{\partial}{\partial x}(\beta_{xx} hUU) + \frac{\partial}{\partial y}(\beta_{yx} hUV) = -gh \frac{\partial z_s}{\partial x} - \frac{1}{\rho}(\tau_x)_b \quad (\text{A.139})$$

となる。Y 方向も同様にして

$$\frac{\partial hV}{\partial t} + \frac{\partial}{\partial x}(\beta_{xy} hVU) + \frac{\partial}{\partial y}(\beta_{yy} hVV) = -gh \frac{\partial z_s}{\partial x} - \frac{1}{\rho}(\tau_x)_b \quad (\text{A.140})$$

と浅水流化される。 $z_s = z_b + h$ であるため、これらが第 3 章で用いた基礎方程式である (高橋ほか, 1986)。

ここに示した浅水流近似においては静水圧近似と流速の平均操作 (断面平均流速への近似) が要諦である。また、適用条件を考える際には緩勾配、水平粘性無視、水面せん断応力の無視などの近似をした式であることに注意が必要がある。

以上は非圧縮性のニュートン流体の式から導かれたものである。この段階では、土石流や掃流状集合流動など流砂形態による流れの特徴は考えていない。その特性をこれらの方程式に加味するのが、付録 A で述べた運動方程式における河床せん断応力 τ_b や連続式における侵食・堆積速度式 i である。なお、ここでは質量密度が一定とした導出を示したが、本来、土石流においては土砂濃度が平衡状態 $DC/Dt = 0$ でなければ一定ではないことに注意が必要である。

60) 高橋ほか (1986) や砂防・地すべり技術センター砂防技術研究所 (2000) では、式 A.123 の段階で水平粘性力を無視した。そのため、X 成分であれば、粘性力を $\nu \partial^2 u / \partial z^2$ と表現した。これを積分すると $\nu \partial u(z_s) / \partial z - \nu \partial u(z_b) / \partial z$ となり、これを $(\tau_x)_s - (\tau_x)_b$ とおいた。簡潔に水面と河床のせん断応力がそれぞれ求められ、水面のせん断応力を零とすれば、同じ式を得る。

61) テイラー展開で 2 次以上の微小項を無視する近似をすれば、 $\cos \theta = 1$ 、 $\sin \theta = \theta$ である。 θ が 6 度程度以下では $\sin \theta < 0.1$ (ラジアン単位) であるから零と近似した、と考えられる。

付録 B 関連する類似式の整理

本章では、高橋土石流理論に関する論文に現れる様々な類似した式を整理する。侵食・堆積速度式は単一粒径、混合粒径などの項目毎に概ね発表順に整理した。また、掃流状集合流動の平衡土砂濃度の式を導く。

B.1 土石流発生下限勾配等

高橋 (1977) と芦田ほか (1978) は、粘着性砂礫で河床堆積層が構成されたときの土石流発生勾配の式を導いた。芦田ほか (1978) は粘着力を零とすれば前章で示した非粘着性材料の土石流発生下限勾配と一致するように拡張された包括的な式を導いた。詳細は省略する。

高橋 (1982) は、掃流状集合流動の発生下限勾配 (式 A.5) について、掃流状集合流動が掃流砂に遷移する場合には、土砂濃度 $C^{(b)}$ 、水深 $h^{(b)}$ の掃流砂層が河床面を抑える効果があることを考慮し、 C_* を $C_* + C^{(b)}h^{(b)}/d_L$ に置き換え

$$\tan \theta = \frac{(C_* + C^{(b)}h^{(b)}/d_L)(\sigma - \rho)}{(C_* + C^{(b)}h^{(b)}/d_L)(\sigma - \rho) + \rho(1 + h_0/d)} \tan \phi \quad (\text{B.1})$$

を提案した。

B.2 土石流の侵食速度式

高橋および共同研究者により様々な侵食速度式が提案された。次のように河床堆積層の水分状態によって分類される。

- 1.飽和時の侵食速度式
- 2.不飽和時の侵食速度式
- 3.両者を総合した侵食速度式

それぞれ基本的な考え方は次の通りである。

- 1.飽和河床堆積層の場合、前章の通り、遅れ時間を伴って流速に連行して移動堆積層厚に相当する土砂が取り込まれる形で侵食速度式が表現される。
- 2.不飽和河床堆積層の場合、掃流砂量式における有効掃流力¹⁾との類推から、侵食速度が、間隙流体が河床に及ぼす無次元河床せん断応力 τ_{*f} と平衡時の無次元河床せん断応力 τ_{*fe} との差 (有効掃流力と同様)、及び摩擦速度に比例すると考えれば、 K を定数として $i/u_* = K(\tau_{*f} - \tau_{*fe})$ と表現される (Takahashi *et al.*, 1987; 高橋ほか, 1991)²⁾。
- 3.河床の飽和・不飽和は実際には不明であるため、不飽和河床堆積層の場合の式を飽和時に適用できるようにパラメータを調整する。

以上を示した後、参考として乱流型土石流の不飽和河床堆積層の侵食速度式の式形 (高橋, 2006) を示し、各式に対する侵食速度係数の数値例を示す。

B.2.1 飽和河床堆積層における侵食速度式

移動堆積層厚 a_L の表現、近似の導入の有無、遅れ時間の定義に応じていくつかの形式が提案されている。

B.2.1.1 単一粒径の侵食速度式

前章に示した高橋ほか (1999) は、

$$i = \delta_e \frac{C_\infty - C}{C_* - C_\infty} \frac{q}{d_L}$$

1) 限界掃流力以上の掃流力が河床の砂礫に作用すると移動が始まる。そのため、「有効掃流力=掃流力-限界掃流力」と定義すると、有効掃流力が零以上になれば砂礫が移動すると考えられる。

2) 高橋ほか (1991) では、 $i = K(\tau_{*f} - \tau_{*fe})\sqrt{\tau_f/\rho_m}$ と示される。 $\tau_f = \rho_m g h I = \rho_m u_*^2$ の関係式を使った。

である (式 A.36)。

川池ほか (2000) は、遅れ時間の代表長さを堆積速度式と同様に水深で定義し、流速に比例した表現である

$$i = \delta_e \frac{C_\infty - C}{C_* - C_\infty} u$$

を示した³⁾。

B.2.1.2 混合粒径の侵食速度式

高橋・匡 (1986) は、混合粒径の時の移動堆積層厚 a_L (式 A.25) を遅れ時間 T_e で除して、

$$i = \delta_e \frac{C_{T\infty} - C_L \frac{\sigma - \rho_m}{\sigma - \rho} \left[\frac{\tan \phi - \frac{C_T}{C_L} \frac{(\sigma - \rho)}{(\sigma - \rho_m)} \tan \theta}{(\tan \phi - \tan \theta)} \right]}{C_* - C_{T\infty}} \frac{q}{d_L} \quad (\text{B.2})$$

を提案した⁴⁾。ここに、 q は単位幅流量、全土砂濃度 $C_T = C_L + (1 - C_L)C_F$ 、粗粒分砂礫の土石流平衡土砂濃度 $C_{L\infty} = (\rho_m \tan \theta) / [(\sigma - \rho_m)(\tan \phi - \tan \theta)]$ 、全粒分の平衡土砂濃度 $C_{T\infty} = (\rho \tan \theta) / [(\sigma - \rho)(\tan \phi - \tan \theta)]$ (高橋・匡, 1986) である。

Takahashi (1987) (和文では高橋・匡 (1989)) は、式 B.2 に、次の関係式

$$\frac{\sigma - \rho_m}{\sigma - \rho} = \frac{(\sigma - \rho_m)(\tan \phi - \tan \theta)}{\rho_m \tan \theta} \frac{\rho \tan \theta}{(\sigma - \rho)(\tan \phi - \tan \theta)} \frac{\rho_m}{\rho} = \frac{C_{T\infty} \rho_m}{C_{L\infty} \rho} \quad (\text{B.3})$$

を代入して、 $C_{T\infty} / (C_* - C_{T\infty})$ で括り、

$$i = \delta_e \frac{C_{T\infty}}{C_* - C_{T\infty}} \left[1 - \frac{C_L \rho_m \tan \phi - \frac{C_T}{C_{T\infty}} \frac{C_{L\infty} \rho}{C_L \rho_m} \tan \theta}{\tan \phi - \tan \theta} \right] \frac{q}{d_L} \quad (\text{B.4})$$

を得た。式 B.2(高橋・匡, 1986) と式 B.4(Takahashi, 1987) は同じ式の別の表現である。

高橋・藤井 (1988) は、高橋・匡 (1986) の堆積速度式 (式 A.41 および次項参照) をそのまま侵食速度式に拡張できる、として

$$i = \delta_e \frac{C_L - C_\infty}{C_{*L}} \frac{q}{d_L} \quad (\text{B.5})$$

を示した⁵⁾。

高橋 (2004, p.128) は、前章における混合粒径の平衡土砂濃度を示す際に述べたように、単一粒径の侵食速度式 (式 A.36) において、議論を簡単にするために河床堆積層における ρ を ρ_m 、流れの土砂濃度 C を C_L と近似的に置き換えて、

$$i = \delta_e \frac{C_\infty - C_L}{C_* - C_\infty} \frac{q}{d_L}$$

と表現した (式 A.37)。

高橋 (2006, p.182) は、混合粒径の侵食速度式について、高橋ほか (1999) の単一粒径の侵食速度式と同様な手続により導出した。混合粒径の移動堆積層厚 a_L は前章に示したように

$$a_L = \frac{(\sigma - \rho_m) C_{L\infty} - C_L h}{(\sigma - \rho) C_* - C_{T\infty}}$$

となるため (式 A.31)、これを遅れ時間 $T_e = (1/\delta_e)(d_L/u)$ で割り、侵食速度式

$$i = \delta_e \frac{\sigma - \rho_m}{\sigma - \rho} \frac{C_{L\infty} - C_L}{C_* - C_{T\infty}} \frac{q}{d_L} \quad (\text{B.6})$$

を得た。これは、移動堆積層厚の導出からわかるように、式 B.2(高橋・匡, 1986) と等価な別の表現である。

3) 侵食速度式は、高橋・匡 (1986) による混合粒径の土石流の議論の中で示された。そのため、考えかた自体は高橋 (1977) 時点で示されるが、単一粒径の侵食速度式が明示された時期は、多くの混合粒径の侵食速度式より遅いようである。

4) 河床堆積層の土砂濃度は、粗粒分砂礫と細粒分砂礫に分割せず、 C_e で表現していることに注意。

5) 高橋・藤井 (1988) では分子の符号は逆である。混合粒径の侵食速度式 (ただし、河床堆積層の間に細粒分砂礫の存在は考えない設定) は式 B.2 のように導かれるが、式 B.5 のような拡張あるいは簡略化された式がいくつか存在する。第 3 章の混合粒径の侵食速度式 (式 3.10, 付録 A では式 A.37) も同様である。このような式は、多くの場合、数値計算の基礎方程式の一部として唐突に単純化した近似式として登場するため、実験で確かめられた式と比較すれば適用条件 (実験等との整合など) が十分に明らかとは言いえない、と考えられる。

B.2.2 不飽和河床堆積層における侵食速度式

Takahashi *et al.* (1987) に従って侵食速度式を導く (砂防学会 1991, p.192-193 に日本語の解説がある)。

無限長斜面を混合粒径土石流が等流で流下するとし、河床堆積物の表層が不飽和であるとする。河床面に作用するせん断応力 τ は粒子同士の衝突による応力 τ_g と間隙流体の受け持つせん断応力 τ_f の和で表されるとする。 $\tau = \rho_T gh \sin \theta$ であり、 $\tau_g = (\sigma - \rho_m) gh C_L \cos \theta \tan \phi$ (バグノルドの考え方に従う) とすれば、

$$\begin{aligned} \tau_f &= \tau - \tau_g \\ &= \rho_m gh \sin \theta \left[\frac{\rho_T}{\rho_m} - \frac{\sigma - \rho_m}{\rho_m} C_L \frac{\tan \phi}{\tan \theta} \right] \\ &= \rho_m gh \sin \theta \left[1 - \frac{\sigma - \rho_m}{\rho_m} C_L \left(\frac{\tan \phi}{\tan \theta} - 1 \right) \right] \end{aligned} \quad (\text{B.7})$$

となる。ここに、 $\rho_T = C_L \sigma + (1 - C_L) \rho_m$, $\rho_m = (\sigma - \rho_m) C_L + \rho_m$ を用いた。この式は粗粒分土砂濃度が増加すると間隙流体のせん断応力が減少することを表す。

さて、流れに十分な間隙があると河床面の土砂を取り込めるが、掃流状集合流動のような集合流動では、掃流砂流のように水が十分という訳ではないため、限界掃流力以上の掃流力が河床に作用しても、流れの土砂濃度がある値に達していると、土砂を取り込めなくなる (侵食できない)。つまり、ある最大土砂濃度のとき、間隙流体のせん断応力が限界値 τ_{fe} に達する (この式の形は τ_f と同様と仮定する)。侵食ができなくなる τ_{fe} のときの土砂濃度の表し方は不明であるため、全粒分の平衡土砂濃度 $C_{T\infty} = \rho \tan \theta / (\sigma - \rho) (\tan \phi - \tan \theta)$ になると仮定する。このとき、式 B.7 で C_L を $C_{T\infty}$ に置き換えられるとすれば

$$\tau_{fe} = \tau|_{C=C_{T\infty}} = \rho_m gh \sin \theta \left[1 - \frac{\sigma - \rho_m}{\rho_m} C_{T\infty} \left(\frac{\tan \phi}{\tan \theta} - 1 \right) \right] \quad (\text{B.8})$$

と表現できる。このとき、土砂濃度 $C_{T\infty}$ は、 $\tau_f = 0$ として式 B.7 から得られる粗粒分砂礫の平衡土砂濃度 $C_{L\infty} = \rho_m \tan \theta / (\sigma - \rho_m) (\tan \phi - \tan \theta)$ より小さい (間隙流体のせん断応力が上昇すると粗粒分土砂濃度が減少するため)。よって、 $C_L < C_{T\infty}$ で侵食、 $C_{T\infty} < C_L < C_{L\infty}$ で平衡、 $C_{L\infty} < C_L$ で堆積となる。侵食の場合、不飽和侵食速度式の想定される形式が

$$\frac{i}{u_{*f}} = K(\tau_{*f} - \tau_{*fe}) \quad (\text{B.9})$$

であるとすれば、式 B.7 と $u_{*f} = \sqrt{\tau_f / \rho_m}$, $\tau_{*f} = \tau_f / [(\sigma - \rho_m) g d_L]$, $\tau_{*fe} = \tau_{fe} / [(\sigma - \rho_m) g d_L]$ を式 B.9 に代入すると、

$$\begin{aligned} i &= K u_{*f} \left(\frac{\tau_f}{(\sigma - \rho_m) g d_L} - \frac{\tau_{fe}}{(\sigma - \rho_m) g d_L} \right) \\ &= K \sqrt{\frac{\tau_f}{\rho_m}} \frac{1}{(\sigma - \rho_m) g d_L} (\tau_f - \tau_{fe}) \\ &= K \sqrt{\frac{\tau_f}{\rho_m}} \frac{\rho_m gh \sin \theta}{(\sigma - \rho_m) g d_L} \left[\frac{\sigma - \rho_m}{\rho_m} (C_{T\infty} - C_L) \left(\frac{\tan \phi}{\tan \theta} - 1 \right) \right] \\ &= K \sqrt{gh \sin \theta} \left[1 - \frac{\sigma - \rho_m}{\rho_m} C_L \left(\frac{\tan \phi}{\tan \theta} - 1 \right) \right]^{1/2} \frac{\rho_m gh \sin \theta}{(\sigma - \rho_m) g d_L} \left[\frac{\sigma - \rho_m}{\rho_m} (C_{T\infty} - C_L) \left(\frac{\tan \phi}{\tan \theta} - 1 \right) \right] \\ &= K \frac{\rho_m}{\sigma - \rho_m} \frac{gh \sin \theta}{g d_L} \sqrt{\sin \theta} \sqrt{gh} \left[1 - \frac{\sigma - \rho_m}{\rho_m} C_L \left(\frac{\tan \phi}{\tan \theta} - 1 \right) \right]^{1/2} \frac{\sigma - \rho_m}{\rho_m} (C_{T\infty} - C_L) \left(\frac{\tan \phi}{\tan \theta} - 1 \right) \end{aligned} \quad (\text{B.10})$$

になるから、整理すれば、

$$\frac{i}{\sqrt{gh}} = K \sin^{3/2} \theta \left[1 - C_L \left(\frac{\sigma - \rho_m}{\rho_m} \right) \left(\frac{\tan \phi}{\tan \theta} - 1 \right) \right]^{1/2} \left(\frac{\tan \phi}{\tan \theta} - 1 \right) (C_{T\infty} - C_L) \frac{h}{d_L} \quad (\text{B.11})$$

が得られる (Takahashi *et al.*, 1987)。

高橋ほか (1991) は、この式 B.11 における右辺最後の丸括弧の部分 ($C_{T\infty} - C_L$) を粗粒分砂礫の平衡土砂濃度に変更し ($C_{L\infty}^{(d)} - C_L$) とした。なぜなら、式 B.7 において、元々等価動摩擦係数 $\tan \alpha$ を値の大きな静止内部摩擦係数 $\tan \phi$ で置き換えているため、実際の流れにおける平衡土砂濃度より、この式では小さい平衡土砂濃度を用いて評価している (付録 A における単一粒径の平衡土砂濃度の脚注参照)。そのため、実際の平衡土砂濃度がここに示す式より大きいとすれば、 τ_f は実際より小さく評価される。従って、 $C_{L\infty}$ (これは元は $\tau_f = 0$ で求まる土砂濃度) のとき $\tau_{fe} = 0$ とは限らない。急勾配な箇所では土石流が発達できるとすれば

$C_{T\infty} < C_L < C_{L\infty}$ で平衡とする条件を付与する必要はなく $C_L < C_{L\infty}$ で侵食が生じると考えられるためである。従って、Takahashi *et al.* (1987) に示された侵食速度式 (式 B.11) は

$$\frac{i}{\sqrt{gh}} = K \sin^{3/2} \theta \left[1 - C_L \left(\frac{\sigma - \rho_m}{\rho_m} \right) \left(\frac{\tan \phi}{\tan \theta} - 1 \right) \right]^{1/2} \left(\frac{\tan \phi}{\tan \theta} - 1 \right) (C_{L\infty} - C_L) \frac{h}{d_L} \quad (\text{B.12})$$

と修正された (高橋ほか, 1991)。粗粒分砂礫の土砂濃度 C_L が河床堆積層の充填土砂濃度 C_* を超え、場合によっては C_* を超える値を持つように計算されるが、このような高濃度では流動できない。そのため、 $0.9C_* < C_L$ を満たす場合、 $i = 0$ である⁶⁾ (高橋ほか, 1991)。

高橋 (2004, p.128) は、この式 B.12 と同様の考察によって次のように掃流状集合流動の場合に導いた。土石流発生下限勾配 (式 A.12) 以下の河床勾配の範囲で掃流状集合流動が十分に発達する場合、図 A.1 (c) に示す設定により、間隙流体のせん断応力は、式 B.7 において底面付近の砂礫移動層における流動深 h_l と土砂濃度 C_l を用いて

$$\tau_f = [(\sigma - \rho_m)C_l h_l + \rho_m h] g \sin \theta - (\sigma - \rho_m)C_l h_l g \cos \theta \tan \phi \quad (\text{B.13})$$

と表される⁷⁾。ここで、 $C_L^{(s)}$ を掃流状集合流動の全層一様な土砂濃度とすると、 $C_l h_l = C_L^{(s)} h$ の関係 (後述式 B.32 参照; その脚中に示すように $\tau_* = 0$ の時の関係式) が成り立つため、代入すると

$$\begin{aligned} \tau_f &= [(\sigma - \rho_m)C_L^{(s)} h + \rho_m h] g \sin \theta - (\sigma - \rho_m)C_L^{(s)} h g \cos \theta \tan \phi \\ &= \rho_m g h \sin \theta \left[1 - \frac{\sigma - \rho_m}{\rho_m} C_L^{(s)} \left(\frac{\tan \phi}{\tan \theta} - 1 \right) \right] \end{aligned}$$

を得る。これは、式 B.7 において土砂濃度 $C_L^{(d)}$ を $C_L^{(s)}$ に置き換えた式である。従って、土石流の時と全く同じ議論が成立する。よって、 τ_{fe} (式 B.8) および侵食速度式 (式 B.12) において、 $C_{T\infty}, C_L$ をそれぞれ $\hat{C}_{L\infty}^{(s)}, C_L^{(s)}$ に置き換えられる。以上から、掃流状集合流動の不飽和堆積層における侵食速度式

$$\frac{i^{(s)}}{\sqrt{gh}} = K \sin^{3/2} \theta \left[1 - C_L^{(s)} \left(\frac{\sigma - \rho_m}{\rho_m} \right) \left(\frac{\tan \phi}{\tan \theta} - 1 \right) \right]^{1/2} \left(\frac{\tan \phi}{\tan \theta} - 1 \right) (\hat{C}_{L\infty}^{(s)} - C_L^{(s)}) \frac{h}{d_L} \quad (\text{B.14})$$

が導かれる (高橋, 2004, 式 3.39)。

高橋 (2006, p.181) は、混合粒径における土石流の式 B.12 と掃流状集合流動の式 B.14 を統一的に表現した。平衡土砂濃度 $C_{L\infty}^{(d)}, C_{L\infty}^{(s)}$ を流砂形態によらず C_{eq} と表現し、また、

$$\frac{\sigma - \rho_m}{\rho_m} \left(\frac{\tan \phi}{\tan \theta} - 1 \right) = \frac{\sigma - \rho_m}{\rho_m} \frac{\tan \phi - \tan \theta}{\tan \theta} = \frac{1}{C_{L\infty}^{(d)}} \quad (\text{B.15})$$

の関係を用いて、

$$\frac{i}{\sqrt{gh}} = K \sin^{3/2} \theta \left[1 - \frac{C_L}{C_{L\infty}^{(d)}} \right]^{1/2} \left(\frac{\tan \phi}{\tan \theta} - 1 \right) (C_{eq} - C_L) \frac{h}{d_L} \quad (\text{B.16})$$

と統一的に表現した (高橋, 2006, p.181)。

高橋・中川 (1992) は、土石流の式 B.12 と掃流状集合流動の式 B.14 において、 $\rho_m \rightarrow \rho$ と形式に変換して、以下の単一粒径の式をそれぞれ示した⁸⁾。

$$\frac{i^{(d)}}{\sqrt{gh}} = K \sin^{3/2} \theta \left[1 - C_L \left(\frac{\sigma - \rho}{\rho} \right) \left(\frac{\tan \phi}{\tan \theta} - 1 \right) \right]^{1/2} \left(\frac{\tan \phi}{\tan \theta} - 1 \right) (C_{L\infty}^{(d)} - C_L) \frac{h}{d_L} \quad (\text{B.17})$$

$$\frac{i^{(s)}}{\sqrt{gh}} = K \sin^{3/2} \theta \left[1 - C_L \left(\frac{\sigma - \rho}{\rho} \right) \left(\frac{\tan \phi}{\tan \theta} - 1 \right) \right]^{1/2} \left(\frac{\tan \phi}{\tan \theta} - 1 \right) (C_{L\infty}^{(s)} - C_L) \frac{h}{d_L} \quad (\text{B.18})$$

高橋ほか (2000) は、河床堆積層における粒径別の侵食速度式を示した。まず記号を定義する。河床変動計算と同様に、河床堆積層における粒径分布を持つ混合粒径を粒径階 k_i ($i = 1, \dots, F, F+1, \dots, L$) に区分する。粒径階は粒径が小さい方から昇順にならべ、 $k_i = k_1, \dots, k_F$ が細粒分砂礫、 $k_i = k_{F+1}, \dots, k_L$ が粗粒分砂礫を表すものとし、各粒径階の存在率を $f(k_i)$ で表す。このとき、粗粒分砂礫の全粒径階の存在率は $f_L^{all} = \sum_{k_i=k_{F+1}}^{k_i=k_L} f(k_i)$ である。水深 h の時に、河床堆積層ではその水深に相当する粒径が移動可能であるとすれば、粗粒分砂礫の粒径 $d_{k_h} < h < d_{k_{h+1}}$ を満たす $k_h (> k_{F+1})$ より小さい粒径階の粗粒分砂礫が移動可能

6) $0.9C_*$ は、実験に基づく流動可能な最大土砂濃度である (高橋・八木, 1983)。

7) この式は高橋 (2004, 式 3.38) であるが、右辺第 2 項に h 、第 3 項に g が抜けていたため修正した。

8) 混合粒径の式との関連を考慮して原著論文における C を C_L とする。なお、前章の混合粒径の平衡土砂濃度の項に示したように、間隙流体密度を単純に変更して単一粒径の式にできない場合がある。この式が代数的に等価か近似式か、筆者は追跡していない。

である。粗粒分砂礫のある一つの粒径階の存在率は $f_L^i = f(k_i)/f_L^{all}$ で表されるから、移動可能な粗粒分砂礫の複数の粒径階 $k_i = k_{F+1}, \dots, k_h$ の存在率は $f_L^{k_h} = \sum_{k_i=k_{F+1}}^{k_i=k_h} f(k_i)/f_L^{all}$ である。また、平均粒径 \bar{d}_L は、粗粒分砂礫の粒径階 $d_L(k_i)$ と存在率 $f(k_i)$ の積、つまり平均値として $\bar{d}_L = \sum_{k_i=k_{F+1}}^{k_i=k_L} [d_L(k_i)f(k_i)]$ と表せる。

さて、土石流の場合について、粒径別侵食速度を求める前に、まずある一つの粒径階 k_i の侵食速度式を求める。式 B.16 の $\tan \phi$ と $\tan \theta$ を消去するように式 B.15 を用いて変形すると、

$$\begin{aligned} \frac{i}{\sqrt{gh}} &= K \sin^{3/2} \theta \frac{\rho_m}{\sigma - \rho_m} \frac{\sigma - \rho_m}{\rho_m} \left(\frac{\tan \phi}{\tan \theta} - 1 \right) \left[\frac{C_{L\infty}^{(d)} - C_L}{C_{L\infty}^{(d)}} \right]^{1/2} (C_{L\infty}^{(d)} - C_L) \frac{h}{\bar{d}_L} \\ &= K \sin^{3/2} \theta \frac{\rho_m}{\sigma - \rho_m} \frac{1}{C_{L\infty}^{(d)}} \left[\frac{C_{L\infty}^{(d)} - C_L}{C_{L\infty}^{(d)}} \right]^{1/2} (C_{L\infty}^{(d)} - C_L) \frac{h}{\bar{d}_L} \\ &= K \sin^{3/2} \theta \frac{\rho_m}{\sigma - \rho_m} \left[\frac{C_{L\infty}^{(d)} - C_L}{C_{L\infty}^{(d)}} \right]^{3/2} \frac{h}{\bar{d}_L} \end{aligned} \quad (B.19)$$

を得る (高橋ほか, 2000, 式 18)。

次に、水深 h における混合粒径の粒径別粗粒分砂礫の侵食速度式を求める。式 B.19 に水深 h で侵食可能な粗粒分砂礫の存在率 $f_L^{k_h}$ を乗じると、その状況で侵食され得る粗粒分砂礫の最大のみかけの侵食速度となる。さらに、粒径別にするためにはある粒径階の粗粒分砂礫の存在率 f_L^i を乗じればよい。最後に、河床から流れに取り込まれるみかけの侵食速度を実質の土砂量に変換するため、河床堆積層の充填土砂濃度 C_{*L} を乗じる。これによって、水深 h の時に河床から土砂に取り込まれる粗粒分砂礫の粒径別侵食速度式が求められ、

$$\frac{i(k_i)}{\sqrt{gh}} = C_{*L} f_L^i f_L^{k_h} \frac{i}{\sqrt{gh}} = C_{*L} f_L^i f_L^{k_h} K \sin^{3/2} \theta \frac{\rho_m}{\sigma - \rho_m} \left[\frac{C_{L\infty}^{(d)} - C_L}{C_{L\infty}^{(d)}} \right]^{3/2} \frac{h}{\bar{d}_L} \quad (B.20)$$

を得る⁹⁾ (高橋ほか, 2000, 式 21)。

側岸侵食への適用は、高橋 (2006) や高橋・中川 (1992) などを参照されたい。

B.2.3 両者を総合した侵食速度式

高橋 (2006, p.182) は、河床堆積層の飽和・不飽和の状態の事前予測は困難であることから、不飽和の侵食速度式を飽和の侵食速度式として適用できるか検討し、不飽和時は約 $K = 0.06$ 、飽和時には約 $K = 2.3$ とすれば実験結果と調和的であることを示した。

B.2.4 乱流型土石流の不飽和河床堆積層における侵食速度式

高橋 (2006, p.183) は、特に乱流型土石流の不飽和堆積層における侵食速度式については、河床底面から粒子が浮遊するには流れの上向きの乱れ速度が粒子の沈降速度より大きいほど容易になることを考慮して

$$i = K_t \frac{C_{eq} - C}{C_*} |u_* - w_s| \quad (B.21)$$

の表現を提案した。ここに K_t は係数、 w_s は粒子の沈降速度である。この式の適用事例があるか文献を追跡できていない。

B.2.5 侵食速度係数

B.2.5.1 飽和侵食速度式に対する侵食速度係数

侵食速度係数は、実験で決める値であり、提案された式に依存すると考えられる。また、代表長さが多くの式は粒径であるが、川池ほか (2000) では堆積速度式と同様に流動深 h である。本項では、文献に記される侵食速度係数 δ_e の値をいくつか示す。

式 B.2 (高橋ほか, 1986) で初めて侵食速度係数が導入され¹⁰⁾、実験により最も適当な値を $\delta_e = 0.0007$ とした。ただし、同論文における泥流の計算では、同じ侵食速度式に対して 0.0001 を与えた。

式 B.4 (Takahashi, 1987; 高橋ほか, 1986) に対して、高橋ほか (1990) では、 $\delta_e = 0.0007, 0.003$ である。後者の値の理由は不明であるが、その後、高橋ほか (1991) において粒径選別の結果を土石流発達過程の計算によりうまく導入できるように改良したことで $\delta_e = 0.003$ とする必要はないことが判明し、高橋ほか (1991) は

9) この部分では記号の定義が高橋ほか (2000, 式 21) と異なる。式 B.19 における i は、高橋ほか (2000, 式 18) では i_{sbo} である。

式 B.20 における $i(k_i)$, f_L^i , $f_L^{k_h}$ は、高橋ほか (2000, 式 21) ではそれぞれ i_{sbk} , f_{bLk} , K_3 である。

10) この時は名称はなく係数として α の記号で示された。

$\delta_e = 0.0007$ とした。中川ほか (1996) は、侵食速度係数がある範囲で変化させて数値計算の再現計算による感度分析の結果、 δ_e の流出土砂量への影響は少ないことを示し、以後 $\delta_e = 0.0007$ を用いた。

式 B.5(高橋・藤井, 1988) の式に対して、同論文では $\delta_e = 0.05$ である。

式 A.36(高橋ほか, 1999) の式に対して、同論文では $\delta_e = 0.0007$ である。

式 A.37(川池ほか, 2000) の式に対して、同論文では $\delta'_e = 0.0007$ である¹¹⁾。

式 B.6(高橋, 2006, p.182) の式に対して、同論文では $\delta_e = 0.003$ である。

B.2.5.2 不飽和侵食速度式に対する侵食速度係数

式 B.11(Takahashi *et al.*, 1987) の式に対して、同論文では実験結果により $K = 0.06$ である。中川ほか (1996) は、数値計算の再現計算による感度分析を行い、 $K = 0.001 \sim 0.005$ では実際と合わず $K = 0.025, 0.05$ の時に実態と一致し、後者 $K = 0.05$ は実験で同定された値、と述べた。

B.3 土石流の堆積速度式

高橋および共同研究者により提案された堆積速度式を示す。考え方は前章の「堆積速度式」に詳述した。ここでは、単一粒径、混合粒径の式を概ね時系列的に整理するとともに、土石流の堆積速度式を準用した掃流状集合流動と掃流砂の堆積速度式も併せて掲載する。

B.3.1 単一粒径の堆積速度式

高橋ほか (1999) は、高橋・匡 (1986) の式 A.41 と同様の形式の単一粒径の堆積速度式

$$i = \delta_d \frac{C_\infty - C}{C_*} \frac{q}{d_L} \quad (\text{B.22})$$

を示した。分母が河床堆積層の充填土砂濃度であり、他の式のように土石流の離脱土砂による河床堆積層土砂濃度ではないことに注意が必要である¹²⁾。

B.3.2 混合粒径の堆積速度式

高橋・匡 (1986) は、前章に示した混合粒径の堆積速度式 (式 A.41)

$$i = \delta_d \frac{C_{L\infty} - C_L}{C_{*DL}} \frac{q}{d_L}$$

を提案した。

高橋ほか (1991) は、さらに、土石流が扇状地に氾濫する状況を考え、遷緩点を通過した土石流は粒分砂礫を分散させるのに十分な流速をもっていれば、平衡土砂濃度より土砂濃度が大きくても堆積は直ちに始まらず減速してから堆積が始まる事実を考慮するため、慣性を考慮した流速の制御因子 pU_e を用いた堆積速度式

$$i^{(d)} = \delta'_d \left(1 - \frac{u}{pU_e}\right) \frac{C_{L\infty}^{(d)} - C_L}{C_{*DL}} u \quad (\text{B.23})$$

を提案した。ここに、 U_e は堆積・侵食せずに流下を続けられる平衡流速 (後述)、 p は 1 より小さい定数、 θ_e は平衡濃度を流れの土砂濃度 C_L として実現する平衡勾配、 δ'_d は遅れ時間の代表長さに水深を用いた時の堆積速度係数である¹³⁾。平衡流速は石礫型土石流の水深平均流速、平衡勾配は平衡土砂濃度式を $\tan \theta$ について解いた式であり、それぞれ

$$U_e = \frac{2}{5d_L} \left[\frac{g \sin \theta_e}{0.02} \left\{ C_L + (1 - C_L) \frac{\rho_m}{\sigma} \right\} \right]^{1/2} \left\{ \left(\frac{C_{*DL}}{C_L} \right)^{1/3} - 1 \right\} h^{3/2} \quad (\text{B.24})$$

$$\tan \theta_e = \frac{C_L(\sigma - \rho_m)}{C_L(\sigma - \rho_m) + \rho_m} \tan \phi$$

11) 代表長さを流動深として遅れ時間を定義した場合、堆積速度係数の表記と同様に ν を付けることとした。

12) 土石流から離脱した後の河床堆積層の充填土砂濃度 C_{*DL} は流動中の分散相 (固相) に対して定義されるはずであり、単一粒径でも混合粒径でも成立するはずである。どのような条件の時に河床堆積層の土砂濃度を用いるかは、本資料のための調査では追跡できなかった。

13) 高橋 (2004, p.129) によると、堆積時には遅れ時間の代表長さは代表粒径 d_L より水深 h を用いることが適当である。それぞれ、 $T_d = (1/\delta_d)(d_L/U)$ 、 $T'_d = (1/\delta'_d)(h/U)$ である。堆積速度式が同じであれば、 $\delta'_d = \delta_d(h/d_L)$ の関係がある (高橋, 2006, p.184)。

である（高橋ほか 1991, 式 18；式 A.54 参照）。 $U > pU_e$ であれば $U = pU_e$, $U < pU_e$ であれば $i = 0$ とする¹⁴⁾（高橋, 2006, p.184）。なお, この堆積速度式は, 遅れ時間の代表長さに水深を用いており, 流速に比例する表現である。堆積時の遅れ時間の定義について, 高橋ほか (1991) は「堆積に必要な時間（著者注：本資料における堆積に関する遅れ時間 T_d のこと）を粒径と流速を用いて表すなど, 少し異なった形で与えてきたが, ここで提案しているような堆積速度式の方がよいようである。」とした。その後の文献において, 高橋 (2004, p.129) では, 遅れ時間には曖昧な点もあるが, 堆積時の遅れ時間の代表長さには代表粒径より水深を用いることが妥当かもしれない, と述べる。

高橋・中川 (1991) は, 高橋ほか (1991) の式 B.23 に多少修正を加え,

$$i = \delta_d \left(1 - \frac{u}{pU_e}\right)^4 \frac{C_{L\infty}^{(d)} - C_L}{C_{*DL}} \frac{q}{d_L} \quad (\text{B.25})$$

のように慣性の効果を強めるために指数を変更した式を提案した¹⁵⁾。慣性的な減速運動を考慮する堆積速度式について, 中川ほか (1996) や高橋 (2006, p.183-184) 等その後の文献を見ると, 1 乗の式 (式 B.23) を用いている。

高橋ほか (2000) は, 高橋・匡 (1986) の混合粒径の堆積速度式を粒度分布における粒径階に拡張した（考え方は侵食速度式の式 B.20 と類似する。ここでは省略する）。

中川ほか (2002) は, 前章に示したように, 上に示した (高橋・匡, 1986) の混合粒径の堆積速度式 (式 A.41) において, 高橋ほか (1991) と同様に遅れ時間の代表長さを水深とし

$$i = \delta'_d \frac{C_{L\infty} - C_L}{C_{*DL}} u$$

を提案した (式 A.44)

B.3.3 他の流砂形態の堆積速度式

土石流の堆積速度式を準用する形で提案された他の流砂形態の堆積速度式を整理する。

B.3.3.1 掃流状集合流動の堆積速度式

高橋ほか (1987) は, 高橋・匡 (1986) による式 B.22 を混合粒径で表現した上で, 形式的に $C_{L\infty}^{(d)}$ を掃流状集合流動の平衡土砂濃度 $\hat{C}_{L\infty}^{(s)}$ として, 掃流状集合流動の侵食速度式

$$i^{(s)} = \delta_d \frac{\hat{C}_{L\infty}^{(s)} - C_L}{C_{*DL}} \frac{q}{d_L} \quad (\text{B.26})$$

を示した。

高橋ほか (1991) は, 高橋ほか (1987) と同じ考え方をするが, 遅れ時間の代表長さを水深で定義して流速に比例する掃流状集合流動の堆積速度式

$$i^{(s)} = \delta'_d \frac{\hat{C}_{L\infty}^{(s)} - C_L}{C_{*DL}} u \quad (\text{B.27})$$

を示した。

B.3.3.2 掃流砂流の堆積速度式

中川ほか (1998) は, 高橋ほか (1991) による掃流状集合流動の平衡土砂濃度を, 掃流砂流の平衡土砂濃度 $\hat{C}_{\infty}^{(b)}$ に置き換えて, 掃流砂流に適用し

$$i^{(b)} = \delta''_d \frac{\hat{C}_{L\infty}^{(b)} - C_L}{C_{*DL}} u \quad (\text{B.28})$$

を提案した。

14) 高橋ほか (1991) によれば, この式を提案する以前に発表した Takahashi and Nakagawa (1989) において, 「堆積に必要な時間を粒径と流速を用いて表すなど, 少し異なった形」の堆積速度式を提案した, という。文献を入手できず, どのような式か調べられなかった。

15) この式 B.25 について, 高橋・中川 (1991) は「実験によれば堆積は直ちに過剰な土砂をその場に置いて行くのではなく, 多少の慣性的な流動の後, 限界速度よりもかなり遅くなった地点から下流のある距離にかけて行われる。著者らが先に提案した堆積速度式はこのことを十分に反映するようにはなっていないだったので, ここでは多少修正を加え」と述べる。ここで, 高橋・中川 (1991) には「先に提案した堆積速度式」を示す引用文献は明示されていない。高橋ほか (1991) がこれに該当すると考えて, 本文に引用した。

B.3.4 堆積速度係数

高橋 (2006, p.184) によると, 石礫型土石流の場合, $\delta_d = 0.05 \sim 0.1, p = 1/3 \sim 2/3$, 一方 $\delta'_d = 1.0$ である。それぞれの堆積速度係数のオーダーは, 相対水深 $h/d_L = 10.0$ で代表させると, $\delta'_d = \delta_d(h/d_L)$ から理解できる (高橋, 2006, p.184)。以下, 文献に記される値をいくつか示す。

式 A.41(高橋・匡, 1986) で, 同論文で提案された混合粒径の堆積速度式に関する係数として初めて堆積速度係数が導入され¹⁶⁾, 実験から最も適当な値は 0.0007 とした。

式 B.26(高橋ほか, 1987) に対して, 高橋・藤井 (1988) では $\delta_d = 0.05$ である。

式 B.27(高橋ほか, 1991) に対して, 同論文では $\delta'_d = 0.00001$, 川池ほか (2000) では $\delta'_d = 0.001$ である。

式 B.23(高橋ほか, 1991) に対して, 同論文では $p = 2/3, \delta'_d = 1.0$ である。

式 B.25(高橋・中川, 1991) に対して, 同論文では $p = 1/3, \delta_d = 0.5$ である。

B.4 掃流状集合流動の平衡土砂濃度

水山 (1980) は, 河床勾配 $0.05 < \tan \theta < 0.25$ かつ掃流力が限界掃流力より十分大きい範囲において, 掃流状集合流動の平衡土砂濃度

$$\hat{C}_\infty^{(s)} = \frac{q^{(s)}}{q} = 5.5(\tan \theta)^2 \quad (\text{B.29})$$

を実験的に導いた。ここに, $q^{(s)}$ は流砂量, q は流量である。この式は, 初めに提案された掃流状平衡土砂濃度の式であり, 第 3 章における図 3.9 の右図に示した。

高橋 (1982) および Takahashi (1991) は, 理論的, 実験的に掃流状集合流動の平衡土砂濃度を導き, この実験式との関係も考察した (図 3.9 参照)。以下, 高橋 (1982) を基に第 3 章で示した掃流状集合流動の平衡土砂濃度を導くことを試みる。導いた後に, 高橋および共同研究者による類似の表現についても整理する。

ここでは, 勾配 θ の無限長斜面の飽和堆積層上に, 掃流状集合流動 (下層に流動深 $h_l (\leq h)$ の砂礫移動層, 上層に $h - h_l$ の水流層をもつ) が等流で流下しており, 河床を侵食も堆積もしない状態を問題とする (図 A.1 (c) 参照)。掃流状集合流動下層部において, 砂礫移動層と全層に関する流動深と断面平均流速をそれぞれ h_l, h, U_l, U とし, 掃流状集合流動の断面平均土砂濃度, 平衡 (輸送) 土砂濃度, 土石流の下限土砂濃度をそれぞれ $C_l, C_\infty^{(s)}, C_c^{(d)}$ とする¹⁷⁾¹⁸⁾。なお, $C_L^{(s)}$ は流れ全体の土砂濃度であるため, 下層の砂礫移動層の土砂濃度 C_l とは異なる。

求めたい平衡土砂濃度 $C_\infty^{(s)}$ は, ある断面の流砂量と全流量をそれぞれ $q^{(s)}, q$ とすれば, $\hat{C}_\infty^{(s)} = q^{(s)}/q$ で定義される輸送濃度である。

B.4.1 平衡と応力のつり合い, 水深の関係式

河床に作用するせん断応力は, 河床面上の物体が重力により流下する力に等しいため,

$$\tau = (\sigma - \rho)C_l h_l g \sin \theta + \rho g h \sin \theta \quad (\text{B.30})$$

と書ける。一方, 砂礫移動層が河床に及ぼすせん断力 τ_L は, 粒子の衝突応力 (バグノルドの考え) τ_b と間隙流体のせん断応力の和となる。後者の間隙流体のせん断応力について, 平衡状態の問題設定において, 砂礫移動層の流れと土石流との類似性を考えると, 式 A.18 における高橋 (1977) の解釈に示されるように, 河床表層

16) このときは名称はなく係数として β として示された。

17) 掃流状集合流動における砂礫移動層の流動深と断面平均流速 h_l, U_l は下層 lower を示す l を下添え字に用いている。この記号は高橋 (1982) では h_d, U_d と記される。その後の文献 (例えば Takahashi 1991) の記法に従った。また, 土石流と掃流状集合流動の境界土砂濃度は, 本資料の他章では C_{dc} と書かれるものである。本章では導入した流砂形態に応じた記号を上添え字に示し, 下添え字に臨界 critical を表す c を用いて, $C_c^{(d)}$ と記す。なお, 掃流状集合流動と掃流砂流の境界土砂濃度は実験から 0.02 程度であることが示され, コードではこれにならって $C_{sc} = C_c^{(s)}$ の記号を用いる。

18) 掃流状集合流動は勾配が緩くなり土石流が下層だけを流れるようになった時に形成される流砂形態である。掃流状集合流動の発生時には砂礫移動層はほぼ水面に位置し, その砂礫移動層の土砂濃度 (掃流状集合流動の上限土砂濃度) は, 土石流の平衡土砂濃度 $C_\infty^{(d)}$ に土石流発生の下限勾配式 A.12 を代入すると $C_c^{(d)}$ になると考えて, 計算すると,

$$C_c^{(d)} = \frac{C_*}{1 + \kappa^{-1}}$$

を得る (第 3 章における河床せん断応力に関する流砂形態の遷移の境界土砂濃度 $0.4C_{*DL}$ は $\kappa = 0.7$ の値をまるめたもの)。高橋 (1982) では, 砂礫移動層が薄くなってもこの層における土砂濃度はほとんど変化しないと仮定して, $C_l = C_c^{(d)}$ と書く (後述するように入侵・堆積が生じない平衡状態で成立)。本節では高橋 (1982) における $C_c^{(d)}$ の表記を C_l のまま記している。

付近の各個運搬である掃流砂流と同様の動きとなるため、河床表面の代表粒径 d_L に対する限界掃流力 τ_c と等しいと考えられる。従って、

$$\tau_L = \tau_b + \tau_c = (\sigma - \rho)C_l h_l g \cos \theta \tan \phi + \tau_c \quad (\text{B.31})$$

と書ける。この両式を等値して計算すると、

$$\frac{h_l}{h} = \frac{C_\infty^{(d)}}{C_l} \left(1 - \frac{\tau_{*c}}{\tau_*}\right) \quad (\text{B.32})$$

となる¹⁹⁾。砂礫流動層の流動深の全流動深に対する割合が土砂濃度で表現できる²⁰⁾。この式において、土石流から掃流状集合流動に遷移する瞬間を考える。砂礫が水面まで分散しているため $h_l = h$ であり、平衡状態の問題設定であるため土砂濃度は $C_l = C_\infty^{(d)}$ となる。このとき、式 B.32 によれば $\tau_c = 0$ とならなければ説明できない(高橋 1982; Takahashi 1991, p.42)。つまり、砂礫移動層の応力は砂礫の分散応力で受け持たれていると解釈できる。

Takahashi (1991) では、この段階で $\tau_c = 0$ として以後の議論を展開する。しかし、ここでは、提案された様々な式との関係を整理したいため、高橋 (1982) と同様に明示したまま式を展開する。

B.4.2 抵抗係数と流砂量との関係式

まず、平衡土砂濃度の定義

$$\hat{C}_\infty^{(s)} = \frac{q^{(s)}}{q} = \frac{C_l h_l U_l}{hU} \quad (\text{B.33})$$

を変形して、抵抗係数と流砂量の関係式を求める。

掃流状集合流動の流速係数(抵抗則)は相対水深に比例すると近似でき²¹⁾、式 A.62 より、断面平均流速 $U = u_* R(h/d_L)$ となる。一方、無次元流砂量の定義 $q_*^{(s)} = q^{(s)}/(u_* d)$ より $q^{(s)} = q_*^{(s)} u_* d$ となる。この2式を、式 B.33 の分母と分子にそれぞれ代入すると、

$$\hat{C}_\infty^{(s)} = R^{-1} \left(\frac{h}{d_L}\right)^{-2} q_*^{(s)} \quad (\text{B.34})$$

を得る。

B.4.3 平衡土砂濃度の導出

高橋 (1982) は、掃流状集合流動における流速分布や抵抗係数を理論的に導いた(詳細は原著論文)。この理論式を単純化するために、 h_l/h が 0~1 の範囲において、実験結果を参考に、 $(\sigma/\rho - 1)C_l$ が 0.25~0.55 程度の範囲で線形に変化するよう近似した無次元掃流砂量

$$q_*^{(s)} = \frac{2}{3} \left(\frac{h}{d_L}\right)^2 (C_\infty^{(d)})^2 \left\{ \frac{0.6 + (\frac{\sigma}{\rho} - 1)C_l}{[\frac{\sigma}{\rho} \lambda^2 a_i \sin \alpha + (\frac{K}{\lambda})]^{1/2} [(\frac{\sigma - \rho}{\rho})C_l + 1]C_l} \right\} \left(1 - \frac{\tau_{*c}}{\tau_*}\right)^2 \quad (\text{B.35})$$

を得た(高橋, 1982, 式 (20))。ここで以後の表記の簡略化のため、波括弧の部分を

$$A = \left\{ \frac{0.6 + (s - 1)C_l}{[s \lambda^2 a_i \sin \alpha + (\frac{K}{\lambda})]^{1/2} [(s - 1)C_l + 1]C_l} \right\} \quad (\text{B.36})$$

とおく。ここに、比重 $s = \sigma/\rho$ 、 λ は線濃度、 $a_i \sin \alpha$ は砂礫間の衝突に関する係数、 K は砂礫流動層内の空隙における流体の乱れの混合距離 $\ell = K d_L/\lambda$ の係数である。

式 B.35 および式 B.36 を式 B.34 に代入すると、

$$\hat{C}_\infty^{(s)} = \frac{2}{3} \frac{A}{R} (C_\infty^{(d)})^2 \left(1 - \frac{\tau_{*c}}{\tau_*}\right)^2 \quad (\text{B.37})$$

を得る。さらに、無次元掃流力の定義

$$\tau_* = \frac{u_*^2}{(s - 1)gd_L} = \frac{gh \sin \theta}{(s - 1)gd_L} = \frac{h \sin \theta}{d_L s - 1} \quad (\text{B.38})$$

19) 素直に計算すると右辺は $(1 - \tau_c/(pgh \sin \theta))$ となる。 $\tau = pgh \sin \theta$ と、無次元掃流力 $\tau_* = \tau/((\sigma - \rho)gh)$ 、無次元限界掃流力 $\tau_{*c} = \tau_c/((\sigma - \rho)gh)$ の関係を用いると、この形式で表現できる。

20) 土石流の平衡土砂濃度は $C_\infty^{(d)} = (\rho \tan \theta)/[(\sigma - \rho)(\tan \phi - \tan \theta)] = (s - 1)^{-1} \tan \theta/(\tan \phi - \tan \theta)$ である。

21) この仮定は、実験結果を根拠とする。集合流動としての土石流と掃流状集合の下層が同じ流動形式であることから類推したものと思われる。掃流状集合流動は二層流(下方に密度が大きいものが存在する密度流)であるが、流速抵抗は全層の平均流速で議論する量であるため、均質化されていることに注意。また、高橋 (1982) では、砂礫流動層と水流層における流速分布と平均流速の流速係数をそれぞれ理論的に導き、各相の平均流速(高橋, 1982, 式 17.22)と全流動層の平均流速式との関係式(高橋, 1982, 式 23)と式 B.32 から、全流動の平均流速による流速係数が相対水深が概ね比例することを実験的に確かめ、抵抗係数を式 B.44 を導いた。

を式 B.37 に代入すると、

$$\hat{C}_\infty^{(s)} = \frac{2}{3} \frac{A}{R} (C_\infty^{(d)})^2 \left(1 - \frac{\tau_{*c}(s-1)}{\sin \theta} \left(\frac{h}{d_L} \right)^{-1} \right)^2 \quad (\text{B.39})$$

を得る。この式では右辺の括弧内の第 2 項が相対水深に反比例することが特徴である。

以上は、掃流状集合流動の抵抗則が相対水深に比例するように近似したこと、実験による無次元掃流砂量を用いたことを除くと、設定した問題から導かれる比較的一般的な式である。ここからは物性値を代入した実験式を示す。

高橋 (1982) によれば、式 B.36 を C_l の関数と考えると、 C_l が 0.2 から 0.33 の範囲において、曲線 A に回帰直線を引くと、

$$A = 4.2 - 0.75C_l \quad (\text{B.40})$$

と近似できる²²⁾。ここに、物性値として $s = 2.61$, $a_i \sin \alpha = 0.02$, $K = 3$, $\kappa = 0.7$, $\lambda = 2.954$ を用いた。この近似式を式 B.39 に代入すると、

$$\hat{C}_\infty^{(s)} = \frac{2}{3} \frac{(4.2 - 0.75C_l)}{R} (C_\infty^{(d)})^2 \left(1 - \frac{\tau_{*c}(s-1)}{\sin \theta} \left(\frac{h}{d_L} \right)^{-1} \right)^2 \quad (\text{B.41})$$

を得る (高橋, 1982, 式 (21))。

Takahashi (1991) に従って、掃流状集合流動の抵抗係数 $R = 0.4$, $\tau_{*c} = 0$, $C_l = (C_c^{(d)} =) 0.4C_*$ (正確には $\frac{\kappa}{\kappa+1}C_*$, $\kappa = 0.7$), $C_* = 0.65$ を式 B.41 に代入すると

$$\hat{C}_\infty^{(s)} = 6.7(C_\infty^{(d)})^2 \left(1 - \frac{\tau_{*c}(s-1)}{\sin \theta} \left(\frac{h}{d_L} \right)^{-1} \right)^2 \quad (\text{B.42})$$

となる。式 B.32 で説明したように限界掃流力 $\tau_c = 0$ (Takahashi, 1991, p.42), あるいは、実験結果によれば相対水深の変化が平衡土砂濃度に与える影響は小さいため $h/d_L = \infty$ (高橋, 1982, Fig.8) とできる。このどちらからの関係を用いると、

$$\hat{C}_\infty^{(s)} = 6.7(C_\infty^{(d)})^2 \quad (\text{B.43})$$

となる (Takahashi, 1991)。これが第 3 章で用いた式 3.13 である。この実験式は図 3.9 の右図に示すように実験を説明する。この式を導く際に、掃流状集合流動の砂礫移動層における (輸送) 土砂濃度は流動深によらず、 $C_l = C_c^{(d)}$ で一定とした。なお、 $\tau_c = 0$ の時は $h_l \approx h$ となり砂礫流動層が全層となる土石流となることを表す。一方、 $h/d_L = \infty$ は砂礫移動層がきわめて薄い掃流砂流であることを表す。この 2 つの極端な条件のいずれであっても式 B.43 は実験結果 (高橋, 1982) を説明することから、掃流状集合流動の平衡土砂濃度は相対水深等の関数でなく勾配の関数として近似できると考えることができる。これは、水山 (1980) による実験式 (式 B.29) を支持する結果である²³⁾。

B.4.4 平衡土砂濃度の他表現

Takahashi (1991) が $\hat{C}_\infty^{(s)} = 6.7(C_\infty^{(d)})^2$ (第 3 章式 3.13) を示す前には、掃流状集合流動の平衡濃度式は色々な表現で論文に掲載されていた。ここでは、それらの表現を整理する。

高橋 (1982) は、式 B.41 に、 $C_l (= C_c^{(d)}) = 0.4C_*$ (正確には、 $C_*\kappa/(\kappa+1)$, $\kappa = 0.7$), $C_* = 0.65$, $\tau_{*c} = 0.05$, 掃流状集合流動の全層の平均流速に対する流速係数と相対水深の比例係数である抵抗係数 (式 A.65)

$$R = 0.23 + 0.65 \tan \theta \quad (\text{B.44})$$

を代入して

$$\hat{C}_l = \frac{2.66(C_\infty^{(d)})^2}{0.23 + 0.65 \tan \theta} \left(1 - \frac{0.05(s-1)}{\sin \theta} \left(\frac{h}{d} \right)^{-1} \right)^2 \quad (\text{B.45})$$

を示した²⁴⁾ (高橋, 1982, 式 26)。

22) 高橋 (1982) において、 $0.2 < C_c^{(d)} < 0.33$ の範囲で A の回帰直線としてこの近似式が示される。筆者が元の式 (式 B.36) から同じ範囲でこの近似式の導出を再現しようとしたところ、近似式が $4.23 - 7.41C_l$ となり、式 B.40 と切片は一致したが勾配が 10 倍程度異なった。同じパラメータと抵抗係数を使うと式 B.42, 式 B.43 の係数 6.7 (≈ 6.66) は 3.74, 河床せん断応力と同様に $R = 0.7$ とすると 2.14 となった。筆者の計算間違いかもしれないが一言付けておく。

23) 水山 (1980) の実験式 $5.5(\tan \theta)^2$ は勾配で決まり相対水深に依存しない。高橋 (1982) は、掃流状集合流動の一連の議論から導かれた掃流状集合流動の平衡土砂濃度式 (式 B.41; 正確には式 B.45) について、物性値を一定にして相対水深を変化させても平衡土砂濃度の変化は小さいかったため、相対水深によらずに一本の平衡土砂濃度式で近似することも妥当であると考えた。このことから、高橋 (1982) は、本節で示したような一連の議論が水山の実験式式 B.29 に理論的な根拠を与えた、とした。

24) 高橋 (1982) では、この式は流砂濃度 (輸送濃度) の式として示される。平衡土砂濃度と言及しておらず、下添字 ∞ は記されていない。

高橋・藤井 (1988, 式 (20)) は, 土石流の平衡土砂濃度 $C_\infty^{(d)}$ を式 B.37 に代入した,

$$\hat{C}_\infty^{(s)} = \frac{2}{3} \frac{A}{R} (s-1)^{-2} \left(\frac{\tan \theta}{\tan \phi - \tan \theta} \right)^2 \left(1 - \frac{\tau_{*c}}{\tau} \right)^2 \quad (\text{B.46})$$

に対して, $R = 0.8/(s-1)^{25}$ を代入した

$$\hat{C}_\infty^{(s)} = \frac{3.3}{s-1} \left(\frac{\tan \theta}{\tan \phi - \tan \theta} \right)^2 \left(1 - \frac{\tau_{*c}}{\tau} \right)^2 \quad (\text{B.47})$$

を示した²⁶⁾。

高橋と共同研究者の論文を調べると, 掃流状集合流動の平衡土砂濃度式については, 特に類似した様々な表現が見つかるが, 基本的に同じ考えから導かれた同一の式であるが, 適用する抵抗則や代入した物性値が異なることのために, 様々な表現が存在すること, またそれらの関連を示した。

なお, Takahashi (1991) 以降, 高橋土石流理論を適用する論文では物性値が固定された式 B.43 を適用する例が多い。

B.4.5 無次元流砂量式などの類似表現や関連式

以上の掃流状集合流動の導出で表れた無次元流砂量式と砂礫移動層厚の式その他表現を整理する。

B.4.5.1 無次元流砂量式

掃流状集合流動の無次元流砂量式 (式 B.35) のふたつの別表現との関係を整理しておく。

A (式 B.36) に式 B.40 を代入し, 平衡土砂濃度の定義

$$(C_\infty^{(d)})^2 = \left[\frac{\rho \tan \theta}{(\sigma - \rho)(\tan \phi - \tan \theta)} \right]^2 = \left(\frac{\rho}{\sigma - \rho} \right)^2 \left(\frac{\sin^2 \theta}{\cos^2 \theta (\tan \phi - \tan \theta)^2} \right) \quad (\text{B.48})$$

および無次元掃流力の定義 $\tau_* = \tau / [(\sigma - \rho)gd_L] = \rho gh \sin \theta / [(\sigma - \rho)gd_L]$ の二乗を变形した

$$\left(\frac{h}{d_L} \right)^2 = \left(\frac{\tau_*}{\sin \theta} \right)^2 \left(\frac{\sigma - \rho}{\rho} \right)^2 \quad (\text{B.49})$$

を, 式 B.35 に代入すると

$$q_*^{(s)} = \frac{2}{3} \frac{(4.2 - 0.75C_l)}{\cos^2 \theta (\tan \phi - \tan \theta)^2} \tau_*^2 \left(1 - \frac{\tau_{*c}}{\tau_*} \right) \quad (\text{B.50})$$

を得る (高橋, 1982, 式 (21))。

また, 掃流状集合流動の砂礫移動層の土砂濃度 C_l は土石流の下限濃度 $C_c^{(d)} = C_*/(1 + \kappa^{-1})$ と考えてよいとすれば, $\kappa = 0.7$, $C_* = 0.6$ とすると, $C_l = C_c^{(d)} \approx 0.25$ であるから, 式 B.40 は $4.2 - 0.75 \cdot 0.25 \approx 4.0$ である。B.4.1 項に示したように $\tau_{*c} = 0$ とし, 式 B.50 に代入すれば,

$$q_*^{(s)} = \frac{8}{3} \frac{\tau_*^2}{\cos^2 \theta (\tan \phi - \tan \theta)^2} \quad (\text{B.51})$$

を得る (高橋, 1992, 式 33)。

B.4.5.2 砂礫移動層の厚さ

式 B.32 において, B.4.1 項に示したように $\tau_{*c} = 0$ とすれば,

$$\frac{h_l}{h} = \frac{C_\infty^{(d)}}{C_l} \quad (\text{B.52})$$

を得る (Takahashi, 1991, 式 2.3.32)。前項に示したように $C_l = C_c^{(d)} = C_*/(1 + \kappa^{-1})$ とし, $\kappa = 0.7$ とし, 式 B.48 と共に代入すると

$$\frac{h_l}{h} = \frac{\tan \theta}{0.4C_*(\sigma/\rho - 1)(\tan \phi - \tan \theta)} \quad (\text{B.53})$$

を得る (土木学会水工学委員会, 1999, p.149, 式 2-3.34)。

25) 高橋・藤井 (1988) による式 A.66 に示される抵抗則から分かる抵抗係数 $R = 0.8\rho_m/(\sigma - \rho_m)$ において, $s = \sigma/\rho_m$ の関係を用いた実験式である。

26) 高橋・藤井 (1988, 式 (20)) では係数は 2.3 である。3.3 は筆者が計算した結果である。

引用文献

- Bagnold, Ralph Alger (1954): Experiments on a gravity-free dispersion of large solid spheres in a Newtonian fluid under shear, Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences, Vol. 225, No. 1160 pp.49–63.
- 岩垣雄一 (1956): 限界掃流力に関する基礎的研究:(I) 限界掃流力の流体力学的研究, 土木学会論文集, No. 41, pp.1–21.
- Bagnold, Ralph Alger (1956): The flow of cohesionless grains in fluids, Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences, Vol. 249, No. 964 pp.235–297.
- 伊藤剛・渡辺二郎・平本和子・日野幹雄 (1963): 高潮数値計算における二三の問題点について, 水理講演会, pp.5–10.
- Bagnold, Ralph Alger (1966): The shearing and dilatation of dry sand and the 'singling' mechanism, Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences, Vol. 295, No. 1442, pp.219–232.
- 谷勲 (1968): 土石流 (山津浪) について, 水利科学, Vol. 12, No. 1, pp.109–126.
- Leigh, D. C. (1968): *Nonlinear Continuum Mechanics*, McGraw-Hill, 240 p. (村上澄男訳, 非線形連続体力学, 共立出版, 1975) .
- 藤田宏・伊藤剛 (1971): 差分近似の数学的背景, 伊藤剛 (編) 数値解析の応用と基礎 (水理学を中心として), 第 10 章, pp.197–222, アテネ出版.
- 芦田和男・道上正規 (1972): 移動床流れの抵抗と掃流砂量に関する基礎的研究, 土木学会論文報告集, Vol. 1972, No. 206, pp.59–69.
- 奥田節夫・諏訪浩・横山康二・枝川尚資 (1972): 土石流の総合的計測システム, 写真測量, Vol. 11, No. 1, pp.13–20.
- 芦田和男・大同淳之・高橋保・水山高久 (1973): 急勾配流れの抵抗と限界掃流力に関する研究, 京都大学防災研究所年報, No. 16 B, pp.481–494.
- 芦田和男・高橋保・水山高久 (1973): 急勾配水路における混合砂礫の移動限界掃流力に関する研究, 新砂防, No. 103, pp.6–13.
- 奥田節夫・土石流研究グループ (1973): 土石流の現地調査について, 京都大学防災研究所年報, No. 16 A, pp.53–69.
- 大同淳之 (1973): 堆積物の集合流動の発生限界, 水理講演会講演集, Vol. 17, pp.85–90.
- 大同淳之 (1974): 急こう配上の多量の砂れきを含む流れの流動特性, 水理講演会講演集, Vol. 18, pp.199–203.
- 芦田和男・高橋保・水山高久 (1975): 流路工計画に関する水理学的研究, 砂防学会誌, Vol. 28, No. 2, pp.9–16.
- 奥田節夫・諏訪浩・仲野公章・横山康二 (1976): 土石流の総合的観測その 2. 1975 年焼岳東斜面上々堀沢, 京都大学防災研究所年報, No. 19 B-1, pp.385–402.
- 高橋保 (1977): 土石流の発生と流動に関する研究, 京都大学防災研究所年報, No. 20 B-2, pp.405–435.
- 水山高久 (1977): 山地河川の掃流砂に関する研究, 博士論文, 96 p.
- Takahashi, Tamotsu (1978): Mechanical characteristics of debris flow, Journal of the Hydraulics Division, Vol. 104, No. 8 pp.1153–1169.
- 芦田和男・高橋保・澤井健二 (1978): 土石流危険度の評価法に関する研究, 京都大学防災研究所年報, No. 21 B-2, pp.1–17.
- 芦田和男・高橋保・水山高久 (1978): 山地河川の掃流砂量に関する研究, 砂防学会誌, Vol. 30, No. 4, pp.9–17.
- 高橋保・浜田士郎・吉田等 (1978): 土石流の堆積過程に関する研究, 土木学会関西支部年次講演会概要集, pp.II-60-1-II-60-2.
- Varnes, David J (1978): Slope movement types and processes, Special report, Vol. 176 pp.11–33.
- 金谷健一 (1979): 粒状体の流動の基礎理論 第 2 報 発達した流れ, 日本機械学会論文集 B 編, Vol. 45, No.

- 高橋保・吉田等 (1979) : 土石流の停止・堆積機構に関する研究 (1) -勾配の急変による堆積-, 京都大学防災研究所年報, No. 22 B-2, pp.315-328.
- 岩佐義朗・井上和也・水鳥雅文 (1980) : 氾濫水の水利の数値解析法, 京都大学防災研究所年報, pp.305-317.
- 高橋保 (1980) : 土石流の力学的機構, 水工学シリーズ, Vol. 80, No. A-1, pp.1-17.
- 水山高久 (1980) : 土石流から掃流に変化する勾配での流砂量, 砂防学会誌, Vol. 33, No. 1, pp.1-6.
- 池谷浩・水山高久 (1982) : 土石流の流動と堆積に関する研究, 土木研究所報告, pp.88-153.
- 高橋保 (1982) : 土石流の停止・堆積機構に関する研究 (3) -土石流扇状地の二次侵食-, 京都大学防災研究所年報, No. 25 B-2, pp.327-348.
- 樫東一郎・橋本晴行・末次忠司 (1982) : 土石流における粒子間応力と流動特性, 土木学会論文報告集, Vol. 317, pp.79-91.
- ベッカー, E.・ビュルガー, W. (1983) : 連続体力学: 熱力学的基礎と応用, 浜田実・田中喜久昭・田中正隆 訳, 森北出版, 260 p.
- 高橋保 (1983) : 土石流, 河川の土砂災害と対策, 第 3 章, pp.55-149, 森北出版.
- 高橋保・八木秀樹 (1983) : 土石流の流量予測, 京都大学防災研究所年報, No. 26 B-2, pp.329-351.
- 日野幹雄 (1983) : 明解水理学, 丸善, 345 p.
- Savage, S.B. and Sayed, M. (1984): Stresses developed by dry cohesionless granular materials sheared in an annular shear cell, *Journal of Fluid Mechanics*, Vol. 142 pp.391-430.
- 高橋保 (1984) : 土石流の流れ学, なかれ, No. 3, pp.307-317.
- 高橋保・辻本浩史 (1984) : 土石流氾濫・堆積の数値シミュレーション, 京都大学防災研究所年報, No. 27 B-2, pp.467-485.
- 芦田和男 (1985) : 土砂生産と流出, 流砂の水理学, 第 11 章, pp.345-362, 丸善.
- 芦田和男・江頭進治・神矢弘・佐々木浩 (1985) : 斜面における土塊の抵抗則と移動速度, 京都大学防災研究所年報, No. 28 B-2, pp.297-307.
- 吉川秀夫 (1985) : 流砂の水理学, 丸善, 543 p.
- パタンカースハス V. (1985) : コンピュータによる熱移動と流れの数値解析, 水谷幸夫・香月正司訳, 森北出版, 198 p.
- 水山高久・下東久巴 (1985) : 土石流氾濫シミュレーション, 土木技術資料, Vol. 27, No. 8, pp.415-421.
- 宮本邦明 (1985) : Newton 流体を含む粒子流の流動機構に関する研究, 博士論文, 155 p.
- 芦田和男編 (1985) : 扇状地の土砂災害, 古今書院, 224 p.
- 高橋保・匡尚富 (1986) : 変勾配流路における土石流の形成, 京都大学防災研究所年報, No. 29 B-2, pp.343-359.
- 高橋保・中川一・西崎丈能 (1986) : 堤防決壊による洪水危険度の評価に関する研究, 京都大学防災研究所年報, No. 29 B-2, pp.431-450.
- 中川博次・辻本哲郎 (1986) : 移動床流れの水理学, 技報堂出版, 311 p.
- 橋本晴行・樫東一郎・平野宗夫 (1986) : 土砂の掃流状集合流動について, 水理講演会論文集, Vol. 30, pp.235-240.
- ファインマン・レイトン・サンズ (1986) : ファインマン物理学 3 : 電磁気学, 宮島龍興訳, 岩波書店, 330 p.
- 矢澤昭夫・水山高久・北原一平 (1986) : 土石流氾濫シミュレーションモデルによる土石流対策工の効果評価に関する研究, 土木研究所資料 第 2392 号, 93 p.
- Takahashi, Tamotsu (1987): High velocity flow in steep erodible channels, *Proceeding of 22th IAHR Congress*, Lausanne pp.42-53.
- Takahashi, T., Nakagawa, H., and Kuang, S. (1987): Estimation of debris flow hydrograph on varied slope bed, *Erosion and Sedimentation in the Pacific Rim (Proceedings of the Corvallis Symposium)* pp.167-177.
- 高橋保・中川一・山路昭彦 (1987) : 土石流氾濫危険範囲の指定法に関する研究, 京都大学防災研究所年報, No. 30 B-2, pp.611-626.
- 水山高久・阿部宗平・下東久巴 (1987) : 土石流氾濫シミュレーションと模型実験による土石流災害の再現, 砂防学会誌, Vol. 40, No. 3, pp.17-22.
- 芦田和男 (1988) : 土砂の生産・流出現象と災害, 土木学会論文集, No. 393/II-9, pp.21-32.
- 高橋保・中川一・佐藤広章 (1988a) : 扇状地における土砂氾濫災害危険度の評価, 京都大学防災研究所年報,

- No. 31 B-2, pp.655-676.
- 高橋保・中川一・佐藤広章 (1988b) : 土石流および土砂流扇状地の形成に関する研究, 第 32 回 水理講演会論文集, pp.497-502.
- 高橋保・藤井由之 (1988) : 流下域における土石流の制御法に関する研究, 京都大学防災研究所年報, No. 31 B-2, pp.633-654.
- 水山高久・北原一平・原義文 (1988) : 土石流氾濫シミュレーションモデルによる土石流対策工の効果評価, 砂防学会誌, Vol. 40, No. 5, pp.14-21.
- Takahashi, T. and Nakagawa, H. (1989): Debris Flow Hazard Zone Mapping, Proceedings of the Japan-China (Taipei) Joint Seminar on Natural Hazard Mitigation, Kyoto pp.363-372.
- 江頭進治・芦田和男・矢島啓・高濱淳一郎 (1989) : 土石流の構成則に関する研究, 京都大学防災研究所年報, No. 32 B-2, pp.1-15.
- 高橋保・匡尚富 (1989) : 合流部における土石流の挙動と変形, 京都大学防災研究所年報, No. 32 B-2, pp.709-732.
- 中川一 (1989) : 洪水および土砂氾濫災害の危険度評価に関する研究, 博士論文, 221 p.
- 高橋保・中川一・原田達夫 (1990) : 混合粒径材料からなる渓床侵食型土石流の予測, 京都大学防災研究所年報, Vol. 33 B-2, pp.443-456.
- 砂防学会 (1991) : 渓流の土砂移動現象, Vol. 4, 砂防学講座, 山海堂, 316 p.
- Takahashi, Tamotsu (1991): *Debris flow*, IAHR Monograph Series, A.A.Balkema, 165 p.
- 高橋保・中川一 (1991) : 豪雨時に発生する石礫型土石流の予測, 砂防学会誌, Vol. 44, No. 3, pp.12-19.
- 高橋保・中川一・山敷庸亮 (1991) : 混合粒径材料からなる土石流扇状地の形成とその侵食, 京都大学防災研究所年報, No. 34 B-2, pp.355-372.
- 日本流体力学会 (1991) : 混相流体の力学, 朝倉書店, 221 p.
- 橋本晴行・平野宗夫・林重徳・梅村順 (1991) : 1990 年 7 月熊本県一の宮町の土石流災害について, 水文・水資源学会誌, Vol. 4, No. 1, pp.25-32.
- 芦田和男 (1992) : 土石流概論, 砂防学会誌, Vol. 44, No. 6, pp.41-47.
- 高橋保・中川一 (1992) : 自然ダムの越流決壊によって形成される洪水の予測, 京都大学防災研究所年報, No. 35 B-2, pp.231-248.
- 高橋保 (1992) : 土石流・泥流の流動機構, 芦田和男先生退官記念事業会 (編) 土砂移動現象に関するシンポジウム論文集 芦田和男先生退官記念, pp.39-55, 学術出版印刷.
- 久田俊明 (1992) : 非線形有限要素法のためのテンソル解析の基礎, 丸善出版, 226 p.
- Gidaspow, Dimiri (1994): *Multiphase Flow and Fluidization*, Academic Press, 467 p.
- 芦田和男・江頭進治・佐々木幹夫・清水義彦・関根正人・玉井昌宏・藤田正治・宮本邦明・森明巨 (1994) : 固液混相流における支配方程式の構造について, 水工学論文集, Vol. 38, pp.563-570.
- 梶島岳夫 (1994): 対流項の差分形式とその保存性, 日本機械学会論文集 B 編, Vol. 60, No. 574, pp.2058-2063.
- 杉原厚吉 (1994) : 計算幾何工学, 培風館, 230 p.
- Hutter, K., Svendsen, B., and Rickenmann, D. (1994): Debris flow modeling: A review, *Continuum Mechanics and Thermodynamics*, Vol. 8 pp.1-35.
- 平野宗夫・橋本晴行・河野通斉 (1994) : 1993 年雲仙における土石流の現地観測, 水工学論文集, Vol. 38, pp.747-752.
- 芦田和男・江頭進治・佐々木幹夫・清水義彦・関根正人・玉井昌宏・藤田正治・宮本邦明・森明巨 (1995) : 固液混相流における支配方程式とその適用性, 水工学論文集, Vol. 39, pp.537-550.
- 岩佐義朗 (1995) : 数値水理学におけるモデルの作成, 岩佐義朗 (編) 数値水理学, pp.17-35, 丸善.
- 岩佐義朗・松尾直規 (1995) : 数値水理学における問題点, 岩佐義朗 (編) 数値水理学, pp.67-82, 丸善.
- 高橋保・里深好文・千代啓三 (1996) : 土石流の慣性領域での流動則, 京都大学防災研究所年報, No. 39 B-2, pp.333-346.
- 中川一・高橋保・澤田豊明・里深好文 (1996) : 計画土石流の設定と避難計画, 京都大学防災研究所年報, No. 39 B-2, pp.347-371.
- 橋本晴行・平野宗夫 (1996) : 掃流状集合流動の抵抗則と流砂量, 土木学会論文集, Vol. 1996, No. 545, pp.33-42.
- 林泰造 (1996) : 基礎水理学, 鹿島出版会, 310 p.

- Iverson, Richard, M (1997): The physics of debris flows, *Reviews of Geophysics*, Vol. 35 pp.245–296.
- 江頭進治・宮本邦明・伊藤隆郭 (1997): 掃流砂量に関する力学的解釈, *水工学論文集*, Vol. 41, pp.789–794.
- 高橋保・辻本浩史 (1997): 斜面上の粒状体流れの流動機構, *土木学会論文集*, No. 565, pp.57–71.
- 宮本邦明 (1997): 河床変動現象と基礎方程式, *砂防学会誌*, Vol. 50, No. 2, pp.61–67.
- 伊藤隆郭・江頭進治 (1998): 土石流に関する構成則の比較研究, *水工学論文集*, Vol. 42, pp.913–918.
- 高濱淳一郎 (1998): 土石流による河床変動, *砂防学会誌*, Vol. 51, No. 3, pp.66–72.
- 中川一・高橋保・里深好文・立川康人・市川温・吉田義則・中村行雄 (1998): 平成9年度鹿児島県出水市針原川で発生した土石流災害について, *京都大学防災研究所年報*, No. 41 B-2, pp.287–298.
- 池田駿介 (1999): 詳述水理学, 技報堂出版, 436 p.
- 池谷浩 (1999): 土石流災害, 岩波新書, 岩波書店, 222 p.
- 高橋保・里深好文 (1999): 慣性土石流の統一モデル, *京都大学防災研究所年報*, No. 42 B-2, pp.201–209.
- 高橋保・中川一・里深好文・鈴木信昭 (1999): 豪雨時における水系一貫の土砂流出予測に関する研究, *河川技術に関する論文集*, Vol. 5, pp.177–182.
- 土木学会水工学委員会水理公式集編集小委員会 (1999): 水理公式集 平成11年版, 土木学会, 713 p.
- 川池健司・井上和也・戸田圭一・中川一・中井勉 (2000): 急傾斜都市における土砂・氾濫災害の解析, *京都大学防災研究所年報*, No. 43 B-2, pp.333–343.
- 砂防・地すべり技術センター砂防技術研究所 (2000): 数値シミュレーション比較検討, *研究年報1999-1*, (財)砂防・地すべり技術センター砂防技術研究所, 161 p.
- 高橋保・井上素行・中川一・里深好文 (2000): 山岳流域における土砂流出の予測, *水工学論文集*, Vol. 44, pp.717–722.
- 高濱淳一郎・藤田裕一郎・近藤康弘 (2000a): 土石流から掃流状集合流動に遷移する流れの解析法に関する研究, *水工学論文集*, Vol. 44, pp.683–686.
- 高濱淳一郎・藤田裕一郎・近藤康弘 (2000b): 土石流シミュレーションモデルの比較研究, *水工学論文集*, Vol. 44, pp.687–692.
- 伊藤隆郭 (2001): 土石流の構成則およびその適用に関する研究, 博士論文, 112 p.
- 中川一・高橋保・里深好文・川池健司 (2001): 1999年ベネズエラのカムリグランデ流域で発生した土砂災害について-数値シミュレーションによる再現計算と砂防施設配置効果の評価, *京都大学防災研究所年報*, No. 44 B-2, pp.207–227.
- 水山高久・江頭進治 (2001): 用語としての「土石流」と「土砂流, 掃流状集合流動」, *砂防学会誌*, Vol. 54, No. 3, p.90.
- 伊藤隆郭・宮本邦明 (2002): 土石流の1次元解析法に関する研究, *水工学論文集*, Vol. 46, pp.671–676.
- 水理委員会水理公式集例題プログラム集編集部会 (2002): 水理公式集 例題プログラム集, 土木学会.
- 高橋保・里深好文 (2002): 石礫型及び乱流型土石流の一般理論とその実用化モデル, *砂防学会誌*, Vol. 55, No. 3, pp.33–42.
- 高濱淳一郎・藤田裕一郎・近藤康弘・蜂谷圭 (2002): 土石流の堆積侵食過程に関する実験と二層流モデルによる解析, *水工学論文集*, Vol. 46, pp.677–682.
- 中川一・高橋保・里深好文・川池健司 (2002): 砂防施設の効果の評価に関する研究, *水工学論文集*, Vol. 46, pp.665–670.
- 今井功 (2003): 古典物理の数理, 岩波書店, 168 p.
- 江頭進治・宮本邦明 (2003): 土石流のメカニズム, *ジオテクノート12 土石流*, 第4章, pp.41–77, 社団法人地盤工学会.
- 鈴木拓郎・堀田紀文・宮本邦明 (2003): 土石流の流れに及ぼす河床粗度の影響についての実験的検討, *砂防学会誌*, Vol. 56, No. 2, pp.5–13.
- 水山高久 (2003): 概説, *ジオテクノート12 土石流*, 第1章, pp.1–14, 社団法人地盤工学会.
- 宮本邦明・山下伸太郎 (2003): 土石流のシミュレーション, *ジオテクノート12 土石流*, 第5章, pp.79–109, 社団法人地盤工学会.
- 伊藤隆郭・江頭進治 (2004): 土石流の流出土砂量に及ぼす形状係数の重要性, *水工学論文集*, Vol. 48, pp.895–900.
- 江頭進治・伊藤隆郭 (2004): 土石流の数値シミュレーション, *日本流体力学会数値流体力学部門 Web 会誌*, Vol. 12, No. 2, pp.33–42.

- 砂防学会編 (2004) : 改訂 砂防用語集, 山海堂, 432 p.
- 高橋保 (2004) : 土石流の機構と対策, 近未来社, 432 p.
- 細田尚 (2004) : 河川流と河床・河道変動の水理解析法, 物性研究, Vol. 83, No. 1, pp.1-76.
- Imaizumi, Fumitoshi, Tsuchiya, Satoshi, and Ohsaka, Okihiro (2005): Behaviour of debris flows located in a mountainous torrent on the Ohya landslide, Japan, Canadian Geotechnical Journal, Vol. 42, No. 3 pp.919-931.
- 金谷健一 (2005) : これならわかる最適化数学, 共立出版, 249 p.
- 関根正人 (2005) : 移動床流れの水理学, 共立出版, 211 p.
- 牛島省 (2006) : OpenMP による並列プログラミングと数値計算法, 丸善, 149 p.
- 高橋保 (2006) : 土砂流出現象と土砂害対策, 近未来社, 420 p.
- 池田暁彦・水山高久・原口勝則 (2007) : 土石流の発生を支配する降雨量に関する考察, 砂防学会誌, Vol. 60, No. 3, pp.26-31.
- 牛島省 (2007) : 数値計算のための Fortran90/95 プログラミング入門, 森北出版, 244 p.
- Hotta, Norifumi and Miyamoto, Kuniaki (2008): Phase classification of laboratory debris flows over a rigid bed based on the relative flow depth and friction coefficients, International Journal of Erosion Control Engineering, Vol. 1, No. 2 pp.54-61.
- 芦田和男・江頭進治・中川一 (2008) : 21 世紀の河川学, 京都大学学術出版会, 265 p.
- 西本晴男 (2008) : 「土石流」のはなし, 全国治水砂防協会, 246 p.
- Takahashi, Tamotsu (2009): A review of Japanese debris flow research, International Journal of Erosion Control Engineering, Vol. 2, No. 1 pp.1-14.
- Osanaï, Nobutomo, Shimizu, Takeshi, Kuramoto, Kazumasa, Kojima, Shinichi, and Noro, Tomoyuki (2010): Japanese early-warning for debris flows and slope failures using rainfall indices with Radial Basis Function Network, Landslides, Vol. 7, No. 3 pp.325-338.
- 田村圭司・内田太郎・山越隆雄・清水武志・吉野弘祐・松岡暁 (2010) : 天然ダムや火山噴火に起因した土石流の被害推定手法について, 砂防学会研究発表会概要集, Vol. 59, No. P-230.
- 新井宗之・水山高久 (2011) : 高濃度泥流サージの生成機構と流動モデルに関する研究, 土木学会論文集 B1 (水工学), Vol. 67, No. 3, pp.92-100.
- 内田太郎・山越隆雄・清水武志・吉野弘祐・木佐洋志・石塚忠範 (2011) : 河道閉塞 (天然ダム) 及び火山の噴火を原因とする土石流による被害範囲を速やかに推定する手法, 土木技術資料, Vol. 53, No. 7, pp.18-23.
- Metcalf, Michael, Reid, John, and Cohen, Malcolm (2011): *Modern Fortran explained*, Oxford University Press, 488 p.
- 清水武志・内田太郎・山越隆雄・石塚忠範 (2012a) : 緊急調査初動期の区域推定計算プログラム QUAD1.0 の設計と開発, 砂防学会研究発表会概要集, Vol. 61, No. Pa-65, pp.494-495.
- 清水武志・内田太郎・山越隆雄・石塚忠範 (2012b) : 天然ダムによる土石流想定範囲計算システム (QUAD-L) の開発と 2011 年台風 12 号災害における適用, 土木技術資料, Vol. 54, No. 10, pp.14-17.
- Uchida, Taro, Nishiguchi, Yuki, Nakatani, Kana, Satofuka, Yoshifumi, Yamakoshi, Takao, Okamoto, Atsushi, and Mizuyama, Takahisa (2013): New Numerical Simulation Procedure for Large-scale Debris Flows (Kanako-LS), International Journal of Erosion Control Engineering, Vol. 6, No. 2 pp.58-67.
- 鈴木拓郎 (2013) : 土石流の数値シミュレーションにおける侵食速度式の制御法, 砂防学会誌, Vol. 65, No. 5, pp.21-28.
- 鈴木拓郎・内田太郎・岡本敦 (2013) : 土砂移動シミュレーション手法における輸送形態の遷移に伴う不連続性の解消に関する研究, 砂防学会誌, Vol. 66, No. 2, pp.21-30.
- Yamakoshi, Takao, Kisa, Hiroshi, Shimizu, Takeshi, Ishizuka, Tadanori, Maeda, Akihiro, and Shigetaka, Takiguchi (2013): Quick analysis method for estimating debris flow prone area caused by rainfall in the volcanically disturbed watersheds, IAVCEI 2013 Scientific Assembly, Kagoshima, Japan, p.1155.
- 安部公輔・東森信就・久保雅義・藤原宏志・磯祐介 (2014) : Courant-Friedrichs-Lewy 条件についての注意, 日本応用数理学会論文誌, Vol. 24, No. 1, pp.15-26.
- 江頭進治・鈴木拓郎・内田太郎 (2014) : (質疑応答) 土砂移動シミュレーション手法における輸送形態の遷移に伴う不連続性の解消に関する研究, 砂防学会誌, Vol. 66, No. 5, pp.73-75.
- 梶島岳夫 (2014) : 乱流の数値シミュレーション 改訂版, 養賢堂, 285 p.

- Takahashi, Tamotsu (2014): *Debris flow: Mechanics, Prediction and Countermeasures, 2nd ed.*, CRC press, 551 p.
- 西口幸希 (2015): 深層崩壊に起因する土石流の数値シミュレーション手法に関する研究, 博士論文, 115 p.
- 国土技術政策総合研究所土砂災害研究部砂防研究室 (2016): 砂防基本計画策定指針 (土石流・流木対策編) 解説, 国土技術政策総合研究所資料 第 904 号, 国土交通省国土技術政策総合研究所, 77 p.
- 全国治水砂防協会 (2016): 砂防関係法令例規集 平成 28 年度版, 一般社団法人 全国治水砂防協会, 1727 p.
- 水野秀明・関根敬・江藤稚佳子・倉本和正 (2016): 静水状態を考慮できる有限体積法による段波の再現, 日本シミュレーション学会論文誌, Vol. 8, No. 1, pp.13–20.
- 和田孝志 (2016): 実用的な土石流数値計算モデルの開発, 博士論文, 195 p.
- 木村匡臣・田中智大・安瀬地一作・中谷加奈・山崎大・吉岡秀和 (2017): 地表水流れの数値解析技術に関する分野横断的視点から見た特徴と最前線, 水文・水資源学会誌, Vol. 30, No. 5, pp.307–334.
- 日本地形学連合 (編) (2017): 地形の辞典, 朝倉書店, 1032 p.
- 太田一行・新井涼允・豊田康嗣・佐藤隆宏 (2018): 連続性と時間刻み依存性を考慮した Diffusion wave 法による 2 次元土石流解析, 砂防学会誌, Vol. 71, No. 3, pp.21–28.
- 高橋佑弥・藤村直樹・水野正樹 (2018): 緊急時における災害想定のための高効率な氾濫計算プログラムの開発, 平成 30 年度砂防学会研究発表会概要集, Vol. 67, No. P-128, pp.625–626.
- 竹林洋史・藤田正治 (2018): 2016 年 4 月熊本地震時に発生した山王谷川の土石流の流動特性, 土木学会論文集 B1 (水工学), Vol. 74, No. 5, pp.1093–1098.
- 中谷加奈・古谷智彦・長谷川祐治・小杉賢一朗・里深好文 (2018): 細粒土砂の液相化が土石流挙動に及ぼす影響とその要因に関する検討, 砂防学会誌, Vol. 70, No. 6, pp.3–11.
- 山崎祐介・江頭進治 (2018): 土石流の規模に及ぼす崩壊個数と供給水量の影響, 土木学会論文集 B1 (水工学), Vol. 74, No. 5, pp.883–888.
- 太田一也 (2019): 雲仙普賢岳噴火回想録, 長崎文献社, 434 p.
- Sakai, Yuichi and Hotta, Norifumi (2019): Numerical simulation of debris flows focusing on the behavior of fine sediment, 7th International Conference on Debris-Flow Hazard Mitigation pp.421–428.
- 土木学会水工学委員会水理公式集編集小委員会 (2019): 水理公式集 2018 年版, 土木学会, 927 p.
- Miyamoto, Kuniaki, Itoh, Takahiro, and Kisa, Hiroshi (2019): Compressibility of solid phase of debris flow and erosion rate, 7th International Conference on Debris-Flow Hazard Mitigation pp.46–53.
- 清水武志・鈴木英夫・泉山寛明・河端智樹・藤村直樹 (2020): アーカイブされたアナログ空中写真への SfM-MVS の適用, 砂防学会誌, Vol. 72, No. 6, pp.3–7.
- 奈佐原顕郎 (2020): ライブ講義 大学生のための応用数学入門, 講談社, 240 p.
- 松島亘志・成瀬元・横川美和・東良慶・今泉文寿・佐々真志・田島芳満・知花武佳 (2020): 土砂動態, 共立出版, 274 p.
- 水山高久 (2020): コアストーン (corestone), 砂防学会誌, Vol. 72, No. 5, p.86.
- 日本混相流学会 (2021): 混相流ハンドブック (新装版), 朝倉書店, 502 p.
- 後藤仁志 (2022): 流れの方程式, 森北出版, 568 p.

第2部

ソースコード

目次

1	arrayc.f90	II-1
2	main.f90	II-5
3	flopen.f90	II-10
4	inputdem.f90	II-12
5	inputhydr.f90	II-18
6	inputparam.f90	II-20
7	clear.f90	II-24
8	calset.f90	II-27
9	wetdry.f90	II-30
10	qmflux.f90	II-34
11	qnflux.f90	II-39
12	boundaryq.f90	II-44
13	hdepth.f90	II-46
14	boundaryh.f90	II-52
15	qbflux.f90	II-53
16	boundaryqb.f90	II-55
17	gradws.f90	II-57
18	gradbed.f90	II-59
19	cinfty.f90	II-61
20	calclcf.f90	II-63
21	calrho.f90	II-66
22	calbedelev.f90	II-67
23	calaii.f90	II-71
24	velcal.f90	II-73
25	flowpower.f90	II-76
26	pickcal.f90	II-77

27	tractive.f90	II-82
28	replace.f90	II-84
29	snapshot.f90	II-86
30	outmaxvalue.f90	II-88
31	tau.f90	II-90
32	errnum.f90	II-96
33	Supplementary Tools	II-97
33.1	makeConnectiv.f90	II-97
33.2	projKawase2011.py	II-110
33.3	getindex.py	II-123
33.4	mkvtk.py	II-125
33.5	Makefile	II-131

1 arrayc.f90

arrayc.f90

```
1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 module arrayc
14     implicit none
15     character(len = 5) mhy(33)
16     character mw*5, mj*10, mk*40
17     character(len = 10) mw_10
18     character(len = 24) jtime0, jtime1, jtime2
19
20     integer nt, mt
21     parameter (nt = 777, mt = 33)
22
23     integer :: ihyf(mt), ibx(mt), jby(mt)
24     integer :: ibg(mt), ibd(mt)
25     integer :: ixbg(mt), ixbd(mt), ixhyf(mt)
26     integer :: iybg(mt), iybd(mt), iyhyf(mt)
27     double precision :: tdlt(nt), tdt( nt), rain( nt)
28     double precision :: qin( mt, nt), qmin( mt, nt), qnin( mt, nt)
29     double precision :: qblxin( mt, nt), qblyin( mt, nt), qbfxin( mt, nt), qbfyin( mt
30     , nt)
31     double precision :: hin( mt, nt), uin( mt, nt), vin( mt, nt)
32     double precision :: cli( mt, nt), cfi( mt, nt)
33     double precision :: sqin( mt), clb( nt), cfb( nt)
34     double precision :: vwi( nt), vli( nt), vfi( nt)
35     double precision :: vwt, vlt, vft, vtt, vflow, voutflow
36     double precision :: twi, tli, tfi, vmn, frh
37     double precision :: bit, bit_d, th, eps, dm, tanp, talp, sig, rho, tauc
38     double precision :: cst, cstl, cstf, cdl, csc, cdc, c90
39     double precision :: scr, wsg, gsr, tcw, cfl, wmax
40     double precision :: sodo, aisa
41     double precision :: bm ! 1.25d0 or 1.0d0
```

```

42  integer :: itr(999), jtr(999)
43  integer :: ncc, ncp
44  integer :: ipr, jpr, idz
45  integer :: im, jm, mlf, mpat, nbc, nxbc, nybc, mstep
46  integer :: Ngrid, Nxs, Nys
47  integer :: iox = 0
48  double precision :: dx, dy, dlt, dt1, dt2, timex
49  double precision :: tvl, tvf
50  double precision :: clmax, cfmax, hclmax, hcfmax, aclmax, acfmax
51  double precision :: czlmax, czfmax
52
53  integer, allocatable :: igc(:, :), iec(:, :)
54  integer, allocatable :: ixsrc(:, :), ixsrc(:, :), ixsgc(:, :)
55  integer, allocatable :: iysrc(:, :), iysrc(:, :), iysgc(:, :)
56  double precision, allocatable :: lonlat(:, :), xy(:, :)
57  double precision, allocatable, dimension(:) :: qm2, qm0, qn2, qn0, &
58      &      qm1, qn1, &
59      &      qmx, qny, qqa, &
60      &      qxmax, qymax, qqmax, tqmax, &
61      &      uul, vvl, taus, &
62      &      uva, uux, vvy, &
63      &      uvmax, tvmax, vxm, vym, &
64      &      vlo, vfo, &
65      &      qblx, qbfy, qbly, qbfy, &
66      &      qblxm, qblym, qbla, qbfxm, qbfym, qbfa, &
67      &      rof, rot, ct3, ct1, &
68      &      cl3, cl1, cf3, cf1, &
69      &      hn3, hn1, hn0, arvt, &
70      &      hmax, thmax, &
71      &      hz, hzmax, thzmax, &
72      &      aii, smn, sei, tapc, &
73      &      zl3, zl1, ezi, zdp, &
74      &      zvl, czl, zvf, czf, &
75      &      zdz, hahz, zahz, &
76      &      zmax, tzmax, &
77      &      zmin, tzmin, &
78      &      clp, &
79      &      fta, ftx, fty, &
80      &      fmax, tfmax, fmx, fmy, &
81      &      hafmax, zafmax
82
83  !— Wet/Dry management
84  integer, parameter :: isDry = 0
85  integer, parameter :: isWet = 1
86  integer :: Nwg, Nwxs, Nwys
87      ! Number of wet grids and edges
88  integer :: Nwg0, Nwxs0, Nwys0

```

```

89      ! Number of wet grids and edges (previous time step)
90  integer, allocatable :: idxWg(:), idxWxs(:), idxWys(:)
91      ! Indices of wet grids and edges
92  integer, allocatable :: WDg(:), WDxs(:), WDys(:)
93      ! Wet/Dry flags on each grid and edge
94
95  !— Outflow boundary conditions
96  integer :: nxout, nyout
97  integer, allocatable :: ixout(:), ixgout(:), iyout(:), iygout(:)
98
99  !— Wall boundary conditions
100 integer :: nxwall, nywall
101 integer, allocatable :: ixwall(:), iywall(:)
102
103 !— Sediment transportation flag
104 integer, allocatable, dimension(:, :) :: meshflag
105
106 ! Direction enumeration
107 integer, parameter :: iEE = 1
108 integer, parameter :: iNN = 2
109 integer, parameter :: iWW = 3
110 integer, parameter :: iSS = 4
111 integer, parameter :: iNE = 1
112 integer, parameter :: iNW = 2
113 integer, parameter :: iSW = 3
114 integer, parameter :: iSE = 4
115 integer, parameter :: iEg = 1
116 integer, parameter :: iNg = 1
117 integer, parameter :: iWg = 2
118 integer, parameter :: iSg = 2
119
120 ! global parameters
121 double precision, parameter :: pi = acos(-1.d0)
122      ! just in case
123 double precision, parameter :: gr = 9.80665d0
124 double precision, parameter :: r13 = 1.d0 / 3.d0
125      ! should use CBRT(a) if MKL is available
126 double precision, parameter :: r43 = 4.d0 / 3.d0
127      ! should use CBRT(a**4) if MKL is available
128 logical :: noSediment = .false.
129
130 double precision, parameter :: MAX_Cf_IN_FLUID = 0.26
131      ! 流れの細粒分土砂濃度の制御に使用 calclcf
132      ! 0.26 (26%)は閉じ込められて堆積する最大細粒分濃度高橋(2006,p.180)
133      ! 第1部 p.32 脚注35 参照
134      ! → cstf (c_{*df}) で制御するため未使用
135 double precision, parameter :: INIT_SOIL_DEPTH = 100.0

```

```
136      ! 計算領域一律に最大堆積可能深を100.0mに設定
137      !   -> 実質的に使用しない
138      double precision, parameter :: MAX_ELEVCHANGE_IN_1STEP = 0.1
139      ! 侵食・堆積速度式の制御において、1時間ステップの最大変化量
140      ! これが必要かは不明。最大堆積深を最大削剥量とすれば良いのかもしれない
141      integer :: flg_rebed
142      ! subroutine
143      calclcfで、過剰粗粒土砂を調整した場合に河床標高に反映する場合1とする。
144      ! File I/O
145      character(256) :: fsnap
146      integer :: iout
147      integer, parameter :: udbg = 50
148
149 end module arrayc
```

2 main.f90

main.f90

```
1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 program debris
14     !$ use omp_lib
15     use arrayc
16     implicit none
17
18     integer :: ntj, m, n, it, iteration !, uof
19     ! Computational time
20     call fdate(jtime1)
21     write(*, *)
22     write(*, *) "Computation_start:", trim(adjustl(jtime1))
23     write(*, *)
24
25     !$omp parallel
26     !$ write(*, '( "Hello, world. I am cpu", i5, ".") ') omp_get_thread_num()
27     !$omp end parallel
28
29     print *, 'INPUT_DATA'
30     ! open file stream
31     call flopen
32
33     ! input DEM
34     call inputdem
35
36     ! input hydrgraph
37     call inputhydr
38
39     ! input paramters
40     call inputparam
41
42
```

```

43  print *, 'END_INPUT_SECTION'
44
45  print *, 'INITIALIZATION'
46  call clear
47  call calset  ! Initial and boundary conditions
48
49  print *, 'input_total_volume:', twi
50  print *, 'input_total_coarse_sed.vol:', tli
51  print *, 'input_total_fine_sed.vol:', tfi
52  if (noSediment .eqv. .true.) then
53      print *, 'sediment_flux_calculation?-->No'
54  else
55      print *, 'sediment_flux_calculation?-->Yes'
56  end if
57
58  vmn = 0.d0
59  vwt = 0.d0
60  vlt = 0.d0
61  vft = 0.d0
62  vtt = 0.d0
63
64  timex = 0.d0
65
66  ! cal_time = 0.d0
67
68  ncc = 0
69
70  print *, 'START_CALCULATION_(TIME_MARCHING)'
71  write(*, *) 'calc_end_by', mlf, 'plot.'
72
73  iteration = 0
74  it = 1
75
76  ! — Output initiel status
77  call snapshot(iteration)
78
79  do m = 1, mlf
80
81      dlt = tdlt(m)
82      dt1 = tdt(m)
83      dt2 = 2.d0 * dt1
84
85      ntj = int(dlt / dt2 + 0.1d0)
86
87      do n = 1, ntj
88
89          timex = timex + dt2

```



```

90
91      !— Computation of discharge flux
92      !—      by using momentum eq. of mixture flow
93      call qmflux()
94      call qnflux()
95      call boundaryq(m)
96
97      !— Computation of water depth
98      !—      by using continuity eq. of mixture flow
99      call hdepth
100     call boundaryh(m)
101
102     if (noSediment .eqv. .false.) then
103         ! if sediment flux is calculated, call following
104
105         !— Computation of sediment transportation
106         !—      by using continuity eq. of
107         !—      coarse & fine sediment flow
108         call qbflux
109         call boundaryqb(m)
110
111
112         !— Computation of gradient
113         ! コメントでどちらかを選択可能
114         ! call gradbed
115         call gradws
116
117         !— Computation of equilibrium concentration
118         call cinfty
119
120         !— Computation of sediment concentration
121         !—      by using continuity eq. of
122         call calclcf
123         call calrho
124
125         !— Computation of river bed deformation
126         !—      by using 3 river bed equations
127         call calbedelev
128
129         !— Computation of erosion/deposition rate
130         call calaii
131
132     end if
133
134     !— Check wet area and update wet indices
135     call wetdry(hn3)
136

```

```

137         !— Update advection velocity
138         call velcal(m)
139
140         !— Computation of flow power
141         call flowpower
142
143         !— Refresh maxvalue
144         call pickcal
145
146         !— Computation of tractive force
147         call tractive
148
149         !— Step change
150         call replace
151
152         iteration = iteration + 1
153
154     end do
155     ! — n-loop-end
156
157     if ( mod(m, 10) == 0 ) then
158         write(*, fmt = '((" ", i0, " "), $)' ) m
159     else
160         write(*, fmt = '(a1, $)' ) '.'
161     end if
162
163     !— Output snap shot
164     if(iout == 1) then
165         call snapshot(m)
166     end if
167
168 end do
169 ! — m-loop-end
170
171 ! — Output terminal status
172 call snapshot(-9999)
173
174 ! — Output max values
175 call outmaxvalue
176
177 call tau
178
179 call fdate(jtime2)
180
181 ! — Check overflow volume
182 if (voutflow > eps) then
183     write(*, *)

```

```
184         write(*, *)
185         write(*, *) 'Outflow_discharge_volume:␣', voutflow, '␣m3'
186         write(80, *) voutflow
187     end if
188
189     write(*, *)
190     write(*, *)
191     write(*, *)
192     write(*, *) 'End_flood_computation:␣', trim(adjustl(jtime2))
193     write(*, *)
194
195 end program debris
```

3 flopen.f90

flopen.f90

```
1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine flopen
14     use arrayc
15     implicit none
16
17     character(50) :: filex
18
19     open(10, file = 'filename.dat', status = 'old')
20     open(80, file = 'errorout.dat')
21
22     ! Open DEM file
23     read(10, 100) filex !, iox, filey
24     open(11, file = filex, status = 'old')
25     write(*, *) 'unit-11(in):DEMfile', filex
26
27     ! Open boundary file
28     read(10, 100) filex
29     write(*, *) 'unit-20(in):boundaryfile', filex
30     open(20, file = filex, status = 'old')
31
32     ! Open time file
33     read(10, 100) filex
34     write(*, *) 'unit-22(in):timefile', filex
35     open(22, file = filex, status = 'old')
36
37     ! Open hydrograph file
38     read(10, 100) filex
39     write(*, *) 'unit-12(in):hydrographfile', filex
40     open(12, file = filex, status = 'old')
41
42     ! Open parameter file
```

```

43  read(10, 100)  filex
44  write(*, *)  'unit-13(in):parameter_file', filex
45  open(13, file = filex, status = 'old')
46
47  ! Open result file (snap shot)
48  read(10, 100)  filex
49  write(*, *)  'unit-31(out):result_file(snap_shot)', filex
50  fsnap = filex(1:30)
51
52  ! Open result file (final state)
53  read(10, 100)  filex
54  write(*, *)  'unit-32(out):result_file(final_state)', filex
55  open(32, file = filex, status = 'unknown')
56
57  ! Open result file (max values)
58  read(10, 100)  filex
59  write(*, *)  'unit-33(out):result_file(max_values)', filex
60  open(33, file = filex, status = 'unknown')
61
62  ! Open check file for input status
63  read(10, 100)  filex
64  write(*, *)  'unit-34(out):check_file_for_input_status', filex
65  open(34, file = filex, status = 'unknown')
66
67  ! Open tau > tau_{*c} file using iwagaki eq. (in tau.f90)
68  read(10, 100)  filex
69  write(*, *)  'unit-99(out):tau_>tau_{*c}with_connectivity', filex
70  open(99, file = filex, status = 'unknown')
71
72  ! Open Check overflow volume (in main.f90)
73  read(10, 100)  filex
74  open(80, file = filex, status='unknown')
75  write(*, *)  'unit-80(out):Check_overflow_volume', filex
76
77  close(10)
78
79  100 format(a50)
80
81  return
82  end subroutine flopen

```

4 inputdem.f90

inputdem.f90

```

1  !-----
2  !
3  ! Copyright 2022
4  !   Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  !   Public Works Research Institute (PWRI)
6  !
7  ! License
8  !   Creative Common (CC) BY-SA 4.0
9  !   (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 !   (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine inputdem
14     use arrayc
15     implicit none
16
17     integer :: igrd, i
18     integer :: is
19     character(256) :: fm
20     character(1), allocatable :: outDir(:)
21
22
23     write(*,*) '▣reading▣dem'
24     !
25     ! scalar grid connectivity
26     !
27     !           +-----+           +-----+           +-----+
28     !           |       |           |       |           |       |
29     !           |  N  |           |   2   |           |  iNN |
30     !           |       |           |       |           |       |
31     !   +-----+ ^ +-----+ +-----+ 2 +-----+ +-----+ iNN +-----+
32     !   |       | |       | |       | |       | |       | |       | |       |
33     !   |  W > P > E  | |  3 3 P 1 1  | | iWW iWW P iEE iEE |
34     !   |       | |       | |       | |       | |       | |       | |       |
35     !   +-----+ ^ +-----+ +-----+ 4 +-----+ +-----+ iSS +-----+
36     !   |       | |       | |       | |       | |       | |       | |       |
37     !   |   S   | |       | |       | |       | |       | |       | |       |
38     !   |       | |       | |       | |       | |       | |       | |       |
39     !           +-----+           +-----+           +-----+
40     !
41     ! vector edge connectivity (x-direction)
42     !

```

```

43      !      +-----+-----+      +-----+-----+      +-----+-----+
44      !      |           |           |           |           |           |           |
45      !      |           >           |           |           2           |           |           |           |           |
46      !      |           |           |           |           |           |           |           |           |           |
47      !      +-----^-----+-----+      +-----2-----+-----1-----+      +-----iNW-----+-----iNE-----+
48      !      |           |           |           |           |           |           |           |           |           |
49      !      >   W   P   E   >           |           |           |           |           |           |           |           |           |
50      !      |           |           |           |           |           |           |           |           |           |
51      !      +-----^-----+-----+      +-----3-----+-----4-----+      +-----iSW-----+-----iSE-----+
52      !      |           |           |           |           |           |           |           |           |           |
53      !      |           >           |           |           |           |           |           |           |           |           |
54      !      |           |           |           |           |           |           |           |           |           |
55      !      +-----+-----+      +-----+-----+      +-----+-----+
56      !
57      ! vector edge connectivity (y-direction)
58      !
59      !      +-----+-----^-----+-----+      +-----+-----2-----+-----+      +-----+-----iNN-----+-----+
60      !      |           |           |           |           |           |           |           |           |           |           |
61      !      |           >   N   >           |           |           |           |           |           |           |           |           |
62      !      |           |           |           |           |           |           |           |           |           |           |
63      !      +-----^-----+-----P-----+-----+      +-----3-----+-----P-----+-----1-----+      +-----iEE-----+-----P-----+-----iEE-----+
64      !      |           |           |           |           |           |           |           |           |           |           |
65      !      |           >   S   >           |           |           |           |           |           |           |           |           |
66      !      |           |           |           |           |           |           |           |           |           |           |
67      !      +-----+-----^-----+-----+      +-----+-----4-----+-----+      +-----+-----iSS-----+-----+
68      !
69      ! iEE = 1, iNN = 2, iWW = 3, iSS = 4
70      ! iNE = 1, iNW = 2, iSW = 3, iSE = 4
71      ! iEg = 1, iNg = 1, iWg = 2, iSg = 2
72      !
73
74      read(11, '(3i10,2E25.15,i10)') Ngrid, Nxs, Nys, dx, dy, idz
75
76      fm = '("UUUUtotal_number_of_grid:U",U,i10,U"UUgrid_size:U",U2f10.5)'
77      write(*, fm) Ngrid, dx, dy
78
79
80      !— scalar arrays
81      allocate(igc(4, 0:Ngrid), iec(4, 0:Ngrid)) ! Neighbor grid and edge
82      allocate(lonlat(2, 0:Ngrid), xy(2, 0:Ngrid)) ! x, y position
83      allocate(ezi(0:Ngrid), zdp(0:Ngrid), &
84              &zl1(0:Ngrid), zl3(0:Ngrid), zdz(0:Ngrid))
85      allocate(vlo(0:Ngrid), vfo(0:Ngrid))
86      allocate(zvl(0:Ngrid), zvf(0:Ngrid), czl(0:Ngrid), czf(0:Ngrid))
87      allocate(hn0(0:Ngrid), hn1(0:Ngrid), hn3(0:Ngrid), hz(0:Ngrid))
88      allocate(rof(0:Ngrid), rot(0:Ngrid))
89      allocate(ct1(0:Ngrid), ct3(0:Ngrid), cl1(0:Ngrid), cl3(0:Ngrid), &

```

```

90         &cf1(0:Ngrid), cf3(0:Ngrid), clp(0:Ngrid))
91 allocate(smn(0:Ngrid), sei(0:Ngrid), aii(0:Ngrid), taus(0:Ngrid))
92 allocate(qqa(0:Ngrid), qmx(0:Ngrid), qny(0:Ngrid))
93 allocate(uux(0:Ngrid), vvy(0:Ngrid), uva(0:Ngrid))
94 allocate(fta(0:Ngrid), ftx(0:Ngrid), fty(0:Ngrid))
95 allocate(idxWg(0:Ngrid), WDg(0:Ngrid))
96
97 ! use this variable in tau, but keep memory here to avoid stack overflow
98 allocate(meshflag(0:Ngrid, 7))
99
100 !— vector arrays
101 allocate(ixsxc(4, 0:Nxs), ixsync(4, 0:Nxs), ixsgc(2, 0:Nxs))
102 allocate(iysxc(4, 0:Nys), iysyc(4, 0:Nys), iysgc(2, 0:Nys))
103 allocate(qm0(0:Nxs), qm1(0:Nxs), qm2(0:Nxs), &
104         &qn0(0:Nys), qn1(0:Nys), qn2(0:Nys))
105 allocate(uul(0:Nxs), vvl(0:Nys))
106 allocate(qblx(0:Nxs), qbfx(0:Nxs), qbly(0:Nys), qbfy(0:Nys))
107 allocate(idxWxs(0:Nxs), WDxs(0:Nxs), idxWys(0:Nys), WDys(0:Nys))
108
109 !— max values
110 allocate(arvt(0:Ngrid))
111 allocate(qqmax(0:Ngrid), qxmax(0:Ngrid), qymax(0:Ngrid), tqmax(0:Ngrid))
112 allocate(uvmax(0:Ngrid), vxm(0:Ngrid), vym(0:Ngrid), tvmax(0:Ngrid))
113 allocate(hmax(0:Ngrid), thmax(0:Ngrid), hzmax(0:Ngrid), thzmax(0:Ngrid), &
114         &hahz(0:Ngrid), zahz(0:Ngrid))
115 allocate(zmax(0:Ngrid), tzmax(0:Ngrid), zmin(0:Ngrid), tzmin(0:Ngrid))
116 allocate(fmax(0:Ngrid), fmx(0:Ngrid), fmy(0:Ngrid), tfmax(0:Ngrid), &
117         &hafmax(0:Ngrid), zafmax(0:Ngrid), tapc(0:Ngrid))
118 allocate(qblxm(0:Ngrid), qblym(0:Ngrid), qbla(0:Ngrid))
119 allocate(qbfxm(0:Ngrid), qbfym(0:Ngrid), qbfa(0:Ngrid))
120 !!!!!!!!!!!!!! allocate end !!!!!!!!!!!!!!!
121
122 !— Bed elevations
123 lonlat(:, :) = 0.d0
124 xy(:, :) = 0.d0
125 ezi(:) = 0.d0
126 hn0(:) = 0.d0
127
128 do i=0,Ngrid
129     read(11, '(6(E25.15))')xy(1:2,i),lonlat(1:2,i),ezi(i),hn0(i)
130 end do
131
132 !— Grid connectivities
133 igc(:, :) = 0
134 iec(:, :) = 0
135
136 do i=0,Ngrid

```



```

137     read(11, '(8i10)') igc(1:4, i), iec(1:4, i)
138 end do
139
140 !— Edge connectivities
141 ixsvc(:, :) = 0
142 ixvsc(:, :) = 0
143 ixsgc(:, :) = 0
144 ivsvc(:, :) = 0
145 ivvsc(:, :) = 0
146 ivsgc(:, :) = 0
147
148 do i=0, Nxs
149     read(11, '(10i10)') ixsvc(1:4, i), ivsvc(1:4, i), ixsgc(1:2, i)
150 end do
151 do i=0, Nys
152     read(11, '(10i10)') ivsvc(1:4, i), ivvsc(1:4, i), ivsgc(1:2, i)
153 end do
154
155 !-----
156
157 !— Outflow boundary
158
159 read(20, '(2i10)') nxout, nyout
160
161 if (nxout > 0) then
162     allocate(outDir(nxout), ixout(nxout), ixgout(nxout))
163     ixout(:) = 0
164     ixgout(:) = 0
165
166     do i=1, nxout
167         read(20, '(i10,a)') ixout(i), outDir(i)
168     end do
169
170     do i = 1, nxout
171         if (trim(adjustl(outDir(i))) == '+') then
172             ixgout(i) = ixsgc(iWg, ixout(i))
173             ! direction * reference grid index
174         else if (trim(adjustl(outDir(i))) == '-') then
175             ixgout(i) = -ixsgc(iEg, ixout(i))
176             ! direction * reference grid index
177         else
178             write(*, *) '□□Error:□outflow□direction□(x)', ixout(i)
179         end if
180     end do
181     deallocate(outDir)
182 end if
183

```

```

184     if (nyout > 0) then
185         allocate(outDir(nyout), iyout(nyout), iygout(nyout))
186         iyout(:) = 0
187         iygout(:) = 0
188
189         do i=1,nyout
190             read(20,'(i10,a)') iyout(i), outDir(i)
191         end do
192
193         do i = 1, nyout
194             if (trim(adjustl(outDir(i))) == '+') then
195                 iygout(i) =  iysgc(iSg, iyout(i))
196                 ! direction * reference grid index
197             else if(trim(adjustl(outDir(i))) == '-') then
198                 iygout(i) = - iysgc(iNg, iyout(i))
199                 ! direction * reference grid index
200             else
201                 write(*, *) 'Error: outflow direction(y)', iyout(i)
202             end if
203         end do
204         deallocate(outDir)
205     end if
206
207     !— Wall boundary
208
209     read(20,'(2i10)') nxwall, nywall
210     if (nxwall > 0) then
211         allocate(ixwall(nxwall))
212         ixwall(:) = 0
213         do i=1,nxwall
214             read(20,'(i10)') ixwall(i)
215         end do
216
217     end if
218     if (nywall > 0) then
219         allocate(iywall(nywall))
220         iywall(:) = 0
221         do i=1,nywall
222             read(20,'(i10)') iywall(i)
223         end do
224     end if
225
226
227     ! get terrain modification along channel
228     if (iox == 99) then
229         do
230             read(21, '(5(i5, f10.2))', iostat = is) igrd, ezi(igrd)

```

```
231         if (is < 0) exit
232     end do
233 end if
234
235     z11(:) = ezi(:)
236     z13(:) = ezi(:)
237     zdp(:) = ezi(:) - INIT_SOIL_DEPTH
238
239     return
240 end subroutine inputdem
```

5 inpuhydr.f90

inpuhydr.f90

```
1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine inpuhydr
14     use arrayc
15     implicit none
16     double precision :: a0, a1, a2, s0, t0, a3
17     integer :: i, i0, i1, is
18
19     ! process time data !
20
21     read(22, '(5x,␣i5)') mlf
22     write(34, '(5x,␣i5)') mlf
23     if (iout <= 0) iout = 0
24
25     ! read time data
26     do i = 1, mlf
27         read(22, '(4f10.0)') a0, a1, a2
28         write(34, '(4f10.3)') a0, a1, a2
29         tdlr(i) = a0
30         tdlr(i) = a1
31     end do
32
33     ! process hydr data !
34     qin(:, :) = 0.d0
35     hin(:, :) = 0.d0
36     sqin(:) = 0.d0
37     cli(:, :) = 0.d0
38     cfi(:, :) = 0.d0
39     mpat = 0
40
41     do
42         read(12, '(a5,␣2i5,␣f10.2)', iostat=is) mw, i0, i1, s0
```

```

43     if(is < 0 .or. mw == '*end*') return
44     write(34, '(a5, 2i5, f10.4)') mw, i0, i1, s0
45
46     mpat = mpat + 1
47     mhy(mpat) = mw
48
49     print *, "reading hydr"
50     print *, "no., name:", mpat, mhy(mpat)
51
52     !c error-check
53     ! check hydr no
54     if(mpat /= i0) call errnum('inputhydr-01')
55     ! check data no in each hydr
56     if(mlf /= i1) call errnum('inputhydr-02')
57     ! check slope of source point
58     if(s0 <= 0.d0) call errnum('inputhydr-03')
59     !c error-check
60
61     sqin(mpat) = s0
62
63     ! read hydr. data
64     ! do i = i0, i1 ! 1, mlf
65     do i = 1, i1
66         read(12, '(5f10.4)') t0, a1, a2, a3
67         write(34, '(5f10.4)') t0, a1, a2, a3
68         qin(mpat, i) = a1
69         cli(mpat, i) = a2
70         cfi(mpat, i) = a3
71     end do
72
73     end do
74
75     return
76 end subroutine inputhydr

```

6 inputparam.f90

inputparam.f90

```
1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine inputparam
14     use arrayc
15     implicit none
16
17     integer :: i, m
18
19     character(len=1) :: m1, m2
20
21     read(13, '(a10,i5)') mj
22     write(34, '(a10)')   mj
23
24     write(*,*) "  _reading_parameter "
25
26     read(13, '(a10,f10.0)')   mw_10, dm
27     write(34, '(a10,f10.4)') mw_10, dm
28     read(13, '(a10,f10.0)')   mw_10, sig
29     write(34, '(a10,f10.4)') mw_10, sig
30     read(13, '(a10,f10.0)')   mw_10, rho
31     write(34, '(a10,f10.4)') mw_10, rho
32     read(13, '(a10,f10.0)')   mw_10, sodo
33     write(34, '(a10,f10.4)') mw_10, sodo
34     read(13, '(a10,f10.0)')   mw_10, tanp
35     write(34, '(a10,f10.4)') mw_10, tanp
36     read(13, '(a10,f10.0)')   mw_10, cst
37     write(34, '(a10,f10.4)') mw_10, cst
38     read(13, '(a10,f10.0)')   mw_10, cstf
39     write(34, '(a10,f10.4)') mw_10, cstf
40     read(13, '(a10,f10.0)')   mw_10, csc
41     write(34, '(a10,f10.4)') mw_10, csc
42     read(13, '(a10,f10.0)')   mw_10, tauc
```

```

43  write(34, '(a10,f10.4)') mw_10, tauc
44  read(13, '(a10,f10.0)') mw_10, bit
45  write(34, '(a10,f10.4)') mw_10, bit
46  read(13, '(a10,f10.0)') mw_10, bit_d
47  write(34, '(a10,f10.4)') mw_10, bit_d
48  read(13, '(a10,f10.0)') mw_10, th
49  write(34, '(a10,f10.4)') mw_10, th
50  read(13, '(a10,f10.0)') mw_10, bm
51  write(34, '(a10,f10.4)') mw_10, bm
52  read(13, '(a10,i10)') mw_10, flg_rebed
53  write(34, '(a10,i10)') mw_10, flg_rebed
54  read(13, '(a10,i10)') mw_10, iout
55  write(34, '(a10,i10)') mw_10, iout
56
57  ! sig and rho kg m^{-3} -> g/cm^3; only these unit is not SI
58  ! not rational, but inheriting original code
59  sig = sig/1000.0
60  rho = rho/1000.0
61
62  c90 = cst * 0.9d0
63  wsg = (sig - rho) / rho
64  gsr = sig / rho
65
66  ! c_{*DL}
67  cdl = (cst - cstf)/(1.d0 - cstf)
68  cstl = cdl
69
70  ! 0.4 C_{*L}
71  cdc = cstl * 0.4d0
72
73
74  read(13, '(a10)') mj
75  write(34, '(a10)') mj
76
77  read(13, '(a5,i5)') mw, nbc
78  write(34, '(a5,i5)') mw, nbc
79
80  vwi(:) = 0.d0 ! Total flow discharge at each inflow point
81  vli(:) = 0.d0 ! Total sediment discharge (coase particles) at each inflow
point
82  vfi(:) = 0.d0 ! Total sediment discharge (fine particles) at each inflow
point
83
84  twi = 0.d0 ! Total flow discharge
85  tli = 0.d0 ! Total sediment discharge (coase particles)
86  tfi = 0.d0 ! Total sediment discharge (fine particles)
87

```

```

88     nxbc = 0
89     nybc = 0
90     ihyf(:) = 0
91     if (nbc > 0) then
92         do i = 1, nbc
93
94             read(13, '(a5,i10,2x,a1,1x,a1)') mw, ibg(i), m1, m2
95             write(34, '(a5,i10,2x,a1,1x,a1)') mw, ibg(i), m1, m2
96
97             do m = 1, mpat
98                 if(mw == mhy(m)) ihyf(i) = m
99             end do
100             !c error-check
101             if(ihyf(i) == 0) then
102                 write(*, *) &
103                     & '  合致するハイドロ名がない:', trim(mw)
104                 ! write(*, *) ' not match the hydr name '
105                 write(*, *) '  Cannot find the hydr name:', trim(mw)
106                 ! pause
107                 stop
108             endif
109
110             if      (m1 == '+') then ! X+ (east) direction
111                 nxbc = nxbc + 1
112                 ixhyf(nxbc) = ihyf(i)
113                 ixbg(nxbc) = ibg(i)
114                 ixbd(nxbc) = iEE
115                 qmin(nxbc, :) = qin(i, :)
116             else if (m2 == '+') then ! Y+ (north) direction
117                 nybc = nybc + 1
118                 iyhyf(nybc) = ihyf(i)
119                 iybg(nybc) = ibg(i)
120                 iybd(nybc) = iNN
121                 qnin(nybc, :) = qin(i, :)
122             else if (m1 == '-') then ! X- (west) direction
123                 nxbc = nxbc + 1
124                 ixhyf(nxbc) = ihyf(i)
125                 ixbg(nxbc) = ibg(i)
126                 ixbd(nxbc) = iWW
127                 qmin(nxbc, :) = - qin(i, :)
128             else if (m2 == '-') then ! Y- (south) direction
129                 nybc = nybc + 1
130                 iyhyf(nybc) = ihyf(i)
131                 iybg(nybc) = ibg(i)
132                 iybd(nybc) = iSS
133                 qnin(nybc, :) = - qin(i, :)
134             else

```



```
135         write(*, *) '  水流入方向エラー'
136         write(*, *) '  Error:inflow_direction'
137     end if
138
139     end do
140 end if
141 close(34)
142 return
143 end subroutine inputparam
```

7 clear.f90

clear.f90

```
1  !  
2  !  
3  ! Copyright 2022  
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and  
5  ! Public Works Research Institute (PWRI)  
6  !  
7  ! License  
8  ! Creative Common (CC) BY-SA 4.0  
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja  
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode  
11 !  
12 !  
13 subroutine clear  
14     use arrayc  
15     implicit none  
16     !$omp parallel  
17  
18     !$omp workshare  
19     ! clear01  
20     qm2(:) = 0.d0  
21     qn2(:) = 0.d0  
22     qm1(:) = 0.d0  
23     qn1(:) = 0.d0  
24     qm0(:) = 0.d0  
25     qn0(:) = 0.d0  
26     qmx(:) = 0.d0  
27     qny(:) = 0.d0  
28     qqa(:) = 0.d0  
29  
30     uva(:) = 0.d0  
31     uux(:) = 0.d0  
32     vvy(:) = 0.d0  
33     sei(:) = 0.d0  
34     uul(:) = 0.d0  
35     vvl(:) = 0.d0  
36  
37     qblx(:) = 0.d0  
38     qbly(:) = 0.d0  
39     qbfx(:) = 0.d0  
40     qbfy(:) = 0.d0  
41  
42     aii(:) = 0.d0
```

```

43
44      ! clear02
45      hn1 (:) = hn0 (:)
46      hn3 (:) = hn0 (:)
47      hz (:) = 0.d0
48      aii (:) = 0.d0
49      clp (:) = 0.d0
50      cl1 (:) = 0.d0
51      cf1 (:) = 0.d0
52      cl3 (:) = 0.d0
53      cf3 (:) = 0.d0
54      ct1 (:) = 0.d0
55      ct3 (:) = 0.d0
56      rof (:) = 1.d0
57      rot (:) = 1.d0
58      zdz (:) = 0.d0
59
60      ! clear03
61      arvt (:) = 0.d0
62      qqmax (:) = 0.d0
63      tqmax (:) = 0.d0
64      qxmax (:) = 0.d0
65      qymax (:) = 0.d0
66      uvmax (:) = 0.d0
67      tvmax (:) = 0.d0
68      vxm (:) = 0.d0
69      vym (:) = 0.d0
70      hmax (:) = 0.d0
71      thmax (:) = 0.d0
72      hzmax (:) = 0.d0
73      thzmax (:) = 0.d0
74      hahz (:) = 0.d0
75      zahz (:) = 0.d0
76      zmax (:) = 0.d0
77      tzmax (:) = 0.d0
78      zmin (:) = 0.d0
79      tzmin (:) = 0.d0
80      fmax (:) = 0.d0
81      tfmax (:) = 0.d0
82      fmx (:) = 0.d0
83      fmy (:) = 0.d0
84      hafmax (:) = 0.d0
85      zafmax (:) = 0.d0
86      qblxm (:) = 0.d0
87      qblym (:) = 0.d0
88      qbla (:) = 0.d0
89      qbfxm (:) = 0.d0

```

```
90     qbfym(:) = 0.d0
91     qbfa(:) = 0.d0
92
93     !$omp end workshare
94
95     !$omp end parallel
96
97     return
98 end subroutine clear
```

8 calset.f90

calset.f90

```
1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine calset
14     use arrayc
15     implicit none
16     integer i, m, ie, ib
17
18     eps = th * th
19
20     !$omp parallel
21
22     !$omp workshare
23     zvl(:) = 0.d0
24     zvf(:) = 0.d0
25     vlo(:) = 0.d0
26     vfo(:) = 0.d0
27     smn(:) = sodo
28     czl(:) = cstl
29     czf(:) = cstf
30     !$omp end workshare
31
32     !$omp end parallel
33
34
35     !— set unit width discharge and normal depth at inflow points
36     do m = 1, mlf
37         do i = 1, nxbc
38             qmin(i, m) = qmin(i, m) / dy
39             hin(ixhyf(i), m) = (abs(qmin(i, m)) * sodo / sqrt(sqin(ixhyf(i)
40 )))**0.6d0
41             if (hin(ixhyf(i), m) > th) then
42                 uin(i, m) = qmin(i, m) / hin(ixhyf(i), m)
```

```

42         end if
43     end do
44 end do
45 do m = 1, mlf
46     do i = 1, nybc
47         qnin(i, m) = qnin(i, m) / dx
48         hin(iyhyf(i), m) = (abs(qnin(i, m)) * sodo / sqrt(sqin(iyhyf(i
49 )))**0.6d0
50         if (hin(iyhyf(i), m) > th) then
51             vin(i, m) = qnin(i, m) / hin(iyhyf(i), m)
52         end if
53     end do
54 end do
55 do m = 1, mlf
56     do i = 1, nbc
57         ib = ibg(i)
58         hn0(ib) = hin(i, m)
59         hn1(ib) = hin(i, m)
60         clb(m) = (cli(i, m) + cli(i, m+1))*0.5
61         cfb(m) = (cfi(i, m) + cfi(i, m+1))*0.5
62     end do
63     do i = 1, nxbc
64         ie = iec(ixbd(i), ixbg(i))
65         qm0(ie) = qmin(i, m)
66         qm1(ie) = qmin(i, m)
67         uul(ie) = uin(i, m)
68         uux(ixbg(i)) = uin(i, m)
69         qblxin(i, m) = qmin(i, m) * clb(m)
70         qbfxin(i, m) = qmin(i, m) * cfb(m) * (1.d0 - clb(m))
71     end do
72     do i = 1, nybc
73         ie = iec(iybd(i), iybg(i))
74         qn0(ie) = qnin(i, m)
75         qn1(ie) = qnin(i, m)
76         vvl(ie) = vin(i, m)
77         vvy(iybg(i)) = vin(i, m)
78         qblyin(i, m) = qnin(i, m) * clb(m)
79         qbfyin(i, m) = qnin(i, m) * cfb(m) * (1.d0 - clb(m))
80     end do
81 end do
82
83 !— compute total inflow discharge
84 do m = 1, mlf
85     do i = 1, nxbc
86         vwi(m) = vwi(m) + abs(qmin(i, m)) * dy * tdlt(m)
87         vli(m) = vli(m) + abs(qmin(i, m)) * dy * tdlt(m) * clb(m)

```

```

88         vfi(m) = vfi(m) + abs(qmin(i , m)) * dy * tdlt(m) * cfb(m) * (1.d0 - clb
(m))
89     end do
90     do i = 1, nybc
91         vwi(m) = vwi(m) + abs(qnin(i , m)) * dx * tdlt(m)
92         vli(m) = vli(m) + abs(qnin(i , m)) * dx * tdlt(m) * clb(m)
93         vfi(m) = vfi(m) + abs(qnin(i , m)) * dx * tdlt(m) * cfb(m) * (1.d0 - clb
(m))
94     end do
95 end do
96
97     twi = sum(vwi)
98     tli = sum(vli)
99     tfi = sum(vfi)
100    if ((tli + tfi) < eps) noSediment = .true.
101
102     voutflow = 0.d0
103
104     !— set wet grid
105     call wetdry(hn0)
106
107     return
108 end subroutine calset

```

9 wetdry.f90

wetdry.f90

```
1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine wetdry(hh)
14     !$ use omp_lib
15     use arrayc
16     implicit none
17
18     real(8), intent(in) :: hh(0:Ngrid)
19
20     integer :: i, ib, ig
21     integer :: ig_e, ig_n, ig_w, ig_s, is_e, is_n, is_w, is_s
22     double precision :: eta
23
24     !$omp parallel
25
26     !$omp do
27     do i = 0, Ngrid
28         WDg(i) = isDry
29     end do
30     !$omp end do
31     !$omp do
32     do i = 0, Nxs
33         WDxs(i) = isDry
34     end do
35     !$omp end do
36     !$omp do
37     do i = 0, Nys
38         WDys(i) = isDry
39     end do
40     !$omp end do
41     !$omp do
42     do i = 0, Ngrid
```



```

43     smn(i) = sodo
44 end do
45 !$omp end do
46
47 ! /// Check water depth
48 !$omp do private(i, ib, ig_e, ig_n, ig_w, ig_s, is_e, is_n, is_w, is_s, eta)
49 gridLoop: do i = 1, Ngrid
50
51     if(hh(i) > th) then
52         do ib = 1, nbc
53             if (i == ibg(ib)) cycle gridLoop
54         end do
55         ig_e = igc(iEE, i)
56         ig_n = igc(iNN, i)
57         ig_w = igc(iWW, i)
58         ig_s = igc(iSS, i)
59
60         is_e = iec(iEE, i)
61         is_n = iec(iNN, i)
62         is_w = iec(iWW, i)
63         is_s = iec(iSS, i)
64
65         eta = z13(i) + hh(i)
66
67         WDg(i) = isWet
68         WDg(ig_e) = isWet
69         WDg(ig_n) = isWet
70         WDg(ig_w) = isWet
71         WDg(ig_s) = isWet
72         WDxs(is_e) = isWet
73         WDys(is_n) = isWet
74         WDxs(is_w) = isWet
75         WDys(is_s) = isWet
76
77     end if
78 end do gridLoop
79 !$omp end do
80
81 !$omp end parallel
82
83 ! Inflow grid is out of computation
84 if (nxbc > 0) then
85     do i = 1, nxbc
86         ib = ixbg(i)
87         WDg(ib) = isDry
88     end do
89 end if

```

```

90  if (nybc > 0) then
91      do i = 1, nybc
92          ib = iybg(i)
93          WDg(ib) = isDry
94      end do
95  end if
96
97  ! Down-stream inflow discharge is wet if boundary grid has sufficient water
depth
98  if (nxbc > 0) then
99      do i = 1, nxbc
100         ib = ixbg(i)
101         ig = igc(ixbd(i), ib)
102         if (hh(ib) > th .and. WDg(ig) /= isWet) then
103             WDg(ig) = isWet
104         end if
105     end do
106 end if
107 if (nybc > 0) then
108     do i = 1, nybc
109         ib = iybg(i)
110         ig = igc(iybd(i), ib)
111         if (hh(ib) > th .and. WDg(ig) /= isWet) then
112             WDg(ig) = isWet
113         end if
114     end do
115 end if
116
117  ! Dummy grid and edge for out of computational domain
118  WDg(0) = isDry
119  WDxs(0) = isDry
120  WDys(0) = isDry
121
122  Nwg = 0
123  Nwxs = 0
124  Nwys = 0
125
126  ! /// Make index of wet grids
127  do i = 1, Ngrid
128      if(WDg(i) == isWet) then
129          Nwg = Nwg + 1
130          idxWg(Nwg) = i
131      end if
132  end do
133
134  ! /// Make index of wet edges
135  do i = 1, Nxs

```

```
136     if (WDxs(i) == isWet) then
137         Nwxs = Nwxs + 1
138         idxWxs(Nwxs) = i
139     end if
140 end do
141 do i = 1, Nys
142     if (WDys(i) == isWet) then
143         Nwys = Nwys + 1
144         idxWys(Nwys) = i
145     end if
146 end do
147
148 return
149 end subroutine wetdry
```

10 qmflux.f90

qmflux.f90

```
1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine qmflux()
14   use arrayc
15   implicit none
16
17   integer :: i, is
18   integer :: isxE, isxN, isxW, isxS, isyNE, isyNW, isySW, isySE
19   integer :: igE, igW
20   double precision :: uxe, uxw, vne, vnw, vsw, vse
21   double precision :: xmp, xme, xmn, xmw, xms
22   double precision :: vmf, vmb
23   double precision :: wx, wy
24   double precision :: xdx, xdy
25   double precision :: hj3, hw0, hdx, ccl, sqx, frx
26   double precision :: a1, a2, frq2, frq1, sor
27
28   !$omp parallel
29
30   !$omp do private(i)
31   do i = 0, Nxs
32     qm2(i) = 0.d0
33   end do
34   !$omp end do
35
36   !$omp do private(i, is, isxE, isxN, isxW, isxS, isyNE, isyNW, isySW, isySE,
37   igE, igW, &
38   uxe, uxw, vne, vnw, vsw, vse, xmp, xme, xmn, xmw, xms, xdx,
39   xdy, &
40   vmf, vmb, hj3, hw0, hdx, ccl, wx, wy, sqx, sor, frx, frq2,
41   frq1)
42   do i = 1, Nwxs
```

```

40     is = idxWxs(i)
41
42     isxE = ixsrc(iEE, is)
43     isxN = ixsrc(iNN, is)
44     isxW = ixsrc(iWW, is)
45     isxS = ixsrc(iSS, is)
46
47     isyNE = isyc(iNE, is)
48     isyNW = isyc(iNW, is)
49     isySW = isyc(iSW, is)
50     isySE = isyc(iSE, is)
51
52     igE = ixsgc(iEg, is)
53     igW = ixsgc(iWg, is)
54
55     uxe = (uul(is ) + uul(isxE)) * 0.5d0
56     uxw = (uul(isxW) + uul(is )) * 0.5d0
57
58     vne = vvl(isyNE)
59     vnw = vvl(isyNW)
60     vsw = vvl(isySW)
61     vse = vvl(isySE)
62
63     xmp = (qm1(is ) + qm0(is )) * 0.5d0
64     xme = (qm1(isxE) + qm0(isxE)) * 0.5d0
65     xmn = (qm1(isxN) + qm0(isxN)) * 0.5d0
66     xmw = (qm1(isxW) + qm0(isxW)) * 0.5d0
67     xms = (qm1(isxS) + qm0(isxS)) * 0.5d0
68
69     !— xdx
70     if (uxe >= 0.d0 .and. uxw >= 0.d0) then
71         xdx = uxe * xmp - uxw * xmw
72     else if (uxe >= 0.d0 .and. uxw < 0.d0) then
73         xdx = uxe * xmp - uxw * xmp
74     else if (uxe < 0.d0 .and. uxw >= 0.d0) then
75         xdx = uxe * xme - uxw * xmw
76     else
77         xdx = uxe * xme - uxw * xmp
78     end if
79     xdx = xdx / dx
80
81     !— xdy
82     if (vne >= 0.d0 .and. vnw < 0.d0 .and. xmp >= 0.d0 .and. xmn < 0.d0)
then
83         vmf = (vne * xmp + vnw * xmn) * 0.5d0
84     else if (vne < 0.d0 .and. vnw >= 0.d0 .and. xmp < 0.d0 .and. xmn >= 0.d0)
then

```

```

85         vmf = (vne * xmn + vnw * xmp) * 0.5 d0
86     else if (vne >= 0.d0 .and. vnw < 0.d0 .and. xmp >= 0.d0 .and. xmn >= 0.d0)
then
87         vmf = vne * xmp
88     else if (vne >= 0.d0 .and. vnw >= 0.d0 .and. xmp >= 0.d0
)
then
89         vmf = vne * xmp
90     else if (vne < 0.d0 .and. vnw >= 0.d0 .and. xmp >= 0.d0 .and. xmn >= 0.d0)
then
91         vmf = vne * xmn
92     else if (vne < 0.d0 .and. vnw < 0.d0
.and. xmn >= 0.d0)
then
93         vmf = vne * xmn
94     else if (vne >= 0.d0 .and. vnw < 0.d0 .and. xmp < 0.d0 .and. xmn < 0.d0)
then
95         vmf = vnw * xmn
96     else if (vne < 0.d0 .and. vnw < 0.d0
.and. xmn < 0.d0)
then
97         vmf = vnw * xmn
98     else if (vne < 0.d0 .and. vnw >= 0.d0 .and. xmp < 0.d0 .and. xmn < 0.d0)
then
99         vmf = vnw * xmp
100    else if (vne >= 0.d0 .and. vnw >= 0.d0 .and. xmp < 0.d0
)
then
101        vmf = vnw * xmp
102    else
103        vmf = 0.d0
104    end if
105
106    if (vse >= 0.d0 .and. vsw < 0.d0 .and. xms >= 0.d0 .and. xmp < 0.d0)
then
107        vmb = (vse * xms + vsw * xmp) * 0.5 d0
108    else if (vse < 0.d0 .and. vsw >= 0.d0 .and. xms < 0.d0 .and. xmp >= 0.d0)
then
109        vmb = (vse * xmp + vsw * xms) * 0.5 d0
110    else if (vse >= 0.d0 .and. vsw < 0.d0 .and. xms >= 0.d0 .and. xmp >= 0.d0)
then
111        vmb = vse * xms
112    else if (vse >= 0.d0 .and. vsw >= 0.d0 .and. xms >= 0.d0
)
then
113        vmb = vse * xms
114    else if (vse < 0.d0 .and. vsw >= 0.d0 .and. xms >= 0.d0 .and. xmp >= 0.d0)
then
115        vmb = vse * xmp
116    else if (vse < 0.d0 .and. vsw < 0.d0
.and. xmp >= 0.d0)
then
117        vmb = vse * xmp

```

```

118     else if (vse >= 0.d0 .and. vsw < 0.d0 .and. xms < 0.d0 .and. xmp < 0.d0)
119     then
120         vmb = vsw * xmp
121     else if (vse < 0.d0 .and. vsw < 0.d0 .and. xmp < 0.d0)
122     then
123         vmb = vsw * xmp
124     else if (vse < 0.d0 .and. vsw >= 0.d0 .and. xms < 0.d0 .and. xmp < 0.d0)
125     then
126         vmb = vsw * xms
127     else if (vse >= 0.d0 .and. vsw >= 0.d0 .and. xms < 0.d0 )
128     then
129         vmb = vsw * xms
130     else
131         vmb = 0.d0
132     end if
133
134 xdy = (vmf - vmb) / dy
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160

```

if (ccl <= csc) **then**
 sor = (smn(igW) + smn(igE)) * 0.5d0
 frx = gr * sor * sor * sqx / hj3**r43
 frq2 = 1.d0 - dt1 * frx
 frq1 = 1.d0 + dt1 * frx
 qm2(is) = (frq2 * qm1(is) - dt2 * (xdx + xdy + hdx)) / frq1

else if (csc < ccl .and. ccl <= cdc) **then**
 frx = dm * dm * sqx / (0.49d0 * hj3 * hj3 * hj3)
 frq2 = 1.d0 - dt1 * frx
 frq1 = 1.d0 + dt1 * frx
 qm2(is) = (frq2 * qm1(is) - dt2 * (xdx + xdy + hdx)) / frq1

else if (cdc < ccl) **then**
 a1 = ccl + (1.d0 - ccl) * (rof(igW) + rof(igE)) * 0.5d0 / sig
 a2 = (cdl / ccl)**r13 - 1.d0
 frx = 0.125d0 * dm * dm * sqx / (a1 * a2 * a2 * hj3 * hj3 * hj3)

```
161         frq2    = 1.d0 - dt1 * frx
162         frq1    = 1.d0 + dt1 * frx
163         qm2(is) = (frq2 * qm1(is) - dt2 * (bm * xdx + bm * xdy + hdx)) / frq1
164
165     else
166         write(*, *) 'Flow type error.'
167         stop
168     end if
169
170
171 end do
172 !$omp end do
173
174 !$omp end parallel
175
176 return
177
178 end subroutine qmflux
```

11 qnflux.f90

qnflux.f90

```
1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine qnflux()
14     use arrayc
15     implicit none
16
17     integer :: i, is
18     integer :: isyE, isyN, isyW, isyS, isxNE, isxNW, isxSW, isxSE
19     integer :: igN, igS
20     double precision :: vyn, vys, une, unw, usw, use
21     double precision :: xnp, xne, xnn, xnw, xns
22     double precision :: unf, unb
23     double precision :: wx, wy
24     double precision :: ydx, ydy
25     double precision :: hj3, hw0, hdy, ccl, sqy, frx
26     double precision :: a1, a2, frq2, frq1, sor
27
28     !$omp parallel
29
30     !$omp do private(i)
31     do i = 0, Nys
32         qn2(i) = 0.d0
33     end do
34     !$omp end do
35
36     !$omp do private(i, is, isyE, isyN, isyW, isyS, isxNE, isxNW, isxSW, isxSE,
37 igN, igS, &
38 !$omp &
39     vyn, vys, une, unw, usw, use, xnp, xne, xnn, xnw, xns, ydy,
40 ydx, &
41 !$omp &
42 unf, unb, hj3, hw0, hdy, ccl, wx, wy, sqy, sor, frx, frq2,
43 frq1)
44     do i = 1, Nwys
```

```

40
41     is = idxWys(i)
42
43     isyE = iysyc(iEE, is)
44     isyN = iysyc(iNN, is)
45     isyW = iysyc(iWW, is)
46     isyS = iysyc(iSS, is)
47
48     isxNE = isxyc(iNE, is)
49     isxNW = isxyc(iNW, is)
50     isxSW = isxyc(iSW, is)
51     isxSE = isxyc(iSE, is)
52
53     igN = isgyc(iNg, is)
54     igS = isgyc(iSg, is)
55
56     vyn = (vvl(is ) + vvl(isyN)) * 0.5d0
57     vys = (vvl(isyS) + vvl(is )) * 0.5d0
58
59     une = uul(isxNE)
60     unw = uul(isxNW)
61     usw = uul(isxSW)
62     use = uul(isxSE)
63
64     xnp = (qn1(is ) + qn0(is )) * 0.5d0
65     xne = (qn1(isyE) + qn0(isyE)) * 0.5d0
66     xnn = (qn1(isyN) + qn0(isyN)) * 0.5d0
67     xnw = (qn1(isyW) + qn0(isyW)) * 0.5d0
68     xns = (qn1(isyS) + qn0(isyS)) * 0.5d0
69
70     !— ydy
71     if (vyn >= 0.d0 .and. vys >= 0.d0) then
72         ydy = vyn * xnp - vys * xns
73     else if (vyn >= 0.d0 .and. vys < 0.d0) then
74         ydy = vyn * xnp - vys * xnp
75     else if (vyn < 0.d0 .and. vys >= 0.d0) then
76         ydy = vyn * xnn - vys * xns
77     else
78         ydy = vyn * xnn - vys * xnp
79     end if
80     ydy = ydy / dy
81
82     !— ydx
83     if (une >= 0.d0 .and. use < 0.d0 .and. xnp >= 0.d0 .and. xne < 0.d0)
then
84         unf = (une * xnp + use * xne) * 0.5d0

```

```

85     else if (une < 0.d0 .and. use >= 0.d0 .and. xnp < 0.d0 .and. xne >= 0.d0)
then
86         unf = (une * xne + use * xnp) * 0.5d0
87     else if (une >= 0.d0 .and. use < 0.d0 .and. xnp >= 0.d0 .and. xne >= 0.d0)
then
88         unf = une * xnp
89     else if (une >= 0.d0 .and. use >= 0.d0 .and. xnp >= 0.d0
)
then
90         unf = une * xnp
91     else if (une < 0.d0 .and. use >= 0.d0 .and. xnp >= 0.d0 .and. xne >= 0.d0)
then
92         unf = une * xne
93     else if (une < 0.d0 .and. use < 0.d0
.and. xne >= 0.d0)
then
94         unf = une * xne
95     else if (une >= 0.d0 .and. use < 0.d0 .and. xnp < 0.d0 .and. xne < 0.d0)
then
96         unf = use * xne
97     else if (une < 0.d0 .and. use < 0.d0
.and. xne < 0.d0)
then
98         unf = use * xne
99     else if (une < 0.d0 .and. use >= 0.d0 .and. xnp < 0.d0 .and. xne < 0.d0)
then
100        unf = use * xnp
101     else if (une >= 0.d0 .and. use >= 0.d0 .and. xnp < 0.d0
)
then
102        unf = use * xnp
103     else
104        unf = 0.d0
105     end if
106
107     if (unw >= 0.d0 .and. usw < 0.d0 .and. xnw >= 0.d0 .and. xnp < 0.d0)
then
108        unb = (unw * xnw + usw * xnp) * 0.5d0
109     else if (unw < 0.d0 .and. usw >= 0.d0 .and. xnw < 0.d0 .and. xnp >= 0.d0)
then
110        unb = (unw * xnp + usw * xnw) * 0.5d0
111     else if (unw >= 0.d0 .and. usw < 0.d0 .and. xnw >= 0.d0 .and. xnp >= 0.d0)
then
112        unb = unw * xnw
113     else if (unw >= 0.d0 .and. usw >= 0.d0 .and. xnw >= 0.d0
)
then
114        unb = unw * xnw
115     else if (unw < 0.d0 .and. usw >= 0.d0 .and. xnw >= 0.d0 .and. xnp >= 0.d0)
then
116        unb = unw * xnp

```

```

117     else if (unw < 0.d0 .and. usw < 0.d0                               .and. xnp >= 0.d0)
118     then
119         unb = unw * xnp
120     else if (unw >= 0.d0 .and. usw < 0.d0 .and. xnw < 0.d0 .and. xnp < 0.d0)
121     then
122         unb = usw * xnp
123     else if (unw < 0.d0 .and. usw < 0.d0                               .and. xnp < 0.d0)
124     then
125         unb = usw * xnp
126     else if (unw >= 0.d0 .and. usw >= 0.d0 .and. xnw < 0.d0          )
127     then
128         unb = usw * xnw
129     else
130         unb = 0.d0
131     end if
132
133     ydx = (unf - unb) / dx
134
135     hj3 = (hn1(igN) + hn1(igS)) * 0.5d0
136     hw0 = (hn1(igN) + z11(igN)) - (hn1(igS) + z11(igS))
137
138     hdy = gr * hw0 * hj3 / dy
139
140     ccl = (c11(igN) + c11(igS)) * 0.5d0
141     wx = (une + unw + usw + use) * 0.25d0
142     wy = vv1(is)
143     sqy = sqrt(wx*wx + wy*wy)
144
145     if (ccl <= csc) then
146         sor = (smn(igN) + smn(igS)) * 0.5d0
147         frx = gr * sor * sor * sqy / hj3**r43 ! /cbrt(hj3**4)
148         frq2 = 1.d0 - dt1 * frx
149         frq1 = 1.d0 + dt1 * frx
150         qn2(is) = (frq2 * qn1(is) - dt2 * (ydx + ydy + hdy)) / frq1
151
152     else if (csc < ccl .and. ccl <= cdc) then
153         frx = dm * dm * sqy / (0.49d0 * hj3 * hj3 * hj3)
154         frq2 = 1.d0 - dt1 * frx
155         frq1 = 1.d0 + dt1 * frx
156         qn2(is) = (frq2 * qn1(is) - dt2 * (ydx + ydy + hdy)) / frq1
157
158     else if (cdc < ccl) then

```

```

159         a1 = ccl + (1.d0 - ccl) * (rof(igN) + rof(igS)) * 0.5d0 / sig ! /2.0/
      sig
160         a2 = (cd1 / ccl)**r13 - 1.d0
161         frx = 0.125d0 * dm * dm * sqy / (a1 * a2 * a2 * hj3 * hj3 * hj3)
162         frq2 = 1.d0 - dt1 * frx
163         frq1 = 1.d0 + dt1 * frx
164         qn2(is) = (frq2 * qn1(is) - dt2 * (bm * ydx + bm * ydy + hdy)) / frq1
165
166     else
167         write(*, *) "□Flow□type□error."
168         stop
169     end if
170
171
172 end do
173 !$omp end do
174
175 !$omp end parallel
176
177 return
178
179 end subroutine qnflux

```

12 boundaryq.f90

boundaryq.f90

```
1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine boundaryq(m)
14     use arrayc
15     implicit none
16     integer, intent(in) :: m
17
18     integer i
19     double precision :: hh
20     double precision, parameter :: coef = 1.6d0 ! 堰の公式の係数
21
22
23     ! Computation of overflow discharge on the edge of the computational domain
24     if (nxout > 0) then
25         do i = 1, nxout
26             hh = hn1(abs(ixgout(i)))
27             if (hh < th) cycle
28             qm2(ixout(i)) = dble(sign(1, ixgout(i))) * coef * sqrt(hh*hh*hh)
29             voutflow = voutflow + abs(qm2(ixout(i)) * dy * dt2)
30         end do
31     end if
32     if (nyout > 0) then
33         do i = 1, nyout
34             hh = hn1(abs(iygout(i)))
35             if (hh < th) cycle
36             qn2(iyout(i)) = dble(sign(1, iygout(i))) * coef * sqrt(hh*hh*hh)
37             voutflow = voutflow + abs(qn2(iyout(i)) * dx * dt2)
38         end do
39     end if
40
41     ! x-direction inflow discharge
42     if (nxbc > 0) then
```

```

43         do i = 1, nxbc
44             qm2(iec(      :, ixbg(i))) = 0.d0
45             qm2(iec(ixbd(i), ixbg(i))) = qmin(i, m)
46         end do
47     end if
48
49     ! y-direction inflow discharge
50     if (nybc > 0) then
51         do i = 1, nybc
52             qn2(iec(      :, iybg(i))) = 0.d0
53             qn2(iec(iybd(i), iybg(i))) = qnin(i, m)
54         end do
55     end if
56
57     ! x-direction wall boundary
58     if (nxwall > 0) then
59         do i = 1, nxwall
60             qm2(ixwall(i)) = 0.d0
61         end do
62     end if
63
64     ! y-direction wall boundary
65     if (nywall > 0) then
66         do i = 1, nywall
67             qn2(iywall(i)) = 0.d0
68         end do
69     end if
70
71     return
72
73 end subroutine boundaryq

```

13 hdepth.f90

hdepth.f90

```
1  !
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !
13 subroutine hdepth
14     use arrayc
15     implicit none
16     integer :: i, ig
17     integer :: ioe(4)
18     integer :: ig_e, ig_n, ig_w, ig_s
19     integer :: is_e, is_n, is_w, is_s
20     integer :: callCount
21     double precision :: qme, qmw, qnn, qns
22     double precision :: dqm, dqn
23     double precision :: dxdy, vneg, v_out, vout_e, vout_n, vout_w, vout_s
24     double precision :: dh_e, dh_n, dh_w, dh_s
25
26     !$omp parallel
27
28     !$omp do private(i)
29     do i = 0, Ngrid
30         hn3(i) = 0.d0
31     end do
32     !$omp end do
33
34     !$omp do private(i, ig, qme, qmw, qnn, qns, dqm, dqn)
35     do i = 1, nWg
36         ig = idxWg(i)
37
38         qme = qm2(iec(iEE, ig))
39         qmw = qm2(iec(iWW, ig))
40         qnn = qn2(iec(iNN, ig))
41         qns = qn2(iec(iSS, ig))
42
```



```

43     dqm = (qme - qmw) / dx
44     dqn = (qnn - qns) / dy
45     hn3(ig) = hn1(ig) - dt2 * (dqm + dqn - aii(ig))
46
47 end do
48 !$omp end do
49 !$omp end parallel
50
51 if (minval(hn3) < 0.d0) then
52     dx dy = dx * dy
53     !$omp parallel do private(ig, ig_e, ig_n, ig_w, ig_s, &
54     !$omp &                               is_e, is_n, is_w, is_s, &
55     !$omp &                               dh_e, dh_n, dh_w, dh_s, &
56     !$omp &                               qme, qnn, qmw, qns, vneg, &
57     !$omp &                               v_out, vout_e, vout_n, vout_w, vout_s,
58     &                               ioe), reduction(+: vmn)
59     do i = 1, nWg
60         ig = idxWg(i)
61         if (hn3(ig) < 0.d0) then
62
63             ig_e = igc(iEE, ig)
64             ig_n = igc(iNN, ig)
65             ig_w = igc(iWW, ig)
66             ig_s = igc(iSS, ig)
67
68             is_e = iec(iEE, ig)
69             is_n = iec(iNN, ig)
70             is_w = iec(iWW, ig)
71             is_s = iec(iSS, ig)
72
73             qme = qm2(is_e)
74             qnn = qn2(is_n)
75             qmw = qm2(is_w)
76             qns = qn2(is_s)
77
78             vneg = - hn3(ig) * dx dy ! Positive value
79             vmn = vmn + vneg ! Summation is accumulated with
80 positive value
81
82             ioe(:) = 0
83             v_out = 0.d0 ! Positive value
84             vout_e = 0.d0
85             vout_n = 0.d0
86             vout_w = 0.d0
87             vout_s = 0.d0

```

```

88         if (qme > 0.d0) then
89             ioe(iEE) = 1
90             vout_e = qme * dy * dt2
91             v_out = v_out + vout_e
92         end if
93         if (qnn > 0.d0) then
94             ioe(iNN) = 1
95             vout_n = qnn * dx * dt2
96             v_out = v_out + vout_n
97         end if
98         if (qmw < 0.d0) then
99             ioe(iWW) = 1
100            vout_w = - qmw * dy * dt2
101            v_out = v_out + vout_w
102        end if
103        if (qns < 0.d0) then
104            ioe(iSS) = 1
105            vout_s = - qns * dx * dt2
106            v_out = v_out + vout_s
107        end if
108
109        callCount = 0
110
111        if (ioe(iEE) == 1) then
112            dh_e = vout_e * vneg / (dxdy * v_out)
113            qm2(is_e) = qme - dh_e * dx / dt2
114            call get_vol(dh_e, ig_e)
115        end if
116        if (ioe(iNN) == 1) then
117            dh_n = vout_n * vneg / (dxdy * v_out)
118            qn2(is_n) = qnn - dh_n * dy / dt2
119            call get_vol(dh_n, ig_n)
120        end if
121        if (ioe(iWW) == 1) then
122            dh_w = vout_w * vneg / (dxdy * v_out)
123            qm2(is_w) = qmw + dh_w * dx / dt2
124            call get_vol(dh_w, ig_w)
125        end if
126        if (ioe(iSS) == 1) then
127            dh_s = vout_s * vneg / (dxdy * v_out)
128            qn2(is_s) = qns + dh_s * dy / dt2
129            call get_vol(dh_s, ig_s)
130        end if
131
132        hn3(ig) = 0.d0
133
134    end if

```

```

135         end do
136         !$omp end parallel do
137     end if
138
139     return
140
141     contains
142         recursive subroutine get_vol(dh, ig)
143             use arrayc
144             implicit none
145             double precision, intent(in) :: dh
146             integer, intent(in) :: ig
147
148             integer :: ig_e, ig_n, ig_w, ig_s
149             integer :: is_e, is_n, is_w, is_s
150             integer :: ioe(4)
151
152             double precision :: dxdy, total_vol
153             double precision :: qme, qnn, qmw, qns
154             double precision :: dh_e, dh_n, dh_w, dh_s
155             double precision :: vneg, v_out, vout_e, vout_n, vout_w,
vout_s
156
157             callCount = callCount + 1
158
159             if (ig == 0 .or. callCount > 10) return
160
161             ig_e = igc(iEE, ig)
162             ig_n = igc(iNN, ig)
163             ig_w = igc(iWW, ig)
164             ig_s = igc(iSS, ig)
165
166             dxdy = dx * dy
167             total_vol = dh * dxdy
168             if (hn3(ig) >= dh) then
169                 hn3(ig) = hn3(ig) - dh
170                 return
171             else
172                 total_vol = total_vol - hn3(ig) * dxdy
173
174                 is_e = iec(iEE, ig)
175                 is_n = iec(iNN, ig)
176                 is_w = iec(iWW, ig)
177                 is_s = iec(iSS, ig)
178
179                 qme = qm2(is_e)
180                 qnn = qn2(is_n)

```

```

181      qmw = qm2(is_w)
182      qns = qn2(is_s)
183
184      vneg = total_vol  ! Positive value
185      vmn = vmn - hn3(ig) * dx dy
186          ! Summation is accumulated with positive value
187
188      ioe(:) = 0
189      v_out = 0.d0 ! Positive value
190      vout_e = 0.d0
191      vout_n = 0.d0
192      vout_w = 0.d0
193      vout_s = 0.d0
194      if (qme > 0.d0) then
195          ioe(iEE) = 1
196          vout_e = qme * dy * dt2
197          v_out = v_out + vout_e
198      end if
199      if (qnn > 0.d0) then
200          ioe(iNN) = 1
201          vout_n = qnn * dx * dt2
202          v_out = v_out + vout_n
203      end if
204      if (qmw < 0.d0) then
205          ioe(iWW) = 1
206          vout_w = - qmw * dy * dt2
207          v_out = v_out + vout_w
208      end if
209      if (qns < 0.d0) then
210          ioe(iSS) = 1
211          vout_s = - qns * dx * dt2
212          v_out = v_out + vout_s
213      end if
214
215      if (ioe(iEE) == 1) then
216          dh_e = vout_e * vneg / (dx dy * v_out)
217          qm2(is_e) = qme - dh_e * dx / dt2
218          call get_vol(dh_e, ig_e)
219      end if
220      if (ioe(iNN) == 1) then
221          dh_n = vout_n * vneg / (dx dy * v_out)
222          qn2(is_n) = qnn - dh_n * dy / dt2
223          call get_vol(dh_n, ig_n)
224      end if
225      if (ioe(iWW) == 1) then
226          dh_w = vout_w * vneg / (dx dy * v_out)
227          qm2(is_w) = qmw + dh_w * dx / dt2

```

```
228             call get_vol(dh_w, ig_w)
229         end if
230         if (ioe(iSS) == 1) then
231             dh_s = vout_s * vneg / (dxdy * v_out)
232             qn2(is_s) = qns + dh_s * dy / dt2
233             call get_vol(dh_s, ig_s)
234         end if
235
236         hn3(ig) = 0.d0
237
238         return
239     end if
240 end subroutine get_vol
241 end subroutine hdepth
```

14 boundaryh.f90

boundaryh.f90

```
1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine boundaryh(m)
14     use arrayc
15     implicit none
16     integer, intent(in) :: m
17
18     integer :: i
19
20     if (nbc > 0) then
21         do i = 1, nbc
22             hn3(ibg(i)) = hin(i, m)
23         end do
24     end if
25
26     return
27
28 end subroutine boundaryh
```

15 qbflux.f90

qbflux.f90

```
1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine qbflux
14     use arrayc
15     implicit none
16     integer i, is, igE, igN, igW, igS
17     ! double precision dv, qv
18     double precision :: ccl, ccf
19
20     !$omp parallel
21     !$omp do
22     do i = 0, Nxs
23         qblx(i) = 0.d0
24     end do
25     !$omp end do
26     !$omp do
27     do i = 0, Nxs
28         qbfx(i) = 0.d0
29     end do
30     !$omp end do
31     !$omp do
32     do i = 1, Nys
33         qbly(i) = 0.d0
34     end do
35     !$omp end do
36     !$omp do
37     do i = 1, Nys
38         qbfy(i) = 0.d0
39     end do
40     !$omp end do
41
42     !$omp do private(i, is, igE, igW, ccl, ccf)
```

```

43  do i = 1, Nwxs
44      is = idxWxs(i)
45      igE = ixsgc(iEg, is)
46      igW = ixsgc(iWg, is)
47
48      ccl = (c11(igW) + c11(igE)) * 0.5d0
49      ccf = (cf1(igW) + cf1(igE)) * 0.5d0
50
51      qblx(is) = qm2(is) * ccl
52      qbfx(is) = qm2(is) * (1.0d0 - ccl) * ccf
53      if (-eps < qbfx(is) .and. qbfx(is) < eps) qbfx(is) = 0.d0
54  end do
55  !$omp end do
56
57  !$omp do private(i, is, igN, igS, ccl, ccf)
58  do i = 1, Nwys
59      is = idxWys(i)
60      igN = iysgc(iNg, is)
61      igS = iysgc(iSg, is)
62
63      ccl = (c11(igN) + c11(igS)) * 0.5d0
64      ccf = (cf1(igN) + cf1(igS)) * 0.5d0
65
66      qbly(is) = qn2(is) * ccl
67      qbfy(is) = qn2(is) * (1.0d0 - ccl) * ccf
68      if (-eps < qbfy(is) .and. qbfy(is) < eps) qbfy(is) = 0.d0
69  end do
70  !$omp end do
71  !$omp end parallel
72
73  return
74 end subroutine qbflux

```

16 boundaryqb.f90

boundaryqb.f90

```
1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine boundaryqb(m)
14     use arrayc
15     implicit none
16     integer i, ix, iy, ixg, iyg
17     integer, intent(in) :: m
18
19     ! Computation of overflow discharge on the edge of the computational domain
20     if (nxout > 0) then
21         do i = 1, nxout
22             ix = ixout(i)
23             ixg = abs(ixgout(i))
24
25             qblx(ix) = qm2(ix) * c11(ixg)
26             qbfx(ix) = qm2(ix) * cf1(ixg) * (1.d0 - c11(ixg))
27         end do
28     end if
29     if (nyout > 0) then
30         do i = 1, nyout
31             iy = iyout(i)
32             iyg = abs(iygout(i))
33
34             qblx(iy) = qm2(iy) * c11(iyg)
35             qbfx(iy) = qm2(iy) * cf1(iyg) * (1.d0 - c11(iyg))
36         end do
37     end if
38
39     ! x-direction inflow discharge
40     if (nxbc > 0) then
41         do i = 1, nxbc
42             ! coarse
```

```

43         qblx(iec(      :, ixbg(i))) = 0.d0
44         qblx(iec(ixbd(i), ixbg(i))) = qblxin(i, m)
45         ! fine
46         qbfx(iec(      :, ixbg(i))) = 0.d0
47         qbfx(iec(ixbd(i), ixbg(i))) = qbfxin(i, m)
48     end do
49 end if
50
51 ! y-direction inflow discharge
52 if (nybc > 0) then
53     do i = 1, nybc
54         ! coarse
55         qbly(iec(      :, iybg(i))) = 0.d0
56         qbly(iec(iybd(i), iybg(i))) = qblyin(i, m)
57         ! fine
58         qbfy(iec(      :, iybg(i))) = 0.d0
59         qbfy(iec(iybd(i), iybg(i))) = qbfyin(i, m)
60     end do
61 end if
62
63 ! x-direction wall boundary
64 if (nxwall > 0) then
65     do i = 1, nxwall
66         qblx(ixwall(i)) = 0.d0
67         qbfx(ixwall(i)) = 0.d0
68     end do
69 end if
70
71 ! y-direction wall boundary
72 if (nywall > 0) then
73     do i = 1, nywall
74         qbly(iywall(i)) = 0.d0
75         qbfy(iywall(i)) = 0.d0
76     end do
77 end if
78
79 return
80
81 end subroutine boundaryqb

```

17 gradws.f90

gradws.f90

```

1  !
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !
13 subroutine gradws
14     use arrayc
15     implicit none
16     integer :: i, ig, ig_e, ig_n, ig_w, ig_s
17     double precision :: dzdx, dzdy
18     double precision :: tanw
19
20     !$omp parallel
21
22     !$omp do private(i)
23     do i = 0, Ngrid
24         sei(i) = 0.d0
25     end do
26     !$omp end do
27
28     !$omp do private(i, ig, ig_e, ig_n, ig_w, ig_s, dzdx, dzdy, tanw)
29     do i = 1, Nwg
30         ig = idxWg(i)
31
32         ig_e = igc(iEE, ig)
33         ig_n = igc(iNN, ig)
34         ig_w = igc(iWW, ig)
35         ig_s = igc(iSS, ig)
36
37         dzdx = ((z11(ig_e) + hn1(ig_e)) - (z11(ig_w)+hn1(ig_w)))/(2.d0*dx)
38         dzdy = ((z11(ig_n) + hn1(ig_n)) - (z11(ig_s)+hn1(ig_s)))/(2.d0*dy)
39
40         tanw = dzdx*dzdx + dzdy*dzdy
41
42         sei(ig) = sqrt(tanw)

```

```
43
44   end do
45   !$omp end do
46
47   !$omp end parallel
48
49   return
50
51 end subroutine gradws
```

18 gradbed.f90

gradbed.f90

```

1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine gradbed
14     use arrayc
15     implicit none
16     integer :: i, ig, ig_e, ig_n, ig_w, ig_s
17     double precision :: dzdx, dzdy
18     double precision :: tanb
19
20
21     !$omp parallel
22
23     !$omp do private(i)
24     do i = 0, Ngrid
25         sei(i) = 0.d0
26     end do
27     !$omp end do
28
29     !$omp do private(i, ig, ig_e, ig_n, ig_w, ig_s, dzdx, dzdy, tanb)
30     do i = 1, Nwg
31         ig = idxWg(i)
32
33         ig_e = igc(iEE, ig)
34         ig_n = igc(iNN, ig)
35         ig_w = igc(iWW, ig)
36         ig_s = igc(iSS, ig)
37
38         dzdx = (z11(ig_e) - z11(ig_w))/(2.d0*dx)
39         dzdy = (z11(ig_n) - z11(ig_s))/(2.d0*dy)
40
41         tanb2 = dzdx*dzdx + dzdy*dzdy
42         sei(ig) = sqrt(tanb)

```

```
43
44   end do
45   !$omp end do
46
47   !$omp end parallel
48
49   return
50
51 end subroutine gradbed
```

19 cinfty.f90

cinfty.f90

```
1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine cinfty
14     use arrayc
15     implicit none
16     integer :: i, ig
17     double precision :: i_e, ts, us
18     double precision :: dls, dlc, alp1, alp2, alp, a0
19
20     !$omp parallel
21
22     !$omp do private(i)
23     do i = 0, Ngrid
24         clp(i) = 0.d0
25     end do
26     !$omp end do
27
28     !!!!!! Equilibrium concentration !!!!!!!
29     !$omp do private(i, ig, i_e, wsg, dls, dlc, us, ts, alp1, alp2, alp, a0)
30     do i = 1, Nwg
31         ig = idxWg(i)
32
33         i_e = sei(ig)
34
35         wsg = (sig - rof(ig)) / rof(ig)
36         dls = tanp - i_e
37         dlc = i_e / (wsg * c90)
38
39         if(i_e < 0.030d0) then
40             ! Bedload transport
41             us = sqrt(gr * hn3(ig) * i_e)
42             ts = us * us / (wsg * gr * dm)
```

```

43     if(ts > tauc) then
44         alp1 = 2.d0 * (0.425d0 - (wsg + 1.0)*i_e / wsg)
45         if(alp1 <= 0.d0) alp1 = 0.d0
46         alp2 = 1.d0 - (wsg + 1.0)*i_e / wsg
47         alp = 0.d0
48         if(alp2 > 0.d0) alp = alp1 / alp2
49         clp(ig) = (1.d0 + 5.d0 * i_e) * i_e / wsg &
50                 & * (1.d0 - alp * alp * tauc / ts) &
51                 & * (1.d0 - alp * sqrt(tauc / ts))
52     end if
53
54     else if(i_e < 0.138d0) then
55         ! immature debris flow
56         a0 = i_e / (wsg * dls)
57         clp(ig) = 6.7d0 * a0 * a0
58
59     else if(i_e < tanp) then
60         ! Debris flow
61         clp(ig) = sei(ig) / (wsg * dls)
62
63         if(clp(ig) > c90) clp(ig) = c90
64
65     end if
66
67     if(clp(ig) < 0.d0) clp(ig) = 0.d0
68
69     end do
70     !$omp end do
71
72     !$omp end parallel
73
74     return
75
76 end subroutine cinfy

```

20 calclcf.f90

calclcf.f90

```

1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine calclcf
14     use arrayc
15     implicit none
16     integer :: i, ig, i_e, i_n, i_w, i_s
17     double precision :: hh, vv11, vv13, vvf1, vvf3
18     double precision :: dqbx, dqby, ai
19     double precision :: v_err, dz, dh_err
20
21     cfmax = 0.d0
22     clmax = 0.d0
23
24     !$omp parallel
25     !$omp do
26     do i = 1, Ngrid
27         cl3(i) = 0.d0
28     end do
29     !$omp end do
30     !$omp do
31     do i = 1, Ngrid
32         cf3(i) = 0.d0
33     end do
34     !$omp end do
35
36     !$omp do private(i, ig, hh, i_e, i_n, i_w, i_s, dqbx, dqby, ai, vv11, vv13,
37     vvf1, vvf3, v_err, dz, dh_err)
38     do i = 1, Nwg
39         ig = idxWg(i)
40         hh = hn3(ig)
41         if (hh < th) cycle

```

```

42     i_e = iec(iEE, ig)
43     i_n = iec(iNN, ig)
44     i_w = iec(iWW, ig)
45     i_s = iec(iSS, ig)
46
47     !— Coarse sediments
48     !— eq. of continuity
49     dqbx = qblx(i_e) - qblx(i_w)
50     dqby = qbly(i_n) - qbly(i_s)
51     if (aai(ig) >= 0.d0) then
52         ! erosion
53     !     a3 = aii(ig) * czl(ig)
54         ai = aii(ig) * cstl
55     else
56         ! deposition
57         ai = aii(ig) * cdl
58     end if
59     vvl1 = hn1(ig) * cl1(ig)
60     vvl3 = vvl1 - (dqbx / dx + dqby / dy)*dt2 + ai*dt2
61     cl3(ig) = vvl3 / hh
62     !
63     !— error removal process
64     if(cl3(ig) < eps) then
65         v_err = vvl3 - (0.d0 - cl3(ig)) * hh
66         cl3(ig) = 0.d0
67     end if
68     if (cl3(ig) > c90) then
69         !— flg_rebed is sat in param.dat
70         if (flg_rebed == 1) then
71             v_err = vvl3 - (cl3(ig) - c90) * hh
72             dh_err = v_err/cl3(ig)
73             dz = dh_err*(cl3(ig) - clp(ig))/(c90 - clp(ig))
74             ! dh_real = dh_err*(cstl - cl3(ig))/(cstl - clp(ig))
75             if (.not. dz - 1 == dz) then ! infの判定
76                 ! infinityでない場合、河床で調整
77                 ! print *, "calclcf:: dz is not Inf", dz
78                 if (dz > th) zll(ig) = zll(ig) + dz
79             end if
80         end if
81         cl3(ig) = c90
82     end if
83
84     !— Fine sediments
85     ! eq. of continuity
86     dqbx = qbfx(i_e) - qbfx(i_w)
87     dqby = qbfx(i_n) - qbfx(i_s)
88

```

```

89     if(aii(ig) >= 0.d0) then
90         ! erosion
91         ai = aii(ig) * (1.d0 - cst1) * cstf
92     else
93         ! deposition
94         ai = aii(ig) * (1.d0 - cd1) * cf1(ig)
95     endif
96
97     vvf1 = hn1(ig) * (1.d0 - cl1(ig)) * cf1(ig)
98     vvf3 = vvf1 - (dqbx / dx + dqby / dy) * dt2 + ai*dt2
99     cf3(ig) = vvf3 / (hh * (1.d0 - cl1(ig)))
100    !
101    !— error removal process
102    !— 細粒分土砂濃度に上限 cstrf を設ける。
103    if(cf3(ig) < 0.d0) then
104        v_err = vvf3 - (0.d0 - cf3(ig)) * (hh * (1.d0 - cl1(ig)))
105        cf3(ig) = 0.d0
106    end if
107    if (cf3(ig) > cstf) then
108        v_err = vvf3 - (cf3(ig) - cstf) * (hh * (1.d0 - cl3(ig)))
109        cf3(ig) = cstf
110    end if
111
112    end do
113    !$omp end do
114    !$omp end parallel
115
116    return
117 end subroutine calcclcf

```

21 calrho.f90

calrho.f90

```
1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine calrho
14     use arrayc
15     implicit none
16     integer :: i, ig
17
18     !$omp parallel
19     !$omp do
20     do i = 1, Ngrid
21         rot(i) = 0.d0
22     end do
23     !$omp end do
24     !$omp do
25     do i = 1, Ngrid
26         rof(i) = 0.d0
27     end do
28     !$omp end do
29     !$omp do private(i, ig)
30     do i = 1, Nwg
31         ig = idxWg(i)
32         if (hn3(ig) < th) cycle
33
34         ! update density
35         rof(ig) = cf3(ig) * sig + (1.d0 - cf3(ig)) * rho
36         rot(ig) = cl3(ig) * sig + (1.d0 - cl3(ig)) * rof(ig)
37     end do
38     !$omp end do
39     !$omp end parallel
40     return
41 end subroutine calrho
```

22 calbedelev.f90

calbedelev.f90

```
1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine calbedelev
14     use arrayc
15     implicit none
16     integer :: i, ig
17     ! double precision :: zzz
18
19     !$omp parallel
20
21
22     !— river bed eq. of mixture
23
24     !$omp do private(i, ig)
25     do i = 1, Nwg
26
27         ig = idxWg(i)
28
29         z13(ig) = z11(ig) - aii(ig) * dt2
30
31     end do
32     !$omp end do
33
34
35
36     !$omp do private(i)
37     do i = 0, Ngrid
38
39         zdz(i) = z13(i) - ezi(i)
40
41     end do
42     !$omp end do
```

```

43
44      ! 河床堆積層の連続式
45      ! 河床堆積層の体積の定義の修正の必要がある。
46      ! 改良しないと河床堆積層の土砂濃度はうまく計算できないが
47      ! 参考に残しておく
48
49
50      !  !$omp do private(i)
51      !do i = 0, Ngrid
52      !  czl(i) = 0.d0
53      !end do
54      !  !$omp end do
55      !  !$omp do private(i)
56      !do i = 0, Ngrid
57      !  czf(i) = 0.d0
58      !end do
59      !  !$omp end do
60
61
62      !— river bed eq. of
63      !—   coarse and fine sediment
64
65      ! tvl = 0.d0
66      ! tvf = 0.d0
67
68      !  !$omp do private(i, ig, zzz), reduction(+: tvl, tvf)
69      ! do i = 1, Nwg
70
71      !   ig = idxWg(i)
72
73
74      !   zzz = zl3(ig) - zdp(ig)
75      !   zzz = zl3(ig) - zl1(ig)
76
77
78      !   if(aii(ig) >= 0.d0) then
79      !       zvl(ig) = zvl(ig) - aii(ig) * dt2 * czl(ig)
80      !       zvf(ig) = zvf(ig) - aii(ig) * dt2 * (1.d0 - czl(ig)) * czf(ig)
81      !   else
82      !       zvl(ig) = zvl(ig) - aii(ig) * dt2 * cdl
83      !       zvf(ig) = zvf(ig) - aii(ig) * dt2 * (1.d0 - cdl) * cf3(ig)
84      !   end if
85
86      !   zzz = zl3(ig) - zdp(ig)
87
88      !   if(zzz > eps) then
89      !       !— coarse

```

```

90      !      czl(ig) = zvl(ig) / zzz
91      !      ! error removal process
92      !      ! it is necessary to revise this process
93      !      ! based on satisfying conservation law
94      !      if (czl(ig) > cdl * 1.01d0) then
95      !          zvl(ig) = zvl(ig) - zzz * (czl(ig) - cdl)
96      !          czl(ig) = cdl
97      !      end if
98      !      if (czl(ig) < 0.d0) then
99      !          zvl(ig) = zvl(ig) - zzz * (0.d0 - czl(ig))
100     !          czl(ig) = 0.d0
101     !      end if
102     !      !
103     !      !-- fine
104     !      czf(ig) = zvf(ig) / (zzz * (1.d0 - czl(ig)))
105     !      if (czf(ig) > cstf * 1.01d0) then
106     !          zvf(ig) = zvf(ig) - zzz * (1.0 - czl(ig)) * (czf(ig) - cstf)
107     !          czf(ig) = cstf
108     !      end if
109     !      if (czf(ig) < 0.d0) then
110     !          zvf(ig) = zvf(ig) - zzz * (1.0 - czl(ig)) * (0.d0 - czf(ig))
111     !          czf(ig) = 0.d0
112     !      end if
113     !  end if
114
115     !  tvl = tvl + (zvl(ig) + hn3(ig) * cl3(ig)) * dx * dy
116     !  tvf = tvf + (zvf(ig) + hn3(ig) * (1.d0 - cl3(ig)) * cf3(ig)) * dx * dy
117
118
119     !  if (a11(ig) >= 0.0) then
120     !      ! erosion
121     !      tvl = tvl + zzz * cstl
122     !      tvf = tvf + zzz * cstf
123     !  else if (a11(ig) < 0.0) then
124     !      ! deposition
125     !      tvl = tvl + zzz * cdl
126     !      tvf = tvf + zzz * (cstl + (1.d0 - cstl)*cf3(ig) )
127     !  end if
128     ! end do
129 !  !$omp end do
130
131
132     !$omp end parallel
133
134     ! 河床堆積層の体積の計算間違いのチェック
135     !  if(minval(zvl) < 0.d0) then
136     !  print *, "minval(zlv)", minval(zvl)

```

```
137      ! call errnum(' aiical-l0 ')
138      ! end if
139      ! if(minval(zvf) < 0.d0) then
140      !   print *, "minval(zvf)", minval(zvf)
141      !   call errnum(' aiical-f0 ')
142      ! end if
143
144      return
145
146 end subroutine calbedelev
```

23 calaii.f90

calaii.f90

```
1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine calaii
14     use arrayc
15     implicit none
16     integer :: i, ig
17     double precision :: cl, dzmax
18     double precision :: aiimax
19
20
21     !$omp parallel
22
23     !$omp do private(i)
24     do i = 0, Ngrid
25         aii(i) = 0.d0
26     end do
27     !$omp end do
28
29     !$omp do private(i, ig, cl, dzmax, aiimax)
30     do i = 1, Nwg
31         ig = idxWg(i)
32
33         if (sei(ig) > tanp) cycle
34         ! 勾配が tan phi 以上では理想的には土石流が発生しない
35         ! 平衡土砂濃度が不合理となるため、設定しない
36
37         cl = cl3(ig)
38
39         ! aiimax = (zl3(ig) - zdp(ig))/dt2
40         dzmax = zl3(ig) - MAX_ELEVCHANGE.IN.1STEP
41         aiimax = dzmax/dt2
42
```

```

43     if(cl <= clp(ig)) then
44         ! erosion aii > 0
45         aii(ig) = bit * (clp(ig) - cl) * uva(ig) / (cst - clp(ig))
46         ! error elimination process
47         !— partly referenced to 山崎ほか(2022)
48         aii(ig) = min(aii(ig), aimax)
49     else
50         ! deposition aii < 0
51         aii(ig) = bit_d * (clp(ig) - cl) * uva(ig) / cdl
52     end if
53
54 end do
55 !$end do
56 !$omp end parallel
57
58 return
59
60 end subroutine calaii

```

24 velcal.f90

velcal.f90

```
1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine velcal(m)
14     use arrayc
15     implicit none
16
17     integer, intent(in) :: m
18
19     integer :: i, is, ig_e, ig_n, ig_w, ig_s
20     integer :: ig, is_e, is_n, is_w, is_s
21     double precision :: hh, tmp
22
23
24     !$omp parallel
25
26     !$omp do private(i)
27     do i = 0, Nxs
28         uul(i) = 0.d0
29     end do
30     !$omp end do
31     !$omp do private(i)
32     do i = 0, Nys
33         vvl(i) = 0.d0
34     end do
35     !$omp end do
36     !$omp do private(i)
37     do i = 0, Ngrid
38         uux(i) = 0.d0
39     end do
40     !$omp end do
41     !$omp do private(i)
42     do i = 0, Ngrid
```

```

43     vvy(i) = 0.d0
44 end do
45 !$omp end do
46
47 !$omp do private(i, is, ig_e, ig_w, is_e, is_w, hh)
48 do i = 1, Nwxs
49     is = idxWxs(i)
50
51     ig_e = ixsgc(iEg, is)
52     ig_w = ixsgc(iWg, is)
53     is_e = ixsrc(iEE, is)
54     is_w = ixsrc(iWW, is)
55
56     hh = hn3(ig_e) + hn3(ig_w) + hn1(ig_e) + hn1(ig_w)
57     if (0.25d0 * hh > th) then
58         tmp = 2.d0 * (qm2(is) + qm1(is)) / hh
59         uul(is) = 2.d0 * (qm2(is) + qm1(is)) / hh
60     end if
61
62 end do
63 !$omp end do
64 !$omp do private(i, is, ig_n, ig_s, is_n, is_s, hh)
65 do i = 1, Nwys
66     is = idxWys(i)
67
68     ig_n = iysgc(iNg, is)
69     ig_s = iysgc(iSg, is)
70     is_n = iysyc(iNN, is)
71     is_s = iysyc(iSS, is)
72
73     hh = hn3(ig_n) + hn3(ig_s) + hn1(ig_n) + hn1(ig_s)
74     if (0.25d0 * hh > th) then
75         vvl(is) = 2.d0 * (qn2(is) + qn1(is)) / hh
76     end if
77
78 end do
79 !$omp end do
80
81 !$omp end parallel
82
83
84 if (nxbc > 0) then
85     do i = 1, nxbc
86         is = iec(ixbd(i), ixbg(i))
87         uul(is) = uin(i, m)
88         uux(ixbg(i)) = uin(i, m)
89     end do

```

```

90     end if
91     if (nybc > 0) then
92         do i = 1, nybc
93             is = iec(iybd(i), iybg(i))
94             vvl(is) = vin(i, m)
95             vvy(iybg(i)) = vin(i, m)
96         end do
97     end if
98
99     ! Velocity never go throught walls
100    if (nxwall > 0) then
101        do i = 1, nxwall
102            uul(ixwall(i)) = 0.d0
103        end do
104    end if
105    if (nywall > 0) then
106        do i = 1, nywall
107            vvl(iywall(i)) = 0.d0
108        end do
109    end if
110
111    !$omp parallel do private(i, ig, is_e, is_n, is_w, is_s)
112    do i = 1, Nwg
113        ig = idxWg(i)
114
115        is_e = iec(iEE, ig)
116        is_n = iec(iNN, ig)
117        is_w = iec(iWW, ig)
118        is_s = iec(iSS, ig)
119
120        uux(ig) = 0.5d0 * (uul(is_e) + uul(is_w))
121        vvy(ig) = 0.5d0 * (vvl(is_n) + vvl(is_s))
122        uva(ig) = sqrt(uux(ig) * uux(ig) + vvy(ig) * vvy(ig))
123    end do
124    !$omp end parallel do
125
126    return
127
128 end subroutine velcal

```

25 flowpower.f90

flowpower.f90

```
1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine flowpower
14     use arrayc
15     implicit none
16     integer :: i
17
18     !$omp parallel
19
20     !$omp do private(i)
21     do i = 1, Ngrid
22         ftx(i) = rot(i) * hn3(i) * uux(i) * uux(i)
23     end do
24     !$omp end do
25     !$omp do private(i)
26     do i = 1, Ngrid
27         fty(i) = rot(i) * hn3(i) * vvy(i) * vvy(i)
28     end do
29     !$omp end do
30     !$omp do private(i)
31     do i = 1, Ngrid
32         fta(i) = rot(i) * hn3(i) * uva(i) * uva(i)
33     end do
34     !$omp end do
35
36     !$omp end parallel
37
38     return
39
40 end subroutine flowpower
```

26 pickcal.f90

pickcal.f90

```

1  !
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !


---


13 subroutine pickcal
14     use arrayc
15     implicit none
16     integer :: i, ig
17
18     double precision, allocatable :: ucfl(:), vcfl(:)
19
20     allocate(ucfl(0:Ngrid), vcfl(0:Ngrid))
21
22     !$omp parallel
23
24     !$omp do private(i, ig)
25     do i = 1, Nwg
26         ig = idxWg(i)
27
28         qmx(ig) = (qm2(iec(iEE, ig)) + qm2(iec(iWW, ig))) * 0.5d0
29         qny(ig) = (qn2(iec(iNN, ig)) + qn2(iec(iSS, ig))) * 0.5d0
30         qqa(ig) = sqrt(qmx(ig) * qmx(ig) + qny(ig) * qny(ig))
31     end do
32     !$omp end do
33
34     !$omp do private(i, ig)
35     do i = 1, Nwg
36         ig = idxWg(i)
37
38         qblxm(ig) = (qblx(iec(iEE, ig)) + qblx(iec(iWW, ig))) * 0.5d0
39         qblym(ig) = (qbly(iec(iNN, ig)) + qbly(iec(iSS, ig))) * 0.5d0
40         qbla(ig) = sqrt(qblxm(ig) * qblxm(ig) + qblym(ig) * qblym(ig))
41         qbfxm(ig) = (qbfx(iec(iEE, ig)) + qbfx(iec(iWW, ig))) * 0.5d0
42         qbfym(ig) = (qbfy(iec(iNN, ig)) + qbfy(iec(iSS, ig))) * 0.5d0

```

```

43     qbfa(ig) = sqrt(qbfxm(ig) * qbfxm(ig) + qbfym(ig) * qbfym(ig))
44 end do
45 !$omp end do
46
47 !$omp do private(i, ig)
48 do i = 1, Nwg
49     ig = idxWg(i)
50     if (arvt(ig) <= 0.d0 .and. hn3(ig) > th) then
51         arvt(ig) = timex
52     end if
53 end do
54 !$omp end do
55
56 !$omp do private(i, ig)
57 do i = 1, Nwg
58     ig = idxWg(i)
59     if (hmax(ig) < hn3(ig)) then
60         thmax(ig) = timex
61         hmax(ig) = hn3(ig)
62     end if
63 end do
64 !$omp end do
65
66 !$omp do private(i, ig)
67 do i = 1, Nwg
68     ig = idxWg(i)
69     if (zmax(ig) < zdz(ig)) then
70         tzmax(ig) = timex
71         zmax(ig) = zdz(ig)
72     end if
73 end do
74 !$omp end do
75
76 !$omp do private(i, ig)
77 do i = 1, Nwg
78     ig = idxWg(i)
79     if (zmin(ig) > zdz(ig)) then
80         tzmin(ig) = timex
81         zmin(ig) = zdz(ig)
82     end if
83 end do
84 !$omp end do
85
86 !$omp do private(i)
87 do i = 1, Ngrid
88     hz(i) = hn3(i)
89 end do

```



```

90     !$omp end do
91
92     !$omp do private(i, ig)
93     do i = 1, Nwg
94         ig = idxWg(i)
95         if (zdz(ig) > 0.d0) then
96             hz(ig) = hz(ig) + zdz(ig)
97         end if
98     end do
99     !$omp end do
100
101     !$omp do private(i, ig)
102     do i = 1, Nwg
103         ig = idxWg(i)
104         if (hzmax(ig) < hz(ig)) then
105             thzmax(ig) = timex
106             hzmax(ig) = hz(ig)
107             hahz(ig) = hn3(ig)
108             zahz(ig) = zdz(ig)
109         end if
110     end do
111     !$omp end do
112
113     !$omp do private(i, ig)
114     do i = 1, Nwg
115         ig = idxWg(i)
116         if (uvmax(ig) < uva(ig)) then
117             tvmax(ig) = timex
118             uvmax(ig) = uva(ig)
119             vxm(ig) = uux(ig)
120             vym(ig) = vvy(ig)
121         end if
122     end do
123     !$omp end do
124
125     !$omp do private(i, ig)
126     do i = 1, Nwg
127         ig = idxWg(i)
128         if (qqmax(ig) < qqa(ig)) then
129             tqmax(ig) = timex
130             qqmax(ig) = qqa(ig)
131             qxmax(ig) = qmx(ig)
132             qymax(ig) = qny(ig)
133         end if
134     end do
135     !$omp end do
136

```

```

137  !$omp do private(i, ig)
138  do i = 1, Nwg
139      ig = idxWg(i)
140      if (fmax(ig) < fta(ig)) then
141          tfmax(ig) = timex
142          fmax(ig) = fta(ig)
143          fmx(ig) = ftx(ig)
144          fmy(ig) = fty(ig)
145          hafmax(ig) = hn3(ig)
146          zafmax(ig) = zdz(ig)
147      end if
148  end do
149  !$omp end do
150
151
152  !$omp do private(i)
153  do i = 1, Ngrid
154      ucfl(i) = 0.d0
155  end do
156  !$omp end do
157  !$omp do private(i)
158  do i = 1, Ngrid
159      vcfl(i) = 0.d0
160  end do
161  !$omp end do
162
163  !$omp do private(i, ig)
164  do i = 1, Nwg
165      ig = idxWg(i)
166      ucfl(ig) = abs(uux(ig) + sqrt(gr * hn3(ig))) * dt2 / dx
167  end do
168  !$omp end do
169  !$omp do private(i, ig)
170  do i = 1, Nwg
171      ig = idxWg(i)
172      vcfl(ig) = abs(vvy(ig) + sqrt(gr * hn3(ig))) * dt2 / dy
173  end do
174  !$omp end do
175
176  !$omp end parallel
177
178  cfl = max(maxval(ucfl), maxval(vcfl))
179
180
181  deallocate(ucfl, vcfl)
182
183

```

```
184     return
185
186 end subroutine pickcal
```

27 tractive.f90

tractive.f90

```

1  !
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !
13 subroutine tractive
14   use arrayc
15   implicit none
16   integer :: i, ig
17   double precision :: hh, si, usj
18
19   !$omp parallel
20
21   !$omp do
22   do i = 1, Ngrid
23     taus(i) = 0.d0
24   end do
25   !$omp end do
26
27   !$omp do private(i, ig, hh, si, usj)
28   do i = 1, nWg
29     ig = idxWg(i)
30     hh = hn3(ig)
31
32     if (hh < th) cycle
33
34     if (noSediment .eqv. .false.) then
35       wsg = (sig - rof(ig)) / rof(ig)
36     end if
37
38     si = smn(ig) * smn(ig) * uva(ig) * uva(ig) / hn3(ig)**r43
39     usj = gr * hn3(ig) * si
40     taus(ig) = usj / (gr * wsg * dm)
41   end do
42   !$omp end do

```

```
43
44   !$omp do private(i, ig)
45   do i = 1, nWg
46     ig = idxWg(i)
47     if (tapc(ig) < taus(ig)) then
48       tapc(ig) = taus(ig)
49     end if
50   end do
51   !$omp end do
52
53   !$omp end parallel
54
55   return
56
57 end subroutine tractive
```

28 replace.f90

replace.f90

```
1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine replace
14     use arrayc
15     implicit none
16
17     integer :: i
18
19     !$omp parallel
20
21     !$omp do
22     do i = 0, Ngrid
23         hn0(i) = hn1(i)
24     end do
25     !$omp end do
26     !$omp do
27     do i = 0, Ngrid
28         hn1(i) = hn3(i)
29     end do
30     !$omp end do
31     !$omp do
32     do i = 0, Nxs
33         qm0(i) = qm1(i)
34     end do
35     !$omp end do
36     !$omp do
37     do i = 0, Nxs
38         qm1(i) = qm2(i)
39     end do
40     !$omp end do
41     !$omp do
42     do i = 0, Nys
```

```

43     qn0(i) = qn1(i)
44 end do
45 !$omp end do
46 !$omp do
47 do i = 0, Nys
48     qn1(i) = qn2(i)
49 end do
50 !$omp end do
51
52 !$omp end parallel
53
54 if(noSediment) return
55
56 !$omp parallel
57
58 !$omp do
59 do i = 0, Ngrid
60     z11(i) = z13(i)
61 end do
62 !$omp end do
63 !$omp do
64 do i = 0, Ngrid
65     cf1(i) = cf3(i)
66 end do
67 !$omp end do
68 !$omp do
69 do i = 0, Ngrid
70     cl1(i) = cl3(i)
71 end do
72 !$omp end do
73 !$omp do
74 do i = 0, Ngrid
75     ct1(i) = ct3(i)
76 end do
77 !$omp end do
78
79 !$omp end parallel
80
81 return
82
83 end subroutine replace

```

29 snapshot.f90

snapshot.f90

```

1  !-----
2  !
3  ! Copyright 2022
4  !   Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  !   Public Works Research Institute (PWRI)
6  !
7  ! License
8  !   Creative Common (CC) BY-SA 4.0
9  !   (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 !   (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine snapshot(ite)
14     use arrayc
15     implicit none
16
17     integer, intent(in) :: ite
18
19     integer :: usnp, i
20     character(255) :: fname
21     character(4)   :: ext   = ".csv"
22     character(5)   :: num
23     character(255) :: fm
24
25     if (ite >= 0) then
26         write(num, '(i0.5)') ite
27         fname = fsnap(1:len(trim(adjustl(fsnap)))-4) &
28             & // '_ ' // num // ext
29         open(newunit = usnp, file = fname, status = 'unknown')
30     else
31         usnp = 32
32     end if
33
34
35     fm = '#,X,Y,longitude,latitude,bed_elevation,water_elevation,&
36     &water_depth,U,V,qx,qy,WetDry,&
37     &tan_theta,dz,qblx,qbly,qbfx,qbfy,&
38     &cl,cf,erosion_rate,cinfty)'
39     write(usnp, fm)
40     fm = '(i0,u",",u&
41     &uug0,u",",ug0,u",",ug0,u",",ug0,u",",ug0,u",",ug0,u",",u&
42     &uug0,u",",ug0,u",",ug0,u",",ug0,u",",ug0,u",",ui0,u",",u&

```



```

43  &ug0,"","ug0,"","ug0,"","ug0,"","ug0,"","ug0,"","ug0,"","ug0,""&
44  &ug0,"","ug0,"","ug0,"","ug0)"'
45  write(usnp, fm) (i, &
46      & xy(1,i),xy(2,i),lonlat(1,i),lonlat(2,i),z13(i),z13(i)+hn3(i),&
47      & hn3(i), uux(i), vvy(i), qmx(i), qny(i), WDg(i), &
48      & sei(i), zdz(i), qblxm(i), qblym(i), qbfxm(i), qbfym(i),&
49      & cl3(i), cf3(i), aii(i), clp(i), i = 1, Ngrid)
50
51  close(usnp)
52
53  return
54  end subroutine snapshot

```

30 outmaxvalue.f90

outmaxvalue.f90

```

1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine outmaxvalue()
14     use arrayc
15     implicit none
16     integer :: i, a0
17     character(1024) :: fm
18
19     fm = '#,X,Y,longitude,latitude,&
20     uuuuuu&init._bed_elv.,max_water_depth,max_water_depth_time,&
21     uuuuuu&max_deposition_bed_elv._change,max_deposition_bed_elv._time,&
22     uuuuuu&max_erosion_bed_elv._change,max_erosion_bed_elv._time,&
23     uuuuuu&max_flow_depth,_max_flow_depth_time,&
24     uuuuuu&water_depth_at_max_flow_depth,elv._change_at_max_flow_depth,&
25     uuuuuu&max_vel.,max_vel._time,max_vel.(x),max_vel.(y),&
26     uuuuuu&max_disch.,max_disch._time,max_disch.(x),max_disch.(y),&
27     uuuuuu&max_f.power,max_f.power_time,_max_f.power(x),max_f.power(y),&
28     uuuuuu&water_depth_at_max_f.power,elv.change_at_max_f.power,&
29     uuuuuu&arival_time,max_trac.f.,sed._mov._flag)
30
31     write(33, fm)
32     fm = '(i0,u",",u&
33     uuuuuu&ug0,u",",ug0,u",",ug0,u",",ug0,u",",ug0,u",",ug0,u",",u&
34     uuuuuu&ug0,u",",ug0,u",",ug0,u",",ug0,u",",ug0,u",",ug0,u",",u&
35     uuuuuu&ug0,u",",ug0,u",",ug0,u",",ug0,u",",ug0,u",",ug0,u",",u&
36     uuuuuu&ug0,u",",ug0,u",",ug0,u",",ug0,u",",ug0,u",",ug0,u",",u&
37     uuuuuu&ug0,u",",ug0,u",",ug0,u",",ug0,u",",ug0,u",",uio)'
38
39     do i = 1, Ngrid
40         a0 = 0
41         if(tauc(i) > tauc) a0 = 1
42         write(33, fm) i, xy(1, i), xy(2, i), lonlat(1, i), lonlat(2, i), &

```

```
43      &ezi(i), hmax(i), thmax(i), &
44      &zmax(i), tzmax(i), &
45      &zmin(i), tzmin(i), &
46      &hzmax(i), thzmax(i), &
47      &hahz(i), zahz(i), &
48      &uvmax(i), tvmax(i), vxm(i), vym(i), &
49      &qqmax(i), tqmax(i), qxmax(i), qymax(i), &
50      &fmax(i), tfmax(i), fmx(i), fmy(i), &
51      &hafmax(i), zafmax(i), &
52      &arvt(i), tapc(i), a0
53  end do
54
55  return
56 end subroutine outmaxvalue
```

31 tau.f90

tau.f90

```

1  !
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !
13 subroutine tau
14     use arrayc
15     implicit none
16     integer :: i, j, itau
17     integer :: ig_e, ig_n, ig_w, ig_s
18     integer :: ig_ne, ig_nw, ig_sw, ig_se
19     integer :: nCallCount
20     integer :: ib, next, nQueue
21     integer :: id, ig(8)
22     integer, allocatable :: Queue(:), QueueStack(:)
23     ! avoid stack overflow to change allocatable
24     ! integer, dimension(im,jm,7):: meshflag
25
26     ! Algorithm of this code is as follows:
27     ! 1. check the cells which contains tractive force over critical
28     !    friction force of target diameter, and checge value is 1
29     ! 2. connect cells betewwn flag cells in which distance is one,
30     !    the cell value is changes to 2, if 1 not good.
31     ! 3. change all of flag values into 1
32     ! 4. flag value of input src point is changed into 2
33     ! 5. connect cells from src points is changed into flag value 2
34
35     ! !!!!!!!!!!!!!!!!!!!!! Ref: Iwagaki !!!!!!!!!!!!!!!!!!!!!!!
36     ! !*** no1 : flag =1 if tau_b > tau_{*c} ***
37     ! !do i=1,imesh
38     ! ! idou(i,1)=0.0044/dm ! 2mm or 1mm, then 0.0034
39     ! ! idou(i,2)=0.025/dm !5mm
40     ! ! idou(i,3)=0.05/dm !1cm
41     ! ! idou(i,4)=0.1/dm !2cm
42     ! ! idou(i,5)=0.25/dm !5cm

```

```

43      !      ! idou(i,6)=0.5/dm !10cm
44      !      !end do
45
46      ! 1. get flag 1 if tau > tau_critical
47      meshflag(:, :) = 0
48      where(tauc(:) > 0.0022d0)
49          meshflag(:, 1) = 1
50      end where
51      where(tauc(:) > tauc)
52          meshflag(:, 2) = 1
53          meshflag(:, 3) = 1
54          meshflag(:, 4) = 1
55          meshflag(:, 5) = 1
56          meshflag(:, 6) = 1
57      end where
58
59      !*** 2. connection of 1-0-1 -> 1-2-1 ***
60      do itau = 1, 6
61          do i = 1, Ngrid
62              if (meshflag(i, itau) /= 0) cycle
63
64                  ig_e = igc(iEE, i)
65                  ig_n = igc(iNN, i)
66                  ig_w = igc(iWW, i)
67                  ig_s = igc(iSS, i)
68                  ig_ne = igc(iEE, ig_n)
69                  ig_nw = igc(iWW, ig_n)
70                  ig_sw = igc(iWW, ig_s)
71                  ig_se = igc(iEE, ig_s)
72
73                  ! link of edge pattern
74                  ! 010
75                  ! 0x0
76                  ! 010
77                  if (meshflag(ig_n, itau) == 1 .and. &
78                      & meshflag(ig_s, itau) == 1) then
79                      meshflag(i, itau) = 2
80                  else if (meshflag(ig_e, itau) == 1 .and. &
81                      & meshflag(ig_w, itau) == 1) then
82                      meshflag(i, itau) = 2
83                  ! keima patter
84                  ! pattern a-1
85                  ! 100
86                  ! 0x0 * 4
87                  ! 010
88                  else if (meshflag(ig_nw, itau) == 1 .and. &
89                      & meshflag(ig_s, itau) == 1) then

```

```

90         meshflag(i, itau) = 2
91     else if (meshflag(ig_ne, itau) == 1 .and. &
92         & meshflag(ig_s, itau) == 1) then
93         meshflag(i, itau) = 2
94     else if (meshflag(ig_sw, itau) == 1 .and. &
95         & meshflag(ig_n, itau) == 1) then
96         meshflag(i, itau) = 2
97     else if (meshflag(ig_se, itau) == 1 .and. &
98         & meshflag(ig_n, itau) == 1) then
99         meshflag(i, itau) = 2
100     ! pattern a-2
101     ! 100
102     ! 0x0
103     ! 001
104     else if (meshflag(ig_ne, itau) == 1 .and. &
105         & meshflag(ig_sw, itau) == 1) then
106         meshflag(i, itau) = 2
107     else if (meshflag(ig_nw, itau) == 1 .and. &
108         & meshflag(ig_se, itau) == 1) then
109         meshflag(i, itau) = 2
110     ! pattern a-3
111     ! 100
112     ! 0x1
113     ! 000
114     else if (meshflag(ig_nw, itau) == 1 .and. &
115         & meshflag(ig_e, itau) == 1) then
116         meshflag(i, itau) = 2
117     else if (meshflag(ig_sw, itau) == 1 .and. &
118         & meshflag(ig_e, itau) == 1) then
119         meshflag(i, itau) = 2
120     else if (meshflag(ig_ne, itau) == 1 .and. &
121         & meshflag(ig_w, itau) == 1) then
122         meshflag(i, itau) = 2
123     else if (meshflag(ig_se, itau) == 1 .and. &
124         & meshflag(ig_w, itau) == 1) then
125         meshflag(i, itau) = 2
126     ! pattern a-4
127     ! 101
128     ! 0x0
129     ! 000
130     ! is not necessary
131     end if
132 end do
133 end do
134
135 ! 3. once all non zero value come to be 1
136 do itau = 1, 6

```

```

137     where(meshflag(:, itau) > 0)
138         meshflag(:, itau) = 1
139     end where
140 end do
141
142 allocate (Queue(Ngrid), QueueStack(Ngrid))
143
144 do itau = 1, 6
145     do ib = 1, nbc
146
147         do i = 1, Ngrid
148             Queue(i) = 0
149         end do
150         do i = 1, Ngrid
151             QueueStack(i) = 0
152         end do
153
154         nQueue = 1
155         Queue(ibg(ib)) = 1
156         QueueStack(nQueue) = ibg(ib)
157
158         do while (nQueue > 0)
159             next = QueueStack(nQueue)
160             meshflag(next, itau) = 2
161             Queue(next) = 0
162             QueueStack(nQueue) = 0
163             nQueue = nQueue - 1
164
165             ig(1) = igc(iEE, next)
166             ig(2) = igc(iNN, next)
167             ig(3) = igc(iWW, next)
168             ig(4) = igc(iSS, next)
169             ig(5) = igc(iEE, ig(2))
170             ig(6) = igc(iWW, ig(2))
171             ig(7) = igc(iWW, ig(4))
172             ig(8) = igc(iEE, ig(4))
173
174             do id = 1, 8
175                 if (meshflag(ig(id), itau) == 1 .and. Queue(ig(id)) == 0 &
176                     &.and. ig(id) /= 0) then
177                     nQueue = nQueue + 1
178                     QueueStack(nQueue) = ig(id)
179                     Queue(ig(id)) = 1
180                 end if
181             end do
182         end do
183     end do

```

```

184     end do
185
186     deallocate(Queue, QueueStack)
187
188     !*** up to here, calculate ***
189
190     ! write process
191     meshflag(:, 7) = 0
192     do itau = 1, 6
193         where(meshflag(:, itau) == 2)
194             meshflag(:, 7) = itau + 10
195         end where
196     end do
197
198     write(99, '(a10, 2a12, 6a10)') "id,", "longitude,", "latitude,", &
199         &"2mm,", "5mm,", "1cm,", "2cm,", "5cm,", "10cm"
200     write(99, '(i10, 6, "&
201         &g0, 6, "&g0, 6, "&
202         &g0, 6, "&g0, 6, "&g0, 6, "&g0, 6, "&g0, 6, "&g0)') &
203         &(i, lonlat(1, i), lonlat(2, i), &
204         &(meshflag(i, j), j = 1, 6), i = 1, Ngrid)
205     !*** up to here, output ***
206
207     return
208
209     contains
210     recursive subroutine lookup(i, it)
211         implicit none
212         integer, intent(in) :: i, it
213         integer :: id, ig(8)
214
215         nCallCount = nCallCount + 1
216         meshflag(i, it) = 2
217
218         ig(1) = igc(iEE, i)
219         ig(2) = igc(iNN, i)
220         ig(3) = igc(iWW, i)
221         ig(4) = igc(iSS, i)
222         ig(5) = igc(iEE, ig(2))
223         ig(6) = igc(iWW, ig(2))
224         ig(7) = igc(iWW, ig(4))
225         ig(8) = igc(iEE, ig(4))
226
227         do id = 1, 8
228             if (meshflag(ig(id), it) == 1 .and. ig(id) /= 0) then
229                 call lookup(ig(id), it)
230             end if

```



```
231         end do
232         return
233     end subroutine lookup
234 end subroutine tau
```

32 errnum.f90

errnum.f90

```
1  !-----
2  !
3  ! Copyright 2022
4  ! Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
5  ! Public Works Research Institute (PWRI)
6  !
7  ! License
8  ! Creative Common (CC) BY-SA 4.0
9  ! (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
10 ! (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
11 !
12 !-----
13 subroutine errnum(m)
14     implicit none
15     character m*10
16
17     write( *, '(11h_--error--_,a10)') m
18     write(80, '(11h_--error--_,a10)') m
19
20     stop
21 end subroutine errnum
```

33 Supplementary Tools

33.1 makeConnectiv.f90

makeConnectiv.f90

```
1 program make_connectiv
2     implicit none
3
4     ! Rectangular grid
5     integer(4) :: i, j, ig, is, idz
6     real(8) :: dx, dy
7     real(8), allocatable :: lonlat(:, :, :)
8     ! latitude and longitude
9     real(8), allocatable :: xy(:, :, :)
10    ! x, y position
11    real(8), allocatable :: ezi(:, :), hh0(:, :)
12    ! Bed elevation and initial water depth distribution
13    integer(4), allocatable :: ixc(:, :), iyc(:, :)
14    ! Computation flag
15    integer(4), allocatable :: numg(:, :), numxc(:, :), numyc(:, :)
16    ! Numbering
17
18    ! Arbitrary area
19    integer(4) :: Ngrid, Nxs, Nys
20    real(8), allocatable :: lonlat_g(:, :)
21    real(8), allocatable :: xy_g(:, :)
22    real(8), allocatable :: ezi_g(:, :), hh0_g(:, :)
23    integer(4), allocatable :: igc(:, :), iec(:, :)
24    integer(4), allocatable :: ixsc(:, :), ixsc(:, :), ixsgc(:, :)
25    integer(4), allocatable :: isxc(:, :), isyc(:, :), isgic(:, :)
26
27    ! Boundaries
28    ! - outflow boundary
29    integer(4) :: nxout, nyout
30    integer(4), allocatable :: ixout(:), iyout(:)
31    character(1), allocatable :: ixdir(:), iydir(:)
32    ! - wall boundary
33    integer(4) :: nxwall, nywall
34    integer(4), allocatable :: ixwall(:), iywall(:)
35
36    ! Direction enumeration
37    integer(4), parameter :: iEE = 1
38    integer(4), parameter :: iNN = 2
39    integer(4), parameter :: iWW = 3
40    integer(4), parameter :: iSS = 4
41    integer(4), parameter :: iNE = 1
```

```

42     integer(4), parameter :: iNW = 2
43     integer(4), parameter :: iSW = 3
44     integer(4), parameter :: iSE = 4
45     integer(4), parameter :: iEg = 1
46     integer(4), parameter :: iNg = 1
47     integer(4), parameter :: iWg = 2
48     integer(4), parameter :: iSg = 2
49
50     ! Parameter
51     real(8), parameter :: pi = acos(-1.d0)
52
53     ! I/O
54     integer(4) :: uasc, udem, ubc, uidx
55     integer(4) :: ierr
56     character(256) :: fm
57
58     ! Arc/Info ASCII Grid
59     character(256) :: hoge
60     real(8) :: xllcorner, yllcorner, cellsize
61     integer(4) :: ncols, nrows, NODATA_value
62     real(8) :: xmin, xmax, xmid, ymin, ymax, ymid
63     real(8) :: l_x, l_y
64
65     ! Hubeny function
66     real(8) :: hubeny
67
68     ! Coordinate and conversion mode
69     integer(4) :: convMode
70     integer(4), parameter :: mLatlon = 1
71     integer(4), parameter :: mRectangular = 2
72     integer(4), parameter :: mQuit = 9
73
74
75     write(*, *)
76     write(*, '(a)') "Converting 'dem.asc' to 'dem.dat', 'boundary.dat' & '
gridindex.asc'"
77     write(*, *)
78
79     ! write(*, '(a)') " ASCファイルの変換を行います。"
80     do
81         write(*, *)
82         ! write(*, '(a)') " 対象 DEMの座標系を選んでください。"
83         ! write(*, '(a)') " 1) 緯度経度座標"
84         ! write(*, '(a)') " 2) 平面直角座標系"
85         ! write(*, '(a)') " 9) 変換せず終了する"
86         write(*, '(a)') "Select Coordinate System of input DEM."
87         write(*, '(a)') "1) Long-Lat CS"

```

```

88     write(*, '(a)') " 2) Japan Plane Rectangular CS"
89     write(*, '(a)') " 9) exit, no conversion"
90     write(*, *)
91     write(*, '(a$)') ">>>"
92
93     read(*, *) convMode
94
95     if      (convMode == mLatlon .or. convMode == mRectangular) then
96         exit
97     else if (convMode == mQuit) then
98         ! write(*, '(a)') "終了します。"
99         write(*, '(a)') "END"
100        stop
101    else
102        ! write(*, '(a)') "1, 2, または 9 の数字を入力してください。"
103        write(*, '(a)') "Please input 1, 2 or 9"
104    end if
105 end do
106
107 ! Get ASC header
108 open(newunit = uasc, file = 'dem.asc')
109 read(uasc, *) hoge, ncols
110 read(uasc, *) hoge, nrows
111 read(uasc, *) hoge, xllcorner
112 read(uasc, *) hoge, yllcorner
113 read(uasc, *) hoge, cellsize
114 read(uasc, *) hoge, NODATA_value
115
116 ! Allocation for rect. grid (including dummy grid and edge)
117 allocate(ezi(0:nrows+1, 0:ncols+1), hh0(0:nrows+1, 0:ncols+1))
118 allocate(lonlat(2, 0:nrows+1, 0:ncols+1), xy(2, 0:nrows+1, 0:ncols+1))
119 allocate(ixe(0:nrows+1, 0:ncols+2), iye(0:nrows+2, 0:ncols+1))
120 allocate(numg(0:nrows+1, 0:ncols+1))
121 allocate(numxe(0:nrows+1, 0:ncols+2), numye(0:nrows+2, 0:ncols+1))
122 lonlat(:, :, :) = 0.d0
123 xy(:, :, :) = 0.d0
124 ezi(:, :) = 0.d0
125 hh0(:, :) = 0.d0
126 ixe(:, :) = 0
127 iye(:, :) = 0
128 numg(:, :) = 0
129 numxe(:, :) = 0
130 numye(:, :) = 0
131
132 ! Get elevation
133 do i = nrows, 1, -1
134     read(uasc, *) (ezi(i, j), j = 1, ncols)

```

```

135     end do
136
137     ! Define computational domain
138     ! Min, max are defined at elevation points
139     xmin = xllcorner + cellsize * 0.5d0
140     ymin = yllcorner + cellsize * 0.5d0
141     xmax = xmin + (ncols - 1) * cellsize
142     ymax = ymin + (nrows - 1) * cellsize
143     xmid = (xmax + xmin) * 0.5d0
144     ymid = (ymax + ymin) * 0.5d0
145     select case (convMode)
146     case (mLatlon)
147         l_x = hubeny(ymid, xmin, ymid, xmax)
148         l_y = hubeny(ymin, xmid, ymax, xmid)
149         dx = l_x / (ncols - 1)
150         dy = l_y / (nrows - 1)
151     case (mRectangular)
152         l_x = xmax - xmin
153         l_y = ymax - ymin
154         dx = cellsize
155         dy = cellsize
156     end select
157
158     write(*, *)
159     write(*, '(a,f20.2,a,f20.2)') 'Bottom-left: ', xmin, ', ', ymin
160     write(*, '(a,f20.2,a,f20.2)') 'Center: ', xmid, ', ', ymid
161     write(*, '(a,f20.2,a,f20.2)') 'Top-right: ', xmax, ', ', ymax
162     write(*, *)
163     write(*, '(a,f10.2,a,f10.2,a)') 'Region size: ', l_x, 'm_x', l_y, 'm
164
165
166
167     ! +-----+
168     ! | + + + + + |
169     ! |
170     ! |
171     ! |
172     ! |
173     ! |
174     ! |
175     ! |
176     ! |
177     ! |
178     ! |
179     ! |
180     ! +-----+

```

```

181
182      ! Latitude and longitude (including dummy grid)
183      ! In case of rectangular coordinate, latlon is copy of xy position.
184      do j = 0, ncols + 1
185          do i = 0, nrows + 1
186              lonlat(1, i, j) = xmin + cellsize * dble(j - 1)
187              lonlat(2, i, j) = ymin + cellsize * dble(i - 1)
188          end do
189      end do
190
191      ! Position (including dummy grid)
192      select case (convMode)
193      case (mLatlon)
194          do j = 0, ncols + 1
195              do i = 0, nrows + 1
196                  xy(1, i, j) = dx * dble(j - 1)
197                  xy(2, i, j) = dy * dble(i - 1)
198              end do
199          end do
200      case (mRectangular)
201          do j = 0, ncols + 1
202              do i = 0, nrows + 1
203                  xy(1, i, j) = xmin + cellsize * dble(j - 1)
204                  xy(2, i, j) = xmin + cellsize * dble(i - 1)
205              end do
206          end do
207      end select
208
209
210      ! Coordinate
211      !
212      !      Y          i+          N
213      !      ^          ^          ^
214      !      |          |          |
215      !      +---> X    +---> j+    +---> E
216      !
217
218      !      (i+1, j)
219      !      +-----+
220      !      |          |
221      !      |  (i, j)  |
222      !      |    +    |
223      !      |          |
224      !      |(i, j)    | (i, j+1)
225      !      +-----+
226      !      (i, j)
227      !

```

```

228      !
229
230      ! Numbering grid
231      Ngrid = 0
232      do j = 1, ncols
233          do i = 1, nrows
234              if(ezi(i, j) >= 0.d0) then
235                  Ngrid = Ngrid + 1
236                  numg(i, j) = Ngrid
237
238                  ixc(i, j) = 1
239                  ixc(i, j+1) = 1
240                  iyc(i, j) = 1
241                  iyc(i+1, j) = 1
242
243              end if
244          end do
245      end do
246
247      ! Numbering edge
248      Nxs = 0
249      Nys = 0
250      do j = 1, ncols + 1
251          do i = 1, nrows
252              if(ixc(i, j) == 1) then
253                  Nxs = Nxs + 1
254                  numxc(i, j) = Nxs
255
256              end if
257          end do
258      end do
259      do j = 1, ncols
260          do i = 1, nrows + 1
261              if(iyc(i, j) == 1) then
262                  Nys = Nys + 1
263                  numyc(i, j) = Nys
264
265              end if
266          end do
267      end do
268
269      ! Make connectivity
270      allocate(lonlat_g(2, 0:Ngrid), xy_g(2, 0:Ngrid))
271      allocate(ezi_g(0:Ngrid), hh0_g(0:Ngrid), igc(4, 0:Ngrid), iec(4, 0:Ngrid)
272      ))
273      allocate(ixsxc(4, 0:Nxs), ixsync(4, 0:Nxs), ixsgc(2, 0:Nxs))
274      allocate(iysxc(4, 0:Nys), iysync(4, 0:Nys), iysgc(2, 0:Nys))

```



```

274
275     xy_g=0.d0
276     lonlat_g=0.d0
277     ezi_g=0.d0
278     hh0_g=0.d0
279     igc=0
280     iec=0
281     ixsvc=0
282     ixsvc=0
283     ixsgc=0
284     iysvc=0
285     iysvc=0
286     iysgc=0
287
288     do j = 1, ncols
289         do i = 1, nrows
290             if(numg(i, j) > 0) then
291                 ig = numg(i, j)
292                 igc(iEE, ig) = numg(i, j+1)
293                 igc(iNN, ig) = numg(i+1, j)
294                 igc(iWW, ig) = numg(i, j-1)
295                 igc(iSS, ig) = numg(i-1, j)
296
297                 iec(iEE, ig) = numxe(i, j+1)
298                 iec(iNN, ig) = numye(i+1, j)
299                 iec(iWW, ig) = numxe(i, j)
300                 iec(iSS, ig) = numye(i, j)
301
302                 lonlat_g(:, numg(i, j)) = lonlat(:, i, j)
303                 xy_g(:, numg(i, j))      = xy(:, i, j)
304                 ezi_g(numg(i, j))        = ezi(i, j)
305                 hh0_g(numg(i, j))        = hh0(i, j)
306
307             end if
308         end do
309     end do
310     do j = 1, ncols + 1
311         do i = 1, nrows
312             if(ixe(i, j) == 1) then
313                 is = numxe(i, j)
314                 ixsvc(iEE, is) = numxe(i, j+1)
315                 ixsvc(iNN, is) = numxe(i+1, j)
316                 ixsvc(iWW, is) = numxe(i, j-1)
317                 ixsvc(iSS, is) = numxe(i-1, j)
318
319                 ixsvc(iNE, is) = numye(i+1, j)
320                 ixsvc(iNW, is) = numye(i+1, j-1)

```

```

321             ixsync(iSW, is) = numye(i , j-1)
322             ixsync(iSE, is) = numye(i , j )
323
324             ixsgc(iEg, is) = numg(i , j )
325             ixsgc(iWg, is) = numg(i , j-1)
326
327         end if
328     end do
329 end do
330 do j = 1, ncols
331     do i = 1, nrows + 1
332         if(iye(i, j) == 1) then
333             is = numye(i, j)
334             isyc(iEE, is) = numye(i , j+1)
335             isyc(iNN, is) = numye(i+1, j )
336             isyc(iWW, is) = numye(i , j-1)
337             isyc(iSS, is) = numye(i-1, j )
338
339             isxc(iNE, is) = numxe(i , j+1)
340             isxc(iNW, is) = numxe(i , j )
341             isxc(iSW, is) = numxe(i-1, j )
342             isxc(iSE, is) = numxe(i-1, j+1)
343
344             isgc(iNg, is) = numg(i , j )
345             isgc(iSg, is) = numg(i-1, j )
346
347         end if
348     end do
349 end do
350
351 ! — Assume that the computational domain is surrounded
352 !           by open boundary (out flow)
353 nxout = 0
354 nyout = 0
355 do i = 1, Nxs
356     if (ixsgc(iEg, i) == 0) then
357         nxout = nxout + 1
358     end if
359     if (ixsgc(iWg, i) == 0) then
360         nxout = nxout + 1
361     end if
362 end do
363 do i = 1, Nys
364     if (isgc(iNg, i) == 0) then
365         nyout = nyout + 1
366     end if
367     if (isgc(iSg, i) == 0) then

```

```

368         nyout      = nyout + 1
369     end if
370 end do
371
372 if(nxout /= 0) then
373     allocate(ixout(nxout), ixdir(nxout))
374     nxout = 0
375     ixout(:) = 0
376     ixdir(:) = ""
377     do i = 1, Nxs
378         if (ixsgc(iEg, i) == 0) then
379             nxout      = nxout + 1
380             ixout(nxout) = i
381             ixdir(nxout) = "+"
382         end if
383         if (ixsgc(iWg, i) == 0) then
384             nxout      = nxout + 1
385             ixout(nxout) = i
386             ixdir(nxout) = "-"
387         end if
388     end do
389 end if
390 if(nyout /= 0) then
391     allocate(iyout(nyout), iydir(nyout))
392     nyout = 0
393     iyout(:) = 0
394     iydir(:) = ""
395     do i = 1, Nys
396         if (iysgc(iNg, i) == 0) then
397             nyout      = nyout + 1
398             iyout(nyout) = i
399             iydir(nyout) = "+"
400         end if
401         if (iysgc(iSg, i) == 0) then
402             nyout      = nyout + 1
403             iyout(nyout) = i
404             iydir(nyout) = "-"
405         end if
406     end do
407 end if
408
409 ! — Assume that the computational domain is surrounded by solid wall
410 ! - wall boundary
411 nxwall = 0
412 nywall = 0
413 ! do i = 1, Nxs
414 ! if (ixsgc(iEg, i) == 0 .or. ixsgc(iWg, i) == 0) then

```

```

415      !           nxwall = nxwall + 1
416      !     end if
417      ! end do
418      ! do i = 1, Nys
419      !   if (iysgc(iNg, i) == 0 .or. iysgc(iSg, i) == 0) then
420      !     nywall = nywall + 1
421      !   end if
422      ! end do
423
424      ! if (nxwall /= 0) then
425      !   allocate(ixwall(nxwall))
426      !   nxwall = 0
427      !   ixwall(:) = 0
428      !   do i = 1, Nxs
429      !     if (ixsgc(iEg, i) == 0 .or. ixsgc(iWg, i) == 0) then
430      !       nxwall = nxwall + 1
431      !       ixwall(nxwall) = i
432      !     end if
433      !   end do
434      ! end if
435      ! if (nywall /= 0) then
436      !   allocate(iywall(nywall))
437      !   nywall = 0
438      !   iywall(:) = 0
439      !   do i = 1, Nys
440      !     if (iysgc(iNg, i) == 0 .or. iysgc(iSg, i) == 0) then
441      !       nywall = nywall + 1
442      !       iywall(nywall) = i
443      !     end if
444      !   end do
445      ! end if
446
447      idz = 0 ! erosion flag 0->off
448
449      ! -----
450      ! === DEM summary ===
451      ! -----
452
453      write(*, *)
454      write(*, *)
455      write(*, '( "Node:␣",␣i10,␣"␣␣␣␣␣X-edge:␣",␣i10,␣"␣␣␣␣␣Y-edge:␣",␣i10)')&
456      & Ngrid, Nxs, Nys
457      write(*, *)
458      write(*, '(a,␣f10.2,␣a)') 'dx:␣', dx, '␣m'
459      write(*, '(a,␣f10.2,␣a)') 'dy:␣', dy, '␣m'
460      write(*, *)
461

```

```

462      ! -----
463      ! === Output data ===
464      ! -----
465
466      ! — Output DEM file
467      write(*, *) 'Writing_grid_data...'
468
469      ! — DEM : text format —
470      open(100, file='dem.dat')
471      !— Bed elevation
472      write(100, '(3i10,2E25.15,i10)') Ngrid, Nxs, Nys, dx, dy, idz
473      do i=0,Ngrid
474          write(100, '(6(E25.15))') xy_g(1:2,i), lonlat_g(1:2,i), ezi_g(i), hh0_g(i)
475      end do
476      !— Grid connectivities
477      do i=0,Ngrid
478          write(100, '(8i10)') igc(1:4,i), iec(1:4,i)
479      end do
480      !— Edge connectivities
481      do i=0,Nxs
482          write(100, '(10i10)') ixsc(1:4,i), ixsc(1:4,i), ixsc(1:2,i)
483      end do
484      do i=0,Nys
485          write(100, '(10i10)') isxc(1:4,i), isyc(1:4,i), isgc(1:2,i)
486      end do
487      close(100)
488      ! <— dem.dat to txt
489      !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
490
491
492      ! — Output boundary conditions
493      write(*, *) 'Writing_boundary_conditions...'
494      ! open(newunit = ubc, file = "boundary.dat", status = "unknown")
495      ! ! — Out flow boundary
496      ! write(ubc, '(2i10)') nxout, nyout
497      ! if(nxout > 0) write(ubc, '(4(i10, a5))') (ixout(i), ixdir(i), i = 1,
nxout)
498      ! if(nyout > 0) write(ubc, '(4(i10, a5))') (iyout(i), iydir(i), i = 1,
nyout)
499      ! ! — Wall boundary
500      ! write(ubc, '(2i10)') nxwall, nywall
501      ! if(nxwall > 0) write(ubc, '(4(2i10))') (i, ixwall(i), i = 1, nxwall)
502      ! if(nywall > 0) write(ubc, '(4(2i10))') (i, iywall(i), i = 1, nywall)
503
504      ! — boundary : text format —
505      open(100, file='boundary.dat')
506      ! — Out flow boundary

```

```

507     write(100, '(2i10)') nxout, nyout
508     if(nxout > 0)then
509         do i=1,nxout
510             write(100, '(i10,a)') ixout(i), ixdir(i)
511         end do
512     end if
513     if(nyout > 0)then
514         do i=1,nyout
515             write(100, '(i10,a)') iyout(i), iydir(i)
516         end do
517     end if
518     ! - Wall boundary
519     write(100, '(2i10)') nxwall, nywall
520     if(nxwall > 0)then
521         do i=1,nxwall
522             write(100, '(i10)') ixwall(i)
523         end do
524     end if
525     if(nywall > 0)then
526         do i=1,nywall
527             write(100, '(i10)') iywall(i)
528         end do
529     end if
530
531
532     ! — Output grid index
533     write(*, *) 'Writing_grid_index...'
534     open(newunit = uidx, file = "gridindex.asc", status = "unknown")
535     write(uidx, '( "ncols#####",i0)') ncols
536     write(uidx, '( "nrows#####",i0)') nrows
537     write(uidx, '( "xllcorner####",f0.12)') xllcorner
538     write(uidx, '( "yllcorner####",f0.12)') yllcorner
539     write(uidx, '( "cellsize#####",f0.12)') cellsize
540     write(uidx, '( "NODATA_value",i0)') NODATA_value
541     write(fm, '( ("",i0,"i10")') ncols
542     do i = nrows, 1, -1
543         write(uidx, fm) (numg(i, j), j = 1, ncols)
544     end do
545
546     write(*, *)
547     write(*, *) "Conversion_end."
548     write(*, *)
549
550 end program make_connectiv
551
552
553 function hubeny(lat1, lon1, lat2, lon2)

```

```

554      ! reffering to:
555      ! http://yamadarake.jp/trdi/report000001.html
556
557      real(8) :: hubeny
558      real(8) :: lat1 , lon1 , lat2 , lon2
559      real(8) :: x1 , y1 , x2 , y2
560      ! real(8) :: dx, dy, M, N, muy, b, e, f, W
561      real(8) :: dx, dy, M, N, muy, e, W
562      real(8), parameter :: deg2rad = acos(-1.d0) / 180.d0
563      real(8), parameter :: a = 6378137.d0           ! GRS80
564      ! real(8), parameter :: a = 6378137.d0       ! WGS84
565      real(8), parameter :: b = 6356752.314140d0   ! GRS80
566      ! real(8), parameter :: b = 6356752.314245d0 ! WGS84
567      real(8), parameter :: rf = 298.257222101d0
568
569      x1 = deg2rad * lon1
570      x2 = deg2rad * lon2
571      y1 = deg2rad * lat1
572      y2 = deg2rad * lat2
573
574      dx = (x1 - x2)
575      dy = (y1 - y2)
576
577      muy = (y1 + y2) * 0.5d0
578      e = sqrt((a * a - b * b) / (a * a))
579      ! f = 1.d0 / rf
580      ! e = f * (2.d0 - f)
581      ! b = a * (1.d0 - f)
582      ! W = sqrt(1.d0 - e * sin(muy) * sin(muy))
583      W = sqrt(1.d0 - e * e * sin(muy) * sin(muy))
584      ! M = b / (W * W * W)
585      M = a * (1 - e * e) / (W * W * W)
586      N = a / W
587
588      hubeny = sqrt((dy * M)**2 + (dx * N * cos(muy))**2)
589
590 end function hubeny

```

33.2 projKawase2011.py

projKawase2011.py

```
1 # -*- coding: utf-8 -*-
2 # Copyright 2022
3 # Ministry of Land, Infrastructure, Transport and Tourism (MLIT) and
4 # Public Works Research Institute (PWRI)
5 #
6 # License
7 # Creative Common (CC) BY-SA 4.0
8 # (JP) https://creativecommons.org/licenses/by-sa/4.0/legalcode.ja
9 # (EN) https://creativecommons.org/licenses/by-sa/4.0/legalcode
10 #
11 # Author
12 # SHIMIZU Takeshi
13 #
14 # Date
15 # 2014/3/13
16 # revision: 2022/3/24
17 #
18 # Comment
19 # This code is translated from javascript to python by the author.
20 # Original Javascript Code is written by Kawase 2011
21 #
22 # Reference
23 # 河瀬和重 Kawase Kazushige, 2011:
24 # 赤道からの子午線弧長を任意に与えて該当する緯度を求めるより簡明な計算方法
25 # A More Concise Method of Calculation for the Coordinate Conversion
26 # between Geographic and Plane Rectangular Coordinates
27 # on the Gauss–Kruger Projection
28 # 国土地理院時報 (2011, 121集)
29 # GSI annual report, vol.121
30 # URL: http://www.gsi.go.jp/REPORT/JIHO/vol121-main.html
31 #
32 import math
33 import sys
34 import os
35
36 def ddd2dms(coord):
37     """
38     input
39         coord(string): dd.dxxxxx
40     output
41         coord(string): ddmms.ss
42
```



```

43     ref: https://github.com/KMR-zoar/dms2d3/blob/master/dms2d3.rb
44     """
45     coord = float(coord)
46     dd = math.floor(coord)
47     mm = (coord - dd)*60.0
48     ss = (mm - math.floor(mm))*60.0
49     if len(str(math.floor(ss))) == 1:
50         sss = "0" + str(ss)
51     else:
52         sss = str(ss)
53     return str((dd * 100 + math.floor(mm))) + sss
54
55 def dms2ddd(coord):
56     """
57     input
58         coord(string): dmmss.ss or dddmss.ss
59     output
60         coord(string): dd.ddd
61
62     ref: https://github.com/KMR-zoar/dms2d3/blob/master/dms2d3.rb
63     """
64     num_dd = 0
65     dms = str(coord)
66
67     if len(str(math.floor(float(dms)))) == 7:
68         num_dd = 3
69     elif len(str(math.floor(float(dms)))) == 6:
70         num_dd = 2
71
72     dd = dms[0:num_dd]
73     mm = dms[num_dd:num_dd+2]
74     ss = dms[num_dd+2:]
75
76     ddd = float(dd) + float(mm)/60 + float(ss)/3600
77
78     return str(ddd)
79
80
81
82 class dms2xy(object):
83     """
84     input lon, lat format is:
85         dddmss.ss, dddmss.ss
86     """
87
88     def __init__(self, num, lon, lat, format="ddd"):
89         # num = number of origin in JGD2000 CS

```

```

90     # phi = longitude in JGS2000
91     # lmbd = latitude in JGS2000
92     self.num = num
93     if format == "ddd":
94         lon = float(ddd2dms(lon))
95         lat = float(ddd2dms(lat))
96     self.lon = float(lon) # phi in calc function
97     self.lat = float(lat) # lmbd in calc function
98
99     # x = x coordinates on target origin (num)
100    # y = y coordinates on target origin (num)
101    # gmm = azimuth from north in radian
102    # m = shyukusyaku keisu
103    x, y, gmm, m = self._calc_deg2xy(num, self.lat, self.lon)
104
105    self.x = x
106    self.y = y
107    self.gmm = gmm
108    self.m = m
109
110    self._azimuth_rad2dms(gmm)
111
112    def output(self):
113        return self.x, self.y
114
115
116    def _calc_deg2xy(self, num, phi, lmbd):
117        # global variables :
118        a = 6378137.0
119        rf = 298.257222101
120        m0 = 0.9999
121        s2r = math.pi/648000.0
122
123        # variables
124        n = 0.5/(rf-0.5)
125        n15 = 1.5*n
126        anh = 0.5*a/(1.0+n)
127        nsq = n*n
128        e2n = 2.0*math.sqrt(n)/(1.0+n)
129        ra = 2.0*anh*m0*(1.0+nsq/4.0 + nsq*nsq/64.0)
130        jt = 5
131        jt2 = 2*jt
132        ep = 1.0
133
134        # list variables
135        e = [0 for x in range(0, jt+jt+1)]
136        e[0] = None

```

```

137     alp = [None, 0.0, 0.0, 0.0, 0.0, 0.0]
138
139     for k in range(1, jt + 1):
140         e[k] = n15/float(k) - n
141         e[k+jt] = n15/float(k+jt) - n
142         ep *= e[k]
143
144     # parameter of expansion
145     alp[1] = (1.0/2.0+(-2.0/3.0+(5.0/16.0+(41.0/180.0-127.0/288.0*n)*n)*n)*n
146 )*n
147     alp[2] = (13.0/48.0+(-3.0/5.0+(557.0/1440.0+281.0/630.0*n)*n)*n)*nsq
148     alp[3] = (61.0/240.0+(-103.0/140.0+15061.0/26880.0*n)*n)*n*nsq
149     alp[4] = (49561.0/161280.0-179.0/168.0*n)*nsq*nsq
150     alp[5] = 34729.0/80640.0*n*nsq*nsq
151
152     # origin coordinates of JGS2000 CS
153     # unit of lat(phi0) is degree, lon(lmbd0) is minute
154     phi0 = [0,33,33,36,33,36,36,36,36,36,40,44,44,44,26,26,26,26,20,26]
155     lmbd0 = [0,7770,7860,7930,8010,8060,8160,8230,8310,8390, \
156             8450,8415,8535,8655,8520,7650,7440,7860,8160,9240]
157
158     phideg = math.floor(phi/10000)
159     phimin = math.floor((phi-phideg*10000)/100)
160     phirad = (phideg*3600+phimin*60+phi-phideg*10000-phimin*100)*s2r
161     lmbddeg = math.floor(lmbd/10000)
162     lmbdmin = math.floor((lmbd-lmbddeg*10000)/100)
163     lmbdsec = lmbddeg*3600+lmbdmin*60+lmbd-lmbddeg*10000-lmbdmin*100
164
165     # main calculation part
166     sphi = math.sin(phirad)
167     nphi = (1.0-n)/(1+n)*math.tan(phirad)
168     dlmbd = (lmbdsec-lmbd0[num]*60)*s2r
169     sdlmbd = math.sin(dlmbd)
170     cdlmbd = math.cos(dlmbd)
171     tchi = math.sinh( math.atanh(sphi) - e2n * math.atanh(e2n*sphi) )
172     cchi = math.sqrt( 1.0 + tchi * tchi )
173     xi = math.atan2( tchi , cdlmbd )
174     xip = math.atan2( tchi , cdlmbd )
175     etap = math.atanh( sdlmbd/cchi )
176     eta = math.atanh( sdlmbd/cchi )
177     sgm = 1
178     tau = 0
179
180     ite = []
181     ite = list(range(len(alp)-1))
182     ite = [x+1 for x in ite]
183     for j in reversed(ite[0:len(ite)]):

```

```

183         alsin = alp[j] * math.sin(2.0 * float(j) * xip)
184         alcos = alp[j] * math.cos(2.0 * float(j) * xip)
185         xi    += alsin * math.cosh(2.0 * float(j) * etap)
186         eta   += alcos * math.sinh(2.0 * float(j) * etap)
187         sgm   += 2.0 * float(j) * alcos * math.cosh(2.0 * float(j) * etap)
188         tau   += 2.0 * float(j) * alsin * math.sinh(2.0 * float(j) * etap)
189
190     x    = ra*xi - m0*self._Merid(2.0*phi0[num]*3600.0*s2r, e, ep, jt, jt2,
anh)
191     y    = ra*eta
192     gmm = math.atan2(tau*cchi*cdlmbd+sgm*tchi*sdlmbd, sgm*cchi*cdlmbd-tau*
tchi*sdlmbd)
193     m    = ra/a*math.sqrt((sgm*sgm+tau*tau)/(tchi*tchi+cdlmbd*cdlmbd)*(1+nphi*
nphi))
194
195
196     return x, y, gmm, m
197
198
199 def _Merid(self, phi2, e, ep, jt, jt2, anh):
200     # function of calculating arc median line length from the equator
201     # using input latitude * 2
202     t = [None]
203     s = []
204
205     dc = 2.0*math.cos(phi2)
206     s.append(0.0) # s[0] = 1.0
207     s.append( math.sin(phi2) )
208     for i in range(1, jt2+1):
209         s.append( dc * s[i] - s[i-1] )
210         t.append( ( 1.0/float(i) - 4.0*float(i) )*s[i] )
211
212     summation = 0.0
213     c1 = ep
214     j = jt
215     while j:
216         c2 = phi2
217         c3 = 2.0
218         l = j
219         m = 0
220         while l:
221             c3 /= e[l]
222             tmp_a = c3 * t[m+1]
223             c3 *= e[2*j-(l-1)]
224             tmp_b = c3 * t[m+2]
225             c2 += tmp_a + tmp_b
226             l -= 1

```

```

227         m += 2
228         summation += c1*c1*c2
229         c1 /= e[j]
230         j -= 1
231
232     return anh*(summation+phi2)
233
234
235     def __azimuth_rad2dms(self, gmm):
236         s2r = math.pi/648000.0
237         sgn = (gmm < 0)
238         gdo = math.floor(gmm/s2r/3600)+sgn
239         gfun = math.floor((gmm/s2r-gdo*3600)/60)+sgn
240         gbyou = gmm/s2r-gdo*3600-gfun*60
241
242         if sgn == True:
243             sgn = "-"
244         else:
245             sgn = "+"
246
247         self.azimuth = sgn+str(int(gdo))+":"+str(int(gfun))+":"+str(gbyou)
248
249         return gdo, gfun, gbyou
250
251
252     def summary(self):
253         sgn = "+" if self.gmm < 0 else "-"
254         gdo, gfun, gbyou = self.__azimuth_rad2dms(self.gmm)
255         sys.stdout.write("Input_\n")
256         sys.stdout.write("lon=%f;lat=%f\n" % (self.lon, self.lat) )
257         sys.stdout.write("Output_\n")
258         sys.stdout.write("no=%i\n" % (self.num) )
259         sys.stdout.write("X_survey=%f;Y_survey=%f\n" % (self.x, self.y)
)
260         sys.stdout.write("Supplementary_Info.\n")
261         sys.stdout.write("lambdaa=%s;f_deg=%f;min=%f\n" % \
262             (sgn, abs(gdo), abs(gfun), abs(gbyou)) )
263         sys.stdout.write("s_azimuth_from_north_clockwise\n" % self.azimuth)
264         sys.stdout.write("m=%f\n" % self.m)
265
266
267
268     class xy2dms(object):
269
270         def __init__(self, num, x, y, format="ddd"):
271
272             self.num = int(num)

```

```

273     self.x = float(x)
274     self.y = float(y)
275
276     lon, lat, gmm, m = self._calc_xy2dms(num, x, y)
277
278     self.lon = lon
279     self.lat = lat
280     self.gmm = gmm
281     self.m = m
282
283     self._azimuth_rad2dms(gmm)
284
285     if format == "ddd":
286         self.lon = dms2ddd(self.lon)
287         self.lat = dms2ddd(self.lat)
288
289     def output(self):
290         return self.lon, self.lat
291
292     def _calc_xy2dms(self, num, x, y):
293         # global variables
294         a = 6378137.0
295         rf = 298.257222101
296         m0 = 0.9999
297         s2r = math.pi/648000.0
298         n = 0.5/(rf-0.5)
299         n15 = 1.5*n
300         anh = 0.5*a/(1+n)
301         nsq = n*n
302         ra = 2.0*anh*m0 * (1.0 + nsq/4.0 + nsq*nsq/64.0)
303         jt = 5
304         jt2 = 2 * jt
305         ep = 1.0
306         e = list(range(0, jt+jt+1))
307         beta = [None]
308         dlt = [None]
309
310         for k in range(1, jt+1):
311             e[k] = n15/float(k) - n
312             e[k+jt] = n15/float(k+jt) - n
313             ep *= e[k]
314
315         # parameters of expansion
316         beta.append( (1.0/2.0+(-2.0/3.0+(37.0/96.0+(-1.0/360.0-81.0/512.0*n)*n)*n)
317 )*n)*n )
318         beta.append( (1.0/48.0+(1.0/15.0+(-437.0/1440.0+46.0/105.0*n)*n)*n)*nsq)
319         beta.append( (17.0/480.0+(-37.0/840.0-209.0/4480.0*n)*n)*n*nsq )

```

```

319         beta.append( (4397.0/161280.0-11.0/504.0*n)*nsq*nsq )
320         beta.append( 4583.0/161280.0*n*nsq*nsq )
321         dlt.append( (2.0+(-2.0/3.0+(-2.0+(116.0/45.0+(26.0/45.0-2854.0/675.0*n)*n
)*n)*n)*n)*n)
322         dlt.append( (7.0/3.0+(-8.0/5.0+(-227.0/45.0+(2704.0/315.0+2323.0/945.0*n
)*n)*n)*n)*nsq)
323         dlt.append( (56.0/15.0+(-136.0/35.0+(-1262.0/105.0+73814.0/2835.0*n)*n)*n
)*n*nsq)
324         dlt.append( (4279.0/630.0+(-332.0/35.0-399572.0/14175.0*n)*n)*nsq*nsq )
325         dlt.append( (4174.0/315.0-144838.0/6237.0*n)*n*nsq*nsq )
326         dlt.append( 601676.0/22275.0*nsq*nsq*nsq )
327
328         # 平面直角座標系の中心
329         # origin coordiantes of JGD2000 CR
330         # lat: degree, lon: min
331         phi0=[0,33,33,36,33,36,36,36,36,40,44,44,44,26,26,26,26,20,26]
332         lmbd0=[0,7770,7860,7930,8010,8060,8160,8230,8310,8390,\
333             8450,8415,8535,8655,8520,7650,7440,7860,8160,9240]
334
335
336         # main part of this function
337         xip=(x + m0 * self._Merid(2*phi0[num]*3600*s2r, e, ep, jt, jt2, anh))/
ra
338         xi = (x + m0 * self._Merid(2*phi0[num]*3600*s2r, e, ep, jt, jt2, anh))/
ra
339         eta = y/ra
340         etap = eta
341         sgmp = 1
342         taup = 0
343         for j in range(len(beta)-1, 0, -1):
344             besin = beta[j] * math.sin(2.0 * float(j) * xi )
345             becos = beta[j] * math.cos(2.0 * float(j) * xi )
346             xip -= besin * math.cosh(2.0 * float(j) * eta )
347             etap -= becos * math.sinh(2.0 * float(j) * eta )
348             sgmp -= 2.0 * float(j) * becos * math.cosh( 2.0 * float(j) * eta )
349             taup += 2.0 * float(j) * besin * math.sinh( 2.0 * float(j) * eta )
350
351         sxip = math.sin(xip)
352         cxip = math.cos(xip)
353         shetap = math.sinh(etap)
354         chetap = math.cosh(etap)
355         phi = math.asin(sxip/chetap)
356         chi = math.asin(sxip/chetap)
357
358         for j in range(len(dlt)-1, 0, -1):
359             phi += dlt[j] * math.sin( 2.0 * float(j) * chi )
360

```

```

361     nphi = (float(1.0) - n)/(float(1.0) + n) * math.tan(phi)
362     lmbd = lmbd0[num]*60.0 + math.atan2(shetap, cxip)/s2r
363     gmm = math.atan2(taup*cxip*chetap + sgmp*sxip*shetap, sgmp*cxip*chetap -
    taup*sxip*shetap)
364     m = ra/a*math.sqrt((cxip*cxip+shetap*shetap)/(sgmp*sgmp+taup*taup
    )*(1.0+nphi*nphi))
365
366     # convert radian to deg, min, second
367     # Format: dmmss.s
368     ido = math.floor(phi/s2r/3600.0)
369     ifun = math.floor((phi/s2r - ido*3600.0)/60.0)
370     ibyou = phi/s2r - ido*3600.0 - ifun*60.0
371     keido = math.floor(lmbd/3600.0)
372     keifun = math.floor((lmbd-keido*3600.0)/60.0)
373     keibyou = lmbd-keido*3600.0 - keifun*60.0
374
375
376     # Format arrange
377     # lon: dddmmss.ss in str
378     # lat: dmmss.ss in str
379     ido = int(math.floor(ido))
380     ifun = int(math.floor(ifun))
381     sido = str(ido)
382     sifun = str(ifun) if ifun >= 10 else "0" + str(ifun)
383     sibyou = str(ibyou) if ibyou >= 10.0 else "0" + str(ibyou)
384
385     keido = int(math.floor(keido))
386     keifun = int(math.floor(keifun))
387     skeido = str(keido)
388     skeifun = str(keifun) if keifun >= 10 else "0" + str(keifun)
389     skeibyou = str(keibyou) if keibyou >= 10.0 else "0" + str(keibyou)
390
391     lat = sido + sifun + sibyou
392     lon = skeido + skeifun + skeibyou
393
394
395
396     return lon, lat, gmm, m
397
398
399
400     # function for calculating meridian arc length separated from equator
401     # using input latitude * 2
402     def __Merid(self, phi2, e, ep, jt, jt2, anh):
403         t = [None]
404         s = []
405

```



```

406     dc = 2.0 * math.cos(phi2)
407     s.append(0.0)
408     s.append( math.sin(phi2) )
409
410     for i in range(1, jt2+1):
411         s.append( dc * s[i] - s[i-1] )
412         t.append( (1.0/float(i) - 4.0*float(i))*s[i] )
413
414     summation = 0.0
415     c1 = ep
416     j = jt
417     while j:
418         c2 = phi2
419         c3 = 2.0
420         l = j
421         m = 0
422         while l:
423             c3 /= e[l]
424             tmp_a = c3 * t[m+1]
425             c3 *= e[2*j - (l-1)]
426             tmp_b = c3 * t[m+2]
427             c2 += tmp_a + tmp_b
428             l -= 1
429             m += 2
430         summation += c1*c1*c2
431         c1 /= e[j]
432         j -= 1
433
434     return anh*(summation + phi2)
435
436
437
438 def __azimuth_rad2dms(self, gmm):
439
440     s2r = math.pi/648000.0
441
442     # note bool is converted as integer: True = 1 or False = 0
443     sgn = (gmm<0)
444     gdo = math.floor( gmm/s2r/3600.0 ) + sgn
445     gfun = math.floor( (gmm/s2r-gdo*3600.0)/60.0 ) + sgn
446     gbyou = gmm/s2r - gdo*3600.0 - gfun*60.0
447
448     if sgn == True:
449         sgn = "-"
450     else:
451         sgn = "+"
452

```

```

453         self.azimuth = sgn+str(int(gdo))+":"+str(int(gfun))+":"+str(gbyou)
454
455     return gdo, gfun, gbyou
456
457
458     def summary(self):
459
460         sgn = "+" if self.gmm < 0 else "-"
461         gdo, gfun, gbyou = self._azimuth_rad2dms(self.gmm)
462         sys.stdout.write("Input_\n")
463         sys.stdout.write("no_\n" % (self.num) )
464         sys.stdout.write("X_survey_\n" % (self.x, self.y)
)
465         sys.stdout.write("Output_\n")
466         sys.stdout.write("lon_\n" % (self.lon, self.lat) )
467         sys.stdout.write("SupplementaryInfo_\n")
468         sys.stdout.write("lambdaa_\n" % \
469             (sgn, abs(gdo), abs(gfun), abs(gbyou)) )
470         sys.stdout.write("s_azimuth_from_north_clockwise_\n" % self.azimuth)
471         sys.stdout.write("m_\n" % self.m)
472
473
474
475
476     def commandline():
477         if len(sys.argv) != 5: # scriptname 1 + arguments 4
478             print("Error: 引数が足りません")
479             print("引数は4つ必要です。引数: {}".format(len(sys.argv)-1)) #
excluding script name
480             print("")
481             helpmessage()
482             return
483
484         inform, no, x, y = sys.argv[1], int(sys.argv[2]), float(sys.argv[3]), float(
sys.argv[4])
485         if inform == "xy":
486             obj = xy2dms(no, x, y, "ddd")
487             print(obj.output())
488             print("----summary_")
489             obj.summary()
490         elif inform == "ddd":
491             obj = dms2xy(no, x, y, "ddd")
492             y, x = obj.output()
493             print(x,y)
494             print("----summary_")
495             obj.summary()
496         elif inform == "dms":

```

```

497         obj = dms2xy(no, x, y, "dms")
498         y, x = obj.output()
499         print(x,y)
500         print("----summary")
501         obj.summary()
502     else:
503         helpmessage()
504
505 def helpmessage():
506     print("緯度経度と平面著角座標系の座標変換を河瀬ほか(2011)で実行するプログラム")
507     print("")
508     print("Usage:")
509     print("python projKawase2011.py [inform] [no] [x] [y]")
510     print("        [inform]: 入力するx,yの書式: 'xy', 'ddd', 'dms' のいずれか")
511     print("        [no]: 平面直角座標系の番号: 整数で1-19")
512     print("        [x]: x座標もしくは経度: 実数")
513     print("        [y]: y座標もしくは緯度: 実数")
514     print("")
515     print("Example:")
516     print("python projKawase2011.py ddd 9 140.0852028 36.103481")
517     print("python projKawase2011.py dms 9 1400506.73 360612.53")
518     print("python projKawase2011.py xy 9 11510.42094831361 22677.527084631052")
519
520
521 def examples():
522     """
523     python -c "import projKawase2011 as proj; proj.examples()"
524     """
525     print("1. check_dms2xyz_class")
526     print("input_coords_is_GSI_in_tsukuba")
527     num = 9
528     lon = 1400506.73
529     lat = 360612.53
530     a = dms2xy(num, lon, lat, "dms")
531     a.summary()
532     print(a.output())
533
534     print("\n")
535
536     print("reverse_process")
537     print("2. check_xy2dms_class")
538     num = 9
539     x = 11510.42094831361
540     y = 22677.527084631052
541     b = xy2dms(num, x, y, "dms")
542     b.summary()
543     print(b.output())

```

```
544
545     print("3. □check□functions")
546     print(dms2ddd(1400506.73))
547     print(dms2ddd(360612.53))
548
549     print(ddd2dms(140.0852027777778))
550     print(ddd2dms(36.10348055555556))
551
552     c = dms2xy(2, 140.0852027777778, 36.10348055555556, "ddd")
553     c.summary()
554     print(c.output())
555
556     print("END")
557
558
559 if __name__ == "__main__":
560     cmdline()
```

33.3 getindex.py

getindex.py

```
1  #!/usr/bin/env python
2  __copyright__ = "Public Works Research Institute"
3  __author__ = "Volcano and Debris flow research team"
4  __license__ = "CC-BY-SA 4.0"
5
6  import os, sys, time
7  import glob
8  import math
9  try:
10     import gdal, gdalconst
11 except ModuleNotFoundError:
12     try:
13         from osgeo import gdal, gdalconst
14     except:
15         pass
16
17
18 helpstring = '''
19 Usage: python getIndex.py xcoords ycoords
20     xcoords: 平面直角座標系におけるX座標
21     ycoords: 平面直角座標系におけるY座標
22
23 Example:
24     1. python getIndex.py 11716.1046 -8113.1567
25     2. python getIndex.py 11716.1046 -8113.1567 input\gridindex.asc
26 '''
27
28 def get_gridid(fasc="gridindex.asc", x=None, y=None):
29     src_root, ext = os.path.splitext(fasc)
30     ftif = src_root + ".tif"
31
32     if os.path.isfile(ftif):
33         os.remove(ftif)
34
35     gdal_options = ""
36     # gdal_options += "-a_srs EPSG:4612"
37     gdal_options += "_ofGTiff"
38     gdal_command = 'gdal_translate%s "%s" "%s"' % (gdal_options, fasc, ftif)
39     os.system(gdal_command)
40
41     gtif = gdal.Open(ftif, gdalconst.GA_ReadOnly)
42     idx = gtif.GetRasterBand(1).ReadAsArray()
```

```

43
44
45 # image coordinate, i.e., y axis is downward direction. then dy is negative.
46 # thus, coordinate system +x-y
47 xmin, dx, xrotate, ymax, yrotate, dy = gtif.GetGeoTransform()
48 ncols = gtif.RasterXSize
49 nrows = gtif.RasterYSize
50
51 print("Input(x,y)={0},{1}".format(x, y))
52 print("(xmin,ymax)={0},{1}".format(xmin, ymax))
53 print("(dx,dy)={0},{1}".format(dx, dy))
54 print("(ncols,nrows)={0},{1}".format(ncols, nrows))
55
56 grid_x = abs( int( (x - xmin)/dx ) )
57 grid_y = abs( int( (y - ymax)/dy ) )
58
59 print("(grid_x,grid_y)={0},{1}".format(grid_x, grid_y))
60 print("")
61 print("hydro_input_grid_index={0}".format(idx[grid_y][grid_x]))
62 print(os.linesep)
63
64
65
66 if __name__ == "__main__":
67     # example Unzen 1993
68     ## 1. akamatsu tani
69     # get_gridid(fasc="gridindex.asc", x=-64364, y=-28028)
70     ## 2. mizunashi kawa
71     # get_gridid(fasc="gridindex.asc", x=-62875, y=-26388)
72     ## 3. example Furue 1991
73     # get_gridid(fasc="gridindex.asc", x=11716.1046, y=-8113.1567)
74     #
75     # Command line Usage
76     argv = sys.argv
77     print(len(argv), argv[0], argv[1], argv[2])
78     if len(argv) == 3:
79         get_gridid(x=float(argv[1]), y=float(argv[2]))
80     elif len(argv) == 4:
81         get_gridid(fasc=argv[3], x=float(argv[1]), y=float(argv[2]))
82     else:
83         print("Error引数")
84         print(helpstring)

```

33.4 mkvtk.py

mkvtk.py

```
1  # -*- coding: utf-8 -*-
2  __copyright__ = "Public Works Research Institute"
3  __author__ = "Volcano and Debris flow research team"
4  __license__ = "CC BY-SA 4.0"
5
6  import os
7  import sys
8  import glob
9  import math
10 import numpy as np
11
12 # VTK CLASS
13 ## for visualizing time series results: map_*.csv
14
15 class makeLegacyVTK:
16
17     def __init__(self):
18         self.vtkstring = ""
19
20     def header(self, name):
21         moji = "#_vtk_DataFile_Version_1.0" + os.linesep
22         moji += "{}".format(name)
23         moji += os.linesep
24         moji += "ASCII"
25         moji += os.linesep
26         self.vtkstring = moji
27
28     def grid_header(self, num, dtype="float"):
29         moji = "DATASET_UNSTRUCTURED_GRID" + os.linesep
30         moji += "POINTS_{}_{}".format(num, dtype)
31         moji += os.linesep
32         self.vtkstring += moji
33
34     def grid_xyz(self, x,y,z):
35         moji = ""
36         for xi, yi, zi in zip(x,y,z):
37             xyzform = "{:.4f}_ {:.4f}_ {:.4f}".format(xi, yi, zi)
38             moji += xyzform
39             moji += os.linesep
40         moji += os.linesep
41         moji += "POINT_DATA_{}".format(len(x))
42         moji += os.linesep
```

```

43         self.vtkstring += moji
44
45     def scalar(self, data, name, dtype="float"):
46         moji = "SCALARS_{ }_{ }".format(name, dtype)
47         moji += os.linesep
48         moji += "LOOKUP_TABLE_default" + os.linesep
49         for d in data:
50             moji += "{:.4f}".format(d)
51             moji += os.linesep
52         self.vtkstring += moji
53
54     def vector(self, u, v, w, name, dtype="float"):
55         moji = "VECTORS_{ }_{ }".format(name, dtype)
56         moji += os.linesep
57         try:
58             for ui, vi, wi in zip(u,v,w):
59                 moji += "{:.4f}_{ }{:.4f}_{ }{:.4f}".format(ui, vi, wi)
60                 moji += os.linesep
61             self.vtkstring += moji
62         except ZeroDivisionError as e:
63             print(e)
64             print("_not_normalization_of_vector")
65
66     def vectornormscalar(self, u, v, w, name, dtype="float"):
67         moji = "SCALARS_{ }_{ }".format(name+"_scalar", dtype)
68         moji += os.linesep
69         moji += "LOOKUP_TABLE_default"
70         moji += os.linesep
71         try:
72             for ui, vi, wi in zip(u,v,w):
73                 norm = math.sqrt(ui*ui + vi*vi + wi*wi)
74                 moji += "{:.4f}".format(norm)
75                 moji += os.linesep
76             self.vtkstring += moji
77         except ZeroDivisionError as e:
78             print(e)
79             print("_not_normalization_of_vector")
80
81     def output(self):
82         return self.vtkstring
83
84
85     # WRITE SCALAR
86
87     def mkvtk_scalar(fname, col=7):
88         try:
89             if isinstance(col, int):

```



```

90         # colが整数の場合
91         pass
92     elif isinstance(col, str):
93         # colが文字列の場合
94         col = scalar[col]
95 except TypeError as e:
96     print(e)
97     return 0
98
99 with open(fname) as fin:
100     names = fin.readline().replace(os.linesep, "").split(",")
101     mat = np.loadtxt(fname, delimiter=",", skiprows=1)
102     ndata, ndkinds = mat.shape
103     x = mat[:,1]
104     y = mat[:,2]
105     z = mat[:,5]
106
107     names[col] = names[col].strip()
108     print("processing...", names[col]+"_"+fname+".vtk")
109
110     vtk = makeLegacyVTK()
111     vtk.header(fname)
112     vtk.grid_header(ndata)
113     vtk.grid_xyz(x,y,z)
114     vtk.scalar(mat[:,col], names[col])
115     vtkstring = vtk.output()
116
117     with open(names[col]+"_"+fname+".vtk", "w") as fout:
118         fout.write(vtkstring)
119
120
121 def map2vtk_scalar(url="map_*.csv", col=7):
122     files = glob.glob(url)
123     try:
124         if isinstance(col, int):
125             # colが整数の場合
126             for f in files:
127                 mkvtk_scalar(f, col)
128         elif isinstance(col, str):
129             # colが文字列の場合
130             col = scalar[col]
131             for f in files:
132                 mkvtk_scalar(f, col)
133     except TypeError as e:
134         print(e)
135     return 0
136

```

```

137 # WRITE VECTOR
138
139 def mkvtk_vector(fname, col=(10,11)):
140     try:
141         if isinstance(col[0], int):
142             # colが整数の場合
143             pass
144         elif isinstance(col[0], str):
145             # colが文字列の場合
146             col = vectors[col]
147     except TypeError as e:
148         print(e)
149         return 0
150
151     with open(fname) as fin:
152         names = fin.readline().replace(os.linesep, "").split(",")
153     mat = np.loadtxt(fname, delimiter=",", skiprows=1)
154     ndata, ndkinds = mat.shape
155     x = mat[:,1]
156     y = mat[:,2]
157     z = mat[:,5]
158
159     dataname = names[col[0]].strip() + "_" + names[col[1]].strip()
160     outname = dataname + "_" + fname + ".vtk"
161     print("processing...□", outname)
162
163     vtk = makeLegacyVTK()
164     vtk.header(fname)
165     vtk.grid_header(ndata)
166     vtk.grid_xyz(x,y,z)
167     u = mat[:, col[0]]
168     v = mat[:, col[1]]
169     w = np.zeros(ndata)
170     vtk.vector(u, v, w, dataname)
171     vtk.vectornormscalar(u, v, w, dataname)
172     vtkstring = vtk.output()
173
174     with open(outname, "w") as fout:
175         fout.write(vtkstring)
176
177
178 def map2vtk_vector(url="map_*.csv", col=(10,11)):
179     files = glob.glob(url)
180     try:
181         if isinstance(col[0], int):
182             # colが整数の場合
183             for f in files:

```

```

184         mkvtk_vector(f, col)
185     elif isinstance(col[0], str):
186         # colが文字列の場合
187         col = (vector[col])
188         for f in files:
189             mkvtk_vector(f, col)
190 except TypeError as e:
191     print(e)
192     return 0
193
194
195 scalar = {
196     "depth": 7,
197     "wetdry": 12,
198     "tantheta": 13,
199     "dz" : 14,
200     "cl" : 19,
201     "cf" : 20,
202     "erosionrate" : 21,
203     "cinfy" : 22
204 }
205
206 vector = {
207     "u": (8,9),
208     "q": (10,11),
209     "qbl": (15,16),
210     "qbf": (17,18)
211 }
212
213
214 if __name__ == "__main__":
215
216     # how to use mk2vtk_scalar
217     #mkvtk_scalar("map_00030.csv", col=7)
218     #map2vtk_scalar(col="depth")
219
220     # how to use mk2vtk_vector
221     # 1 file
222     #mkvtk_vector("map_00030.csv", col=(8,9))
223     # qx, qy
224     #map2vtk_vector(col=(10,11))
225     # qblx, qbly
226     #map2vtk_vector(col=(15,16))
227     # qbfx, qbfy
228     #map2vtk_vector(col=(17,18))
229     #map2vtk_vector(col=vector["u"])
230

```

```
231     # how to use one line
232     #python -c "from mkvtk import *; map2vtk_scalar(col='cinfy ')"
233     #python -c "from mkvtk import *; map2vtk_vector(col='qbf ')"
234
235     for key in scalar:
236         map2vtk_scalar(col=key)
237     for key in vector:
238         map2vtk_vector(col=key)
```

33.5 Makefile

Makefile

```
1 #
2 #FCOMP = flang
3 #FCOMP = ifort
4 FCOMP = gfortran
5
6 # for mac
7 #LIB = -L/Library/Developer/CommandLineTools/SDKs/MacOSX.sdk/usr/lib/
8
9 FFLAGS = -O3 -fmax-stack-var-size=10485760 -Wall
10 # FFLAGS = -O3 -v -Wall
11 # for parallel computing ugin OpenMP
12 # FFLAGS = -O3 -Wall -fopenmp -fmax-stack-var-size=10485760
13 # for debug
14 #FFLAGS = -Wall -fbounds-check -Wuninitialized -ffpe-trap=invalid,zero,overflow -
    fbacktrace -g
15
16 OUTNAME = debrisflow.exe
17
18 OBJ = arrayc.o main.o errnum.o flopen.o inputdem.o inputhydr.o inputparam.o
    calset.o \
19 clear.o qmflux.o qnflux.o boundaryq.o hdepth.o boundaryh.o boundaryqb.o qbflux.
    o \
20 gradbed.o gradws.o cinfy.o calaii.o calcclcf.o calrho.o calbedelev.o \
21 wetdry.o velcal.o flowpower.o \
22 pickcal.o tractive.o replace.o snapshot.o outmaxvalue.o tau.o
23
24
25 .PHONY: clean
26
27 one:
28     $(FCOMP) $(FFLAGS) *.f90 -o $(OUTNAME) $(LIB)
29
30
31 all: $(OBJ)
32 #for macOS
33 # $(FCOMP) $(FFLAGS) -o $(OUTNAME) $(OBJ) $(LIB)
34 #for windows
35     $(FCOMP) $(FFLAGS) -o $(OUTNAME) $(OBJ)
36
37 arrayc.o:
38     $(FCOMP) $(FFLAGS) -c arrayc.f90
39
```

```
40 main.o:
41     $(FCOMP) $(FFLAGS) -c main.f90
42
43 flopen.o:
44     $(FCOMP) $(FFLAGS) -c flopen.f90
45
46 inputdem.o:
47     $(FCOMP) $(FFLAGS) -c inputdem.f90
48
49 inputhydr.o:
50     $(FCOMP) $(FFLAGS) -c inputhydr.f90
51
52 inputparam.o:
53     $(FCOMP) $(FFLAGS) -c inputparam.f90
54
55 calset.o:
56     $(FCOMP) $(FFLAGS) -c calset.f90
57
58 clear.o:
59     $(FCOMP) $(FFLAGS) -c clear.f90
60
61 qmflux.o:
62     $(FCOMP) $(FFLAGS) -c qmflux.f90
63
64 qnflux.o:
65     $(FCOMP) $(FFLAGS) -c qnflux.f90
66
67 boundaryq.o:
68     $(FCOMP) $(FFLAGS) -c boundaryq.f90
69
70 hdepth.o:
71     $(FCOMP) $(FFLAGS) -c hdepth.f90
72
73 boundaryh.o:
74     $(FCOMP) $(FFLAGS) -c boundaryh.f90
75
76 qbflux.o:
77     $(FCOMP) $(FFLAGS) -c qbflux.f90
78
79 boundaryqb.o:
80     $(FCOMP) $(FFLAGS) -c boundaryqb.f90
81
82 gradbed.o:
83     $(FCOMP) $(FFLAGS) -c gradbed.f90
84
85 gradws.o:
86     $(FCOMP) $(FFLAGS) -c gradws.f90
```

```
87
88  cinfty.o:
89      $(FCOMP) $(FFLAGS) -c cinfty.f90
90
91  calclcf.o:
92      $(FCOMP) $(FFLAGS) -c calclcf.f90
93
94  calrho.o:
95      $(FCOMP) $(FFLAGS) -c calrho.f90
96
97  calbedelev.o:
98      $(FCOMP) $(FFLAGS) -c calbedelev.f90
99
100 calaii.o:
101     $(FCOMP) $(FFLAGS) -c calaii.f90
102
103 wetdry.o:
104     $(FCOMP) $(FFLAGS) -c wetdry.f90
105
106 velcal.o:
107     $(FCOMP) $(FFLAGS) -c velcal.f90
108
109 flowpower.o:
110     $(FCOMP) $(FFLAGS) -c flowpower.f90
111
112 pickcal.o:
113     $(FCOMP) $(FFLAGS) -c pickcal.f90
114
115 tractive.o:
116     $(FCOMP) $(FFLAGS) -c tractive.f90
117
118 replace.o:
119     $(FCOMP) $(FFLAGS) -c replace.f90
120
121 snapshot.o:
122     $(FCOMP) $(FFLAGS) -c snapshot.f90
123
124 outmaxvalue.o:
125     $(FCOMP) $(FFLAGS) -c outmaxvalue.f90
126
127 tau.o:
128     $(FCOMP) $(FFLAGS) -c tau.f90
129
130 errnum.o:
131     $(FCOMP) $(FFLAGS) -c errnum.f90
132
133
```

```
134 clean :  
135     rm *.mod  
136     rm *.o  
137     rm *.exe
```

土木研究所資料
TECHNICAL NOTE of PWRI
No.4415 March 2022

編集・発行 ©国立研究開発法人土木研究所

本資料の転載・複写の問い合わせは

国立研究開発法人土木研究所 企画部 業務課
〒305-8516 茨城県つくば市南原1-6 電話029-879-6754