



HAL
open science

Vers une représentation parcimonieuse de la variabilité morphologique des rivières non-jaugées adaptée au problème inverse hauteur-débit

Mounir Mahdade

► **To cite this version:**

Mounir Mahdade. Vers une représentation parcimonieuse de la variabilité morphologique des rivières non-jaugées adaptée au problème inverse hauteur-débit. Hydrologie. Sorbonne Université, 2022. Français. NNT : 2022SORUS168 . tel-03813516

HAL Id: tel-03813516

<https://theses.hal.science/tel-03813516v1>

Submitted on 13 Oct 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Sorbonne Université

École Doctorale 398 : Géosciences, Ressources Naturelles et
Environnement

Laboratoire : UMR METIS

Vers une représentation parcimonieuse de la variabilité morphologique des rivières non-jaugées adaptée au problème inverse hauteur-débit

Par: Mounir Mahdade

Thèse de doctorat d'hydrologie

Présentée et soutenue publiquement le 08 juillet 2022

Dirigée par: Nicolas Le Moine et Pierre Ribstein

Devant un jury composé de :

Mme Marie-George TOURNOUD	Université de Montpellier, France	Rapportrice
M. Pierre-Olivier MALATERRE	INRAE, Montpellier, France	Rapporteur
Mme Hélène ROUX	INP-IMFT Toulouse, France	Examinatrice
M. Pierre BRIGODE	Université Côte d'Azur, France	Examinateur
Mme Agnès DUCHARNE	CNRS-UMR METIS, Paris, France	Examinatrice
M. Roger MOUSSA	INRAE-UMR LISAH, Montpellier, France	Examinateur

Vers une représentation parcimonieuse de la variabilité morphologique des rivières non-jaugées adaptée au problème inverse hauteur-débit

Auteur:
Mounir Mahdade

Encadrement par:
Nicolas Le Moine et Pierre Ribstein

Sorbonne Université, UMR Milieux Environnementaux, Transferts et Interactions dans les Hydrosystèmes et les Sols (METIS)

Case 105 – Tour 46/56 – 3ème étage – 4 place Jussieu – 75252 Paris Cedex 05 – France

Remerciements

Je dois ma gratitude à toutes les personnes qui ont rendu cette thèse possible et grâce à qui mon expérience de doctorat restera gravée dans ma mémoire.

Je remercie tout d'abord mon directeur de thèse Pierre Ribstein pour le suivi et les conseils depuis la préparation de l'audition de l'école doctorale et durant mes 4 ans de thèse. Ma plus profonde gratitude va à mon encadrant et à mon co-directeur de thèse Nicolas Le Moine pour l'opportunité qu'il m'avait donné depuis le stage jusqu'à la thèse, pour l'encadrement, le suivi et les connaissances scientifiques. J'ai eu la chance inouïe d'avoir un encadrant qui m'a aidé avec sa patience et son soutien à surmonter de nombreuses situations d'échec et à terminer cette thèse.

Mon comité de thèse m'a guidé tout au long de ces années. Merci à Andréassian Vazken, François Bourgin et Roger Moussa d'avoir été mes principaux conseillers, pour leur implication dans le suivi de mes travaux pendant ces 4 ans avec leurs encouragements et suggestions appréciées. Merci également à tous les membres du Jury de thèse pour leurs critiques constructives et leurs remarques réfléchies. Je tiens à remercier Hélène Roux, Pierre Brigode, Agnès Ducharne et Roger Moussa pour avoir examiné et discuté avec soin mon travail, ainsi que Marie-George Tournoud et Pierre-Olivier Malaterre pour avoir lu et rapporté avec soin le manuscrit de ma thèse.

J'apprécie profondément l'équipe d'enseignement de METIS à savoir Ludovic Oudin, Valérie Plagnes, Nicolas Le Moine et Jamel Benmamar qui m'ont toujours aidé et facilité la tâche d'enseignement pendant ma thèse et mon ATER.

Je suis également reconnaissant au personnel administratif et informatique de METIS, de l'école doctorale GRNE et de l'UFR TEB, Bénédicte Sabatier, Sylvain Théry, Aurelien Baro, Valérie Girard, Isabelle Bauwe et Laurence Pastor.

Tout au long de cette aventure, j'ai rencontré beaucoup de doctorants, ingénieurs et des chercheurs qui m'ont aidé à passer un bon moment au sein de METIS. D'abord, je commence avec les doctorants du fameux bureau 410. Je remercie Noëlie qui n'a jamais hésité à me fournir des conseils et de l'aide même si son travail est loin de l'hydro. Je remercie Ardalan et Ningxin pour les nombreuses discussions culturelles et les repas de midi à la cantine avec Diane Diane que je considère comme une 410ienne. Je remercie encore les nouveaux arrivants Pierre, Thibault, Nouradine et Morgane pour leurs aides même en période de COVID. Je suis aussi reconnaissant d'avoir eu la chance de connaître Axel lors de mon Master 2 et avec lui, j'ai partagé des projets, des discussions intéressantes sur notre futur et en ce moment d'écriture de ces phrases, il est en train de préparer sa soutenance. Bon courage !

Je tiens à exprimer également ma gratitude pour Sylvain de m'avoir toujours aidé et conseillé, pour Marc de m'avoir donné des encouragements lors de mon premier comité de thèse, pour Vincent de m'avoir bien accueilli dans son bureau et de me répondre à chaque fois que j'avais besoin de quelque chose. Je remercie le fameux Momo "le petit cochon" pour le soutien, l'aide dans la recherche et l'enseignement, toutes les discussions philosophiques et les MEMES qu'on a partagé.

Je remercie Gurpreet, Pedro, Radegonde, Antoine, Maxime, Lucia, Etienne et Alexandre pour les discussions de midi sur tout et n'importe quoi. Je remercie aussi les autres personnes agréables de METIS qui sont là maintenant ou qui ont quitté le labo : Salma, Paul, Audrey, Thomas, Marie, Amen, Flore, Manon et Xingcheng. Je vous souhaite bon courage dans la suite !

Finalement, je remercie mes amis Elhoucine et Marouane pour leur soutien et leur attention. Je suis particulièrement redevable à ma femme Laila qui m'a aidé à rester sain d'esprit pendant ces années difficiles, Merci pour ses conseils, ses sacrifices, son aide et nos discussions sur tous les sujets. Je suis également reconnaissant à mes parents de tout mon cœur et de toute mon âme. Ma famille, à qui cette thèse est dédiée, a été une source constante d'amour, de préoccupation, de soutien et de force pendant toutes ces années. Je tiens à exprimer ma sincère gratitude à ma famille.

... À ma grande mère Fatima qui nous a quittés !

Résumé

L'absence de mesures *in situ* dans les rivières non-jaugées empêche la construction de courbes de tarage, utiles à plusieurs applications hydrologiques et hydrauliques. Ces dernières décennies, l'idée d'estimer les débits par des méthodes de télédétection a émergé, toujours sur le principe de construire un lien entre hauteur d'eau et débit. Cependant, ce changement s'accompagne d'un changement d'échelle de la mesure de hauteur, qui n'est plus rattachée à une section mais au tronçon, conduisant à la notion de *courbe de tarage au tronçon* sous les mêmes hypothèses qu'une courbe de tarage à la section (notamment de stationnarité). Cette thèse aborde la question de la construction d'une telle courbe.

Etant donné que les paramètres de friction, de la bathymétrie du lit mineur et du débit sont inconnus, la première étape est de réduire la dimensionnalité du problème. Une étude hydro-morphologique sur quelques tronçons de rivières montre que la variabilité géométrique des cours d'eau naturels (ex. alternances des mouille-radier) peut être représentée sous forme d'un modèle périodique 2D dont la forme en plan est basée sur une courbe de Kinoshita, permettant de réduire le nombre d'inconnues.

Afin de tester et de valider ce modèle, la deuxième étape est consacrée à la production d'une simulation de référence 2D sur un tronçon de 40km de la Garonne entre Toulouse et la station hydrométrique de Verdun-sur-Garonne, avec une topographie continue de haute résolution en lit mineur et majeur. La surface libre simulée peut être considérée comme un jeu de "pseudo-observations" similaires à celles qui seront produites par la mission SWOT (*Surface Water and Ocean Topography*).

Dans la troisième étape, nous construisons un modèle hydraulique direct 2D et une procédure d'inversion associée. Le modèle direct se base sur une simplification géométrique non-uniforme (modèle périodique) et un solveur des équations de Saint-Venant (le modèle Basilisk). Une inversion stochastique par algorithme génétique permet d'estimer la courbe de tarage tronçon dans un régime stationnaire en testant les paramètres géométriques et de frottements qui reconstituent au mieux les signatures observées (courbes caractéristiques calculées directement à l'échelle du tronçon).

Les résultats préliminaires de l'inversion ont montré que le modèle périodique arrive à bien identifier les paramètres morphologiques du tronçon, mais pas les coefficients de frottements (problème d'équifinalité). Un *a priori* sur les coefficients de frottements pourrait mieux encadrer la courbe de tarage au tronçon résultante.

Mots-clés: rivières non-jaugées, courbe de tarage, inversion, morphologie, simulation de crue, modèle hydraulique.

Abstract

The lack of in situ measurements in ungauged rivers prevents the construction of rating curves, useful for several hydrological and hydraulic applications. In recent decades, the idea of estimating discharges by remote sensing methods has emerged, based on the principle of constructing a link between water elevation and discharge. However, this change is accompanied by a change in the scale of the elevation measurement, which is no longer attached to a cross-section but to the reach, leading to the notion of a reach-average rating curve under the same assumptions as a cross-section rating curve (notably stationarity). This thesis addresses the question of the construction of such a curve.

Since the parameters of friction, channel bed bathymetry and discharge are unknown, the first step is to reduce the dimensionality of the problem. A hydro-morphological study on some river reaches shows that the geometrical variability of natural streams (e.g. alternating pool-riffle) can be represented in the form of a 2D periodic model whose plan form is based on a Kinoshita curve, allowing to reduce the number of unknowns.

In order to test and validate this model, the second step is devoted to the production of a 2D reference simulation on a 40km reach of the Garonne River between Toulouse and the hydrometric station of Verdun-sur-Garonne, with a continuous high-resolution topography in the channel and floodplain. The simulated free surface can be considered as a set of "pseudo-observations" similar to those that will be produced by the SWOT mission (Surface Water and Ocean Topography).

In the third step, we build a 2D direct hydraulic model and an associated inversion procedure. The direct model is based on a non-uniform geometric simplification (periodic model) and a solver of the Saint-Venant equations (the Basilisk model). A stochastic inversion by genetic algorithm allows to estimate the reach-averaged rating curve in a stationary regime by testing the geometrical and friction parameters that best reconstitute the observed signatures (characteristic curves calculated directly at the reach scale).

Preliminary results of the inversion show that the periodic model is able to identify the morphological parameters of the reach, but not the friction coefficients. An *a priori* on the friction coefficients could better frame the resulting reach-averaged rating curve.

Keywords: ungauged rivers, reach-averaged rating curve, inversion, morphology, flood simulation, hydraulic model.

Table des matières

Remerciements	iii
Résumé	v
Abstract	vii
Liste des symboles	xiii
1 Introduction générale	1
1.1 Introduction	1
1.2 Estimation des courbes de tarage à la section	2
1.2.1 Méthodes classiques de mesure des hauteurs d'eau et des débits	2
1.2.2 Méthodes non-intrusives de mesure à la section	5
1.2.3 Approches pour l'estimation des courbes de tarage à la section	5
1.2.4 La télédétection au service des rivières non-jaugées	8
1.3 Estimation des courbes de tarage au tronçon dans les rivières non-jaugées	9
1.3.1 Problème direct et inverse	9
1.3.2 Etat de l'art des méthodes inverses en utilisant des données de télédétection	10
1.3.2.1 Approches avec une paramétrisation uniforme	10
1.3.2.2 Approches avec une paramétrisation implicite non-uniforme	11
1.3.2.3 Vers des approches avec une paramétrisation explicite non-uniforme	11
1.4 Contexte et objectifs de l'étude	12
1.4.1 Modèle stationnaire spatialement périodique comme hypothèse de travail	12
1.4.2 Démarche à suivre pour tester l'approche périodique sur la Garonne	13
1.5 Plan de la thèse	15
2 Caractérisation de la variabilité morphologique des rivières	17
2.1 Introduction	17
2.2 Variabilité des unités morphologique à l'échelle du tronçon	18
1 - Introduction	18
1.1 - State-of-the-art methods for a quantitative assessment of morphological variability within a reach	20
1.2 - Study objectives	22
2 - Dataset and study reaches	23
3 - Wavelet method	24
3.1 - Wavelet analysis and ridge extraction	25
3.2 - Univariate case	26
3.3 - Multivariate case	27
4 - Results	29
4.1 - Univariate vs. multivariate	29

4.2 - Comparison with benchmark method	32
5 - Discussion	34
6 - Conclusions	35
2.3 Conclusion du chapitre	44
3 Production d’une simulation de référence sur la Garonne	45
3.1 Introduction	45
3.2 La Garonne comme tronçon d’étude	46
3.3 Construction de la bathymétrie	48
1 - Introduction	48
2 - Principles of Remotely-Sensed Estimation of River Bathymetry Using Passive Optical Measurements	50
3 - Beyond Spectral Ratios : A Definition of Multispectral Hue	52
3.1 - Rationale for the Definition of Multispectral Hue	52
3.2 - Multispectral Hue as a Directional Variable	54
3.3 - Mixture Models for the Analysis of Multispectral Hue Distribution	56
4 - Using Multispectral Hue as a Predictor for Depth	58
4.1 - Dataset	58
4.2 - Hue Sphere \mathbf{S}^2 for $n = 4$ Bands	59
4.3 - Masks	60
4.4 - Hue–Depth Visual Correlation on \mathbf{S}^2	61
4.5 - Mixture Models on \mathbf{S}^2 and Higher Dimension, and Depth Predictor	62
4.6 - Modified EM Algorithm for Depth Estimation	63
5 - Results	64
5.1 - Error Statistics	64
5.2 - Focus on Some Cross-Sections	67
6 - Discussion	68
6.1 - Sources of Error	68
6.2 - Perspectives	68
3.4 Simulation de référence de crue sur la Garonne	73
1 - Introduction	74
2 - Flood mapping model	76
2.1 - Basilisk solver	76
2.1.1 - Numerical scheme	76
2.1.2 - Adaptive mesh refinement	77
2.1.3 - Inlet with the mask feature	77
2.2 - New developments in Basilisk	80
2.2.1 - Defining an inlet without the mask feature	80
2.2.2 - Source terms implementation	84
3 - Case study: Garonne River	85
3.1 - Study reach and data	85
3.1.1 - Study reach and associated subcatchment	85
3.1.2 - Preparation of forcing inflow discharge data	87
3.1.3 - Topographic data in the Basilisk domain	90
3.2 - Calibration of the channel friction term	91
3.2.1 - First estimate of channel roughness using downstream rating curve in steady state	92
3.2.2 - Comparison / validation on the June 2002 non-overflowing flood event	95
3.3 - Calibration of floodplain friction term	98
3.3.1 - First estimate of floodplain roughness using flood marks	98
3.3.2 - Evaluation of Manning floodplain coefficient using flood dynamics	103
4 - Conclusion	105
3.5 Conclusion du chapitre	108

4	Signatures morphologiques, bathymétriques et hydrauliques au tronçon	109
4.1	Introduction	109
4.2	Signatures relatives au tracé en plan	109
4.2.1	Sinuosité	109
4.2.2	Variance angulaire	110
4.2.3	Longueur d’onde des alternances de faciès hydrauliques	113
4.2.4	Courbure maximale, rayon de courbure minimal	113
4.2.5	Voisinage tubulaire et difficulté à positionner des sections en lit majeur	113
4.3	Définition d’un niveau de plein-bord au tronçon	113
4.3.1	Analyse de la topographie sans tendance	114
4.3.2	Rayon hydraulique équivalent au tronçon	115
4.3.3	Largeur équivalente de plein-bord au tronçon	117
4.3.4	Influence d’un seuil de visibilité	117
4.4	Signatures spécifiques à la topographie en lit majeur	118
4.4.1	Principe et intérêt	118
4.4.2	Rayon hydraulique équivalent du lit majeur	118
4.5	Signatures hydrauliques en écoulement stationnaire	119
4.5.1	Signatures relatives à la cote de la surface libre	121
4.5.2	Signature hydrométrique : courbe de tarage au tronçon	122
4.6	Tableau synthétique des signatures définies	123
5	Modèle équivalent stationnaire en géométrie périodique : formulation et premiers tests d’inversion	125
5.1	Introduction	125
5.2	Construction du modèle équivalent périodique d’un tronçon	126
5.2.1	Paramétrisation de la forme en plan	126
5.2.1.1	Paramétrisation : modèle de Kinoshita à 3 paramètres	127
5.2.2	Bathy-topographie	129
5.2.2.1	Construction de coordonnées conformes (Φ)	129
5.2.2.2	Fonction largeur de plein-bord $W_{\text{bf}}(s)$, demi-largeurs de plein-bord rive gauche $W_{\text{bf}}^{(l)}(s)$ et rive droite $W_{\text{bf}}^{(r)}(s)$	131
5.2.2.3	Bathymétrie en lit mineur	133
5.2.2.4	Topographie en lit majeur	134
5.2.3	Résumé des paramètres du modèle géométrique périodique	139
5.2.4	Implémentation sous Basilisk	139
5.2.4.1	Procédure de génération de la topographie équivalente périodique	139
5.2.4.2	Approximation de l’onde diffusante	141
5.3	Modélisation inverse	141
5.3.1	Procédure d’inversion stochastique	141
5.3.1.1	Principe	141
5.3.1.2	Mise en oeuvre avec l’optimiseur CaRaMEL	143
5.3.2	Test d’inversion sur un cas synthétique	144
5.3.2.1	Simulation de référence en géométrie périodique	144
5.3.2.2	Critères d’optimisation	148
5.3.2.3	Résultats	148
5.3.3	Test d’inversion sur les simulations en régime stationnaire de la Garonne	150
5.3.3.1	Analyse des signatures à reproduire	151
5.3.3.2	Quelle variance pour la surface sans tendance η' ?	152
5.3.3.3	Résultats	155
5.4	Conclusions et perspectives	158

6 Conclusion générale	159
6.1 Modèle périodique 2D comme approche	159
6.2 Basilisk comme outil de modélisation hydraulique	161
6.3 Traitements et jeux de données produits pendant la thèse	161
Bibliographie	171
A Repères de crue de Juin 2000 de la Garonne	173
B Implémentation de l’approximation de l’onde diffusante 2D sous Basilisk	185
B.1 Dérivation de l’approximation de l’onde diffusante	185
B.2 Cas périodique en topographie sans tendance	186
B.3 Conservation de la masse	187
B.4 Implémentation Basilisk C	188
C Codes sur Basilisk	193
C.1 Injection des débits pour le calage du coefficient de Manning en lit mineur	193
C.2 Injection des hydrogrammes pour le calage du coefficient de Manning en lit majeur	201
C.3 Inversion avec Caramel	209
C.4 Fonctions développées sur Basilisk	215
C.4.1 Fonction <i>kinoshita.h</i>	215
C.4.2 Fonction <i>live_terrain.h</i>	217
C.4.3 Fonction <i>esri-binary-grids.h</i>	221
C.4.4 Fonction <i>inflow-tilt.h</i>	228
C.4.5 Fonction <i>mannig-tilt.h</i>	238

Liste des symboles

[AB]	Segment du point A au point B
a	Fraction de la demi largeur rive gauche par rapport à la largeur totale de la section
a_0	Pente transverse du lit majeur à proximité du chenal m/m
a_{max}	Valeur maximale de la fraction $a(s)$ dans un tronçon
a_∞	Pente transverse du lit majeur très loin du chenal m/m
\mathcal{A}	Section mouillée d'un canal en m^2
$A(\eta')$	Aire inondée plane pour le niveau uniforme η' de topographie sans tendance en m^2
A_{bf}	Aire inondée de plein bord en m^2
$A_{fp}(\eta')$	Aire inondée plane en lit majeur pour le niveau η' en m^2
$\tilde{A}(\bar{\eta}')$	Aire inondée en fonction de la valeur moyenne $\bar{\eta}'$
d_{AB}	Distance euclidienne entre A et B en m
D_{max}	Distance maximale en m pour atteindre un débit stable au niveau de l'injection
e	Fonction à minimiser dans le PID pour avoir le débit à injecter
\mathcal{E}	Espace des paramètres ou des scores/objectifs
f	Fonction objective
F	Flux
\mathcal{F}	Fonction de transformée de Fourier
g	Accélération de la pesanteur en m/s^2
h	Hauteur d'eau en m
$h_{HW,point}$	Hauteur d'eau (High-Water) dans un point donné en m
h_{talweg}	Profondeur maximale à chaque section (profondeur de talweg) en m
h_{pool}	Profondeur de talweg dans les mouilles en m
h_{run}	Profondeur de talweg dans les radiers en m
I_{reg}	Pente régionale du lit majeur en m/m
I_x	Pente dans la direction des x en m/m
I_y	Pente dans la direction des y en m/m
k	Nombre d'onde en rad/m (analogue spatial de la pulsation pour une analyse temporelle en rad/m)
k_0	Nombre d'onde du motif en rad/m
K_d	Gain dérivé (PID)
K_i	Gain intégral (PID)
K_p	Gain proportionnel (PID)
K_{min}	Coefficient de Strickler du lit mineur en $m^{1/3}/s$
K_{maj}	Coefficient de Strickler du lit majeur en $m^{1/3}/s$
L	Longueur du tronçon selon la direction régionale en m
\mathbf{n}	Vecteur normal sortant d'une surface
n_{ch}	Coefficient de Manning du lit mineur en $s/m^{1/3}$
n_{fp}	Coefficient de Manning du lit majeur en $s/m^{1/3}$
N	Nombre des débits testés pour calibrer le coefficient de Manning en lit mineur
\mathbf{N}	Vecteur unitaire normal à la direction du chenal et dirigé vers la rive gauche
Q	Débit d'écoulement en m^3/s
Q_{bc}	Débit d'écoulement à la frontière du domaine en m^3/s

Q_{inlet}	Débit d'écoulement au niveau de l'injection en m^3/s
$Q_{setpoint}$	Débit d'écoulement testé dans le PID en m^3/s
$Q_{response}$	Débit d'écoulement en réponse du PID en m^3/s
R	Rayon de courbure local en m
R_{min}	Rayon de courbure minimal en m
$R_h^*(\eta')$	Rayon hydraulique équivalent pour le niveau uniforme η' de topographie sans tendance en m
$R_{h,app}^*(\eta')$	Rayon hydraulique équivalent apparent pour le niveau η' en m
$R_{h,fp}^*(\eta')$	Rayon hydraulique équivalent en lit majeur pour le niveau η' en m
s	Abscisse curviligne
s_A	Abscisse curviligne d'un point A
s_B	Abscisse curviligne d'un point B
S_0	Terme source topographique
S_f	Terme source de friction
t	Temps en s
\mathbf{T}	Vecteur unitaire tangent définissant la direction du chenal
\mathbf{T}_{reg}	Vecteur moyen de la direction du chenal
\mathbf{t}_{AB}	Vecteur tangent au segment AB
u	Vitesse d'écoulement m/s
u_n	Composante normale de la vitesse d'écoulement m/s
u_t	Composante transversale de la vitesse d'écoulement m/s
u_x	Composante de la vitesse d'écoulement m/s sur l'axe des x
$u_x[.]$	Composante de la vitesse d'écoulement m/s sur l'axe des x au niveau de la cellule [.]
u_y	Composante de la vitesse d'écoulement m/s sur l'axe des y
\mathbf{u}	Vecteur de vitesse d'écoulement m/s
$V(\eta')$	Volume inondé pour le niveau uniforme η' de topographie sans tendance en m^3
V_{app}	Volume apparent sous le seuil de visibilité en m^3
$V_{fp}(\eta')$	Volume inondé en lit majeur pour le niveau η' en m^3
W	Largeur d'un canal en m
w	Coordonnée transversale distance à la ligne de talweg
W_{bf}	Largeur de plein bord en m
$W_{bf}^{(r)}$	Demi largeur de plein bord rive droite en m
$W_{bf}^{(l)}$	Demi largeur de plein bord rive gauche en m
W_{bf}	Largeur de plein bord en m
W^*_{bf}	Largeur équivalente de plein bord au tronçon en m
\mathbf{W}	Vecteur contenant les variables conservatives
x	Abscisse en m
x_{L93}	Coordonnée en x d'un point dans le système de projection Lambert 93
y	Ordonnée en m
y_{L93}	Coordonnée en y d'un point dans le système de projection Lambert 93
\mathbf{z}	Point (x,y) dans un plan complexe: $\mathbf{z} = x + iy$
\mathbf{z}'	Résidu à la tendance régionale ou divagation local du chenal par rapport à la tendance régionale
\mathbf{z}_{reg}	Tendance régionale rectiligne de la direction du chenal
z_b	Cote du fond du canal en m
$z_{b,point}$	Cote topographique dans un point donné en m
z_b'	Cote du fond du canal débarrassée de la pente régionale en m
$z_b[.]$	Cote du fond dans la cellule [.] en m
α_f	Coefficient d'aplatissement (flattening)
α_s	Coefficient d'asymétrie (skewness)
Δt	Intervalle de temps en s
ρ	Densité de l'eau en Kg/m^3

τ_x	Contrainte de cisaillement en Pa sur l'axe des x
τ_y	Contrainte de cisaillement en Pa sur l'axe des y
η	Cote de la surface libre en m
η'	Cote de la surface libre débarrassée de la pente régionale en m
η'_{bf}	Niveau η' maximisant $R *_{h}(\eta')$ en m
η'_{BF}	Composante de η' basse fréquence en m
η'_{facies}	Composante de η' liée aux mouilles et radiers en m
η'_{vis}	Niveau d'eau de seuil visibilité sur la topographie dépouillée de sa tendance en m
$\bar{\eta}'$	Valeur moyenne de η' en m
η'_{ini}	Cote de la surface libre initiale débarrassée de la pente régionale en m
$\eta'[\cdot]$	Cote de la surface libre dans la cellule $[\cdot]$ en m
η_{bc}	Cote de la surface libre à la frontière du domaine en m
$\eta_{HW,point}$	Cote d'eau (High-Water) dans un point donné en m
η_{sim}	Cote de la surface libre simulée en m
η_{obs}	Cote de la surface libre observée en m
λ	Longueur d'onde des alternances de faciès hydrauliques en m
θ	Angle de la direction locale du chenal par rapport à l'axe des x en rad
θ_0	Déviations angulaire maximale par rapport à la direction moyenne en rad
θ_{reg}	Angle de la tendance régionale en rad
σ	Sinuosités d'un bief
σ_{θ}	Variance / écart-type angulaire de la direction du chenal en rad
$\sigma_{\eta'}$	Ecart-type de η' par rapport à sa moyenne $\bar{\eta}'$
σ_{instru}	Ecart-type liée à l'erreur de l'instrument
ϵ	Différence entre la cote d'eau observée et simulée dans le repère en m
γ	Courbure locale signée
γ_{max}	Maximum en valeur absolue sur tout le chenal de la courbure
Φ	Coordonnée transverse conforme
Φ_{min}	Valeur minimale de $ \Phi $ localisée dans le côté intérieur des méandres
Φ_c	Coordonnée transverse conforme de la transition en lit majeur proche et loin du chenal
Φ_{bf}	Coordonnée transverse conforme de plein bord
Φ_{LB}	Coordonnée transverse conforme de la rive gauche
Φ_{RB}	Coordonnée transverse conforme de la rive droite

Chapitre 1

Introduction générale

1.1 Introduction

Les rivières constituent une composante majeure du cycle de l'eau continental. Il s'agit de la principale source d'eau de surface pour l'agriculture, la production d'énergie hydroélectrique, l'eau potable, le transport et les loisirs (*Oki et Kanae, 2006*). La connaissance des débits, de la vitesse et du temps de propagation des écoulements des rivières est essentielle pour la prévision et le suivi des inondations, l'exploitation des réservoirs et la modélisation des bassins versants (*Smith et Pavelsky, 2008*). Le moyen le plus précis pour mesurer ces grandeurs relatives à l'état de la rivière est la station hydrométrique (*Buchanan et Somers, 1969*). Dans ce type de station, les enregistrements des hauteurs d'eau accompagnés généralement de campagnes de mesure des vitesses d'écoulement sur une section mouillée permettent de déterminer le débit de l'écoulement. Le débit est le volume d'eau passant à travers une section d'un cours d'eau pendant une unité de temps. Il est défini par :

$$Q = \frac{\text{Volume}}{\Delta t} \quad (1.1.1)$$

Où Q le débit en m^3/s , Volume en m^3 et Δt le temps en s.

Les mesures¹ de débit s'effectuent périodiquement et permettent de construire une courbe dite de tarage liant chaque débit à une hauteur d'eau (ou une cote d'eau) (Figure 1.1). Cette courbe est propre à chaque site de mesure.

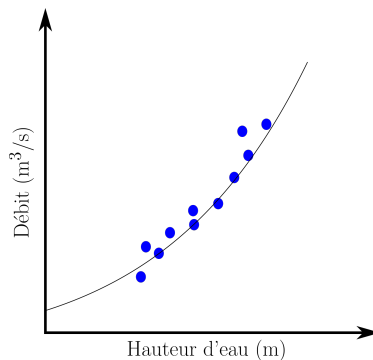


Figure 1.1: Courbe de tarage ou relation hauteur-débit. Les points bleus sont les mesures et la courbe en noir représente un ajustement de la courbe de tarage.

¹ce sont des "estimations" ou des mesures indirectes mais on préfère garder le mot "mesures" pour la simplicité

Le débit d'écoulement dépend de plusieurs caractéristiques liées au cours d'eau (pente, largeur, hauteur d'eau, etc.) et au bassin versant (taille, précipitation, infiltration, etc.). Notons qu'en hydrologie, l'estimation du débit repose sur les modèles hydrologiques (pluie-débit) qui utilisent des données de précipitation, d'évapotranspiration, de stockage, etc. pour estimer le débit (ruissellement au niveau de l'exutoire) (*Bates et al.*, 2010; *Häfliger et al.*, 2015; *Lin et al.*, 2019; *Papa et al.*, 2012; *Blöschl et Sivapalan*, 1995; *Oudin et al.*, 2010; *Young*, 2006). Cette thèse se focalise sur l'estimation des courbes de tarage à l'aide des variables géométriques et hydrauliques et non pas l'estimation des débits à l'échelle du bassin versant.

1.2 Estimation des courbes de tarage à la section

Dans les stations de jaugeage classiques, le débit repose sur l'existence d'une relation connue univoque entre la hauteur d'eau h et le débit $Q = f(h)$ (Figure 1.1) à une section donnée. Cette relation est déterminée par des contrôles hydrauliques (*Le Coz et al.*, 2014). Il existe deux types de contrôles : le contrôle à la section dans lequel la relation hauteur-débit est déterminée à partir de la géométrie de la section de contrôle. Cette section peut être une section naturelle (par exemple, au niveau d'un radier), artificielle (par exemple, déversoir) ou toute autre section rectiligne capable de maintenir une relation relativement stable entre la hauteur d'eau et le débit. Le contrôle par tronçon est un contrôle dans lequel la courbe de tarage est gouvernée par la géométrie et la résistance de l'écoulement du chenal de l'amont à l'aval (lit mineur et lit majeur). Les sections de contrôle ne sont souvent efficaces qu'aux faibles débits et sont complètement submergés par le contrôle du tronçon aux débits moyens et élevés (*Rantz*, 1982). Dans les deux cas, la courbe de tarage résultante est une relation dite "à la section". Les mesures des hauteurs et des débits (jaugeage) sont effectuées pour différents régimes de débit dans les conditions possibles de mesure. Dans le cas d'étiage et de crue, une extrapolation de la relation est nécessaire.

La relation univoque entre la hauteur d'eau et le débit à la section est basée sur des mesures. Pour acquérir plusieurs jaugeages donnant des couples hauteur-débit, nous présentons plusieurs méthodes dans la section suivante (1.2.1).

1.2.1 Méthodes classiques de mesure des hauteurs d'eau et des débits

L'estimation d'une courbe de tarage à la section repose sur des mesures du couple (hauteur, débit). En effet, il existe deux types de mesure de la hauteur d'eau :

- mesures par les limnimètres : c'est la plus ancienne méthode. Elle repose sur la lecture humaine de la hauteur de la surface libre. L'outil de mesure est une échelle ou tige graduée en métal, placée verticalement, et permettant la lecture directe de la hauteur d'eau à la station (échelle limnimétrique) ;
- mesures par les limnigraphes : la méthode repose sur l'enregistrement automatique de la hauteur d'eau à l'aide d'un capteur. Les principaux types des limnigraphes sont : limnigraphes à flotteur, à pression ou "bulle à bulle" et électroniques.

Les outils traditionnels utilisés pour le jaugeage (débit) d'une section sont multiples. La méthode capacitive (volumétrique) consiste à mesurer le temps de remplissage par l'écoulement de la rivière d'un récipient d'un volume déterminé et d'en déduire le débit selon la définition et l'équation 1.1.1. Cette méthode est plus précise, mais uniquement pour les faibles débits. La méthode par dilution (chimique) consiste à injecter en une section à l'amont d'un cours d'eau, une solution d'une concentration connue, et de suivre la concentration du traceur à l'aval en déduisant le débit par conservation de la masse. Cette méthode est très pratique dans les conditions d'écoulement torrentiel. Elle nécessite moins de matériel et donne une précision très satisfaisante (*Rantz*, 1982). Par ailleurs, la méthode hydraulique nécessite des ouvrages bien déterminés et calibrés de type déversoir. Le débit Q est donc obtenu à partir de la hauteur d'eau h mesurée. La précision de cette méthode dépend de la qualité des mesures, du calibrage de l'ouvrage et aussi des conditions réelles lors de la mesure.

Les méthodes d'exploration du champ de vitesse sont les plus répandues. Elles consistent à déterminer la

vitesse de l'écoulement en différents points de la section, tout en mesurant la surface de la section mouillée. Notons que le débit se calcule en multipliant la surface d'une section donnée du lit par la vitesse d'écoulement de l'eau dans la même section. Il s'agit d'intégrer la vitesse \mathbf{u} de l'écoulement sur la surface \mathcal{A} de la section :

$$Q = \iint_{\mathcal{A}} \mathbf{u} \cdot \mathbf{n} d\mathcal{A}$$

Avec \mathbf{n} la normale sortante à la section d'une surface totale \mathcal{A} . Les techniques de mesure des vitesses nécessitent un matériel spécifique (moulinet, perche, saumon, courantomètre, etc.) et un personnel formé à son utilisation. Dans le même objectif d'exploration du champ de vitesse de la section, les vélocimètres acoustiques et électro-magnétiques (par exemple, ADCP) offrent une analyse Doppler des ondes acoustiques émise par l'appareil dans l'eau et réfléchi par des particules rétrodiffusantes portées par le courant (pour plus de détails voir par exemple *Simpson, 2001*).

Tous ces capteurs sont intrusifs, c'est-à-dire que l'appareil de mesure doit être partiellement ou totalement immergé dans l'eau. Toutefois, l'élaboration et l'actualisation des courbes de tarage à la section avec ces méthodes traditionnelles comportent plusieurs limites :

- la courbe de tarage est une relation bijective entre la hauteur et le débit, alors qu'il peut exister un effet d'hystérésis entre la montée et la descente de crue (Figure 1.2) ;
- le lit d'une rivière n'est pas stable et uniforme. De ce fait, il est nécessaire de mettre à jour régulièrement la courbe de tarage ce qui rend le coût des campagnes de mesures très élevé ;
- les mesures en périodes des événements extrêmes (étiages et crues) sont difficiles voire parfois dangereuses à obtenir.

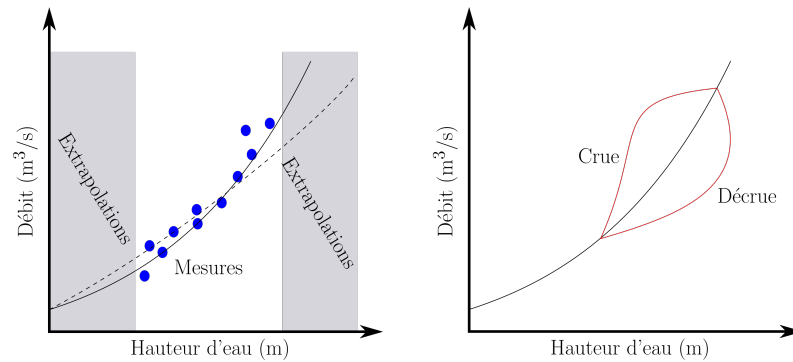


Figure 1.2: À gauche, exemple d'extrapolation douteuse de la courbe de tarage. Les points bleus sont les mesures des couples hauteur-débit. Aux faibles et aux forts débits, l'extrapolation donne d'autres courbes (traits noirs). À droite, un exemple d'hystérésis dans un régime non permanent.

Face à ces limites, on est souvent contraint de se contenter d'extrapolations des courbes de tarage qui reposent sur des bases hydrauliques et sur la topographie du site (Figure 1.2). Cela induit de fortes incertitudes sur la courbe de tarage en périodes d'étiage, où la géométrie est très instable, et dans les périodes de crue, où l'extrapolation est douteuse et des écoulements complexes peuvent intervenir, notamment en cas de débordement.

Dans certains cas, le manque de personnel oblige à abandonner des sites de mesures essentiels et nécessaires. En effet, d'après le GRDC (*Global Runoff Data Centre*), le nombre de stations hydrométriques fonctionnelles est en déclin depuis les années 1990 et il est actuellement inférieur à 1000 stations (Figure 1.3). Ces réseaux de stations jaugées sont peu nombreux et leur distribution spatiale est hétérogène à la surface du globe, d'autant plus dans les pays en développement. La Figure 1.4 illustre cette répartition disparate de ces

stations sur Terre. Ce déséquilibre s'explique, entre autres, par le coût de maintenance requis, des problèmes hydro-politiques (dans le cas des rivières transfrontalières), etc. La question du manque criant de données est un véritable problème pour appréhender l'estimation des courbes de tarage dans les rivières non-jaugées.

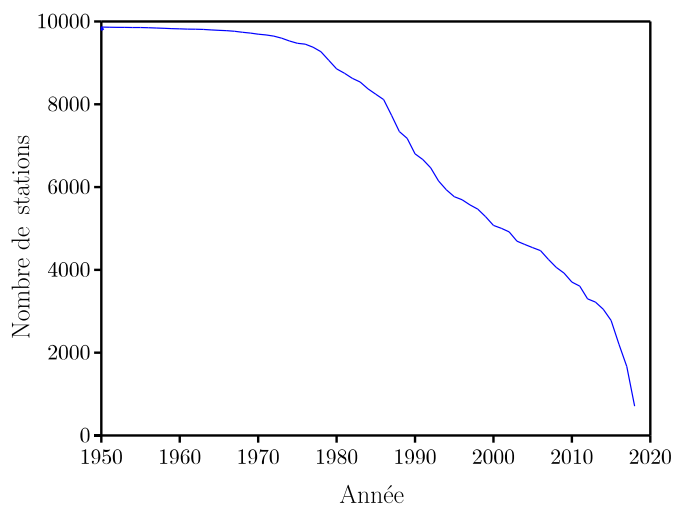


Figure 1.3: Nombre de stations hydrométriques existantes et qui produisent des données depuis 1950 jusqu'à 2018 (données prises à partir de la base de données GRDC)

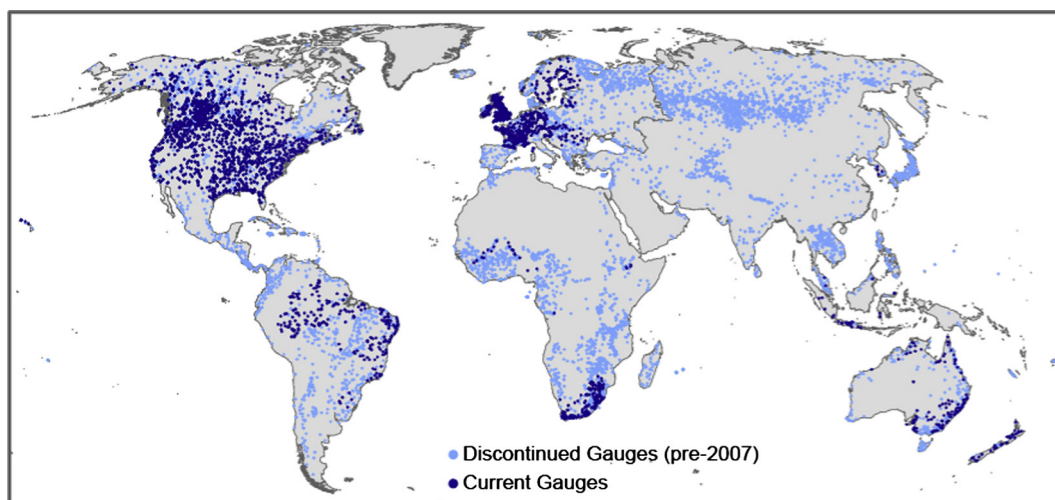


Figure 1.4: Carte de l'emplacement des stations jaugées à partir des données de GRDC. Le bleu clair indique des données discontinues (aucune donnée depuis 2007), tandis que le bleu foncé indique des stations jaugées avec des données de débits relativement récentes (*Pavelsky et al.*, 2014).

Vu ces problèmes et ces limites, de nombreuses méthodes non-intrusives ont été proposées pour répondre aux contraintes mentionnées précédemment que ce soit dans les rivières jaugées ou non-jaugées.

1.2.2 Méthodes non-intrusives de mesure à la section

Pour améliorer le jaugeage à la section des tronçons, plusieurs méthodes de mesure alternatives aux méthodes traditionnelles ont été mises au point. Ces méthodes, non-intrusives, explorent la bathymétrie et la hauteur d'eau. Parmi celles-ci, nous pouvons trouver les radars pénétrant au sol (*Ground Penetrating Radar*, *Haeni et al.*, 2000). Ces radars sont utilisés normalement dans la géophysique. Elles fonctionnent en transmettant une impulsion électromagnétique avec une fréquence porteuse dans la gamme des MHz (pour l'eau) vers le fond de la rivière à partir d'une antenne de transmission à la surface de l'eau. La réflexion de cette impulsion est captée par une antenne réceptrice. En revanche, cette méthode nécessite d'avoir les antennes au-dessus de la surface de l'eau et de balayer verticalement. Elle n'est pas utilisable dans les environnements estuariens et pendant les faibles débits dans les eaux fluviales hautement conductrices et qui ne sont pas diluées (*Rantz*, 1982).

Les capteurs ultrason sont aussi utilisés pour la mesure de la cote de la surface libre en utilisant le temps de propagation de l'onde sonore émise par le capteur. Les distancemètres optiques sont basés sur le même principe des capteurs à ultrason, mais avec une onde lumineuse. Ils donnent également des hauteurs d'eau avec une bonne précision (*Hauet*, 2006).

D'autres méthodes mesurent le champ de vitesse de la surface libre comme les radars Doppler (*Fulton et Ostrowski*, 2008). Ces derniers mesurent la vitesse des particules en surface de l'eau en utilisant l'effet Doppler (décalage en fréquence de l'onde électromagnétique réfléchie, fonction de la vitesse). De la même façon, les LiDARs Doppler mesurent la vitesse de la surface libre, mais dans un domaine spectral différent des radars : le radar fonctionne dans le domaine des ondes radio ; le LiDAR couvre le domaine de la lumière visible, et également les domaines ultra-violet et infra-rouge. Une étude sur la Cowlitz River (USA) a été menée en utilisant le GPR et le radar Doppler en parallèle pour obtenir, depuis un hélicoptère, une mesure de la bathymétrie et des vitesses de la surface libre d'une manière non-intrusive. *Melcher et al.* (2002) ont trouvé une estimation du débit très proche du débit de référence (à 5%). Les techniques de *Particle Image Velocimetry* (PIV, *Hauet*, 2009) permettent de mesurer le champ de vitesse de l'écoulement en introduisant un traceur ou particules visibles dans une partie éclairée de l'écoulement. L'idée est d'enregistrer une séquence d'images des particules qui se déplacent dans la région d'écoulement et ensuite traiter ces images pour en déduire les vitesses d'écoulement (*Creutin et al.*, 2003).

Ces systèmes peuvent prendre des données d'écoulement dans des conditions trop dangereuses pour les méthodes qui nécessitent un équipement immergé (inondation) et dans des situations où l'intérieur de l'écoulement est, pour une raison ou une autre, inaccessible ou difficile à atteindre. Elles peuvent améliorer les courbes de tarage existantes pendant les inondations.

Ces méthodes sont élaborées pour des raisons techniques de mesure, en particulier pour protéger l'instrument des aléas en cas de crue ou d'embâcles. Elles permettent de mesurer des vitesses en surface (par exemple, PIV, Lidar Doppler, etc.) ou des hauteurs d'eau (par exemple, GPR, distancemètres, etc.) sur des petites surfaces. Notons que ces techniques de mesure se limitent à des observations des petits cours d'eau avec la nécessité d'être proche du tronçon étudié, mais sans contact avec l'eau.

1.2.3 Approches pour l'estimation des courbes de tarage à la section

La courbe de tarage à la section peut être établie en utilisant des données issues des méthodes intrusives, non-intrusives, ou les deux en même temps. La géométrie hydraulique à l'aval (*Downstream Hydraulic Geometry*, DHG), inventée par *Leopold et Maddock* (1953), permet de prendre en compte implicitement le lien entre les caractéristiques du bassin versant et le débit. Elle permet de décrire des tendances similaires entre le débit moyen Q et la largeur W , la hauteur d'eau h et la vitesse moyenne u à une section donnée. Les fonctions suivent les relations de la géométrie hydraulique "à la section" (*At-a-station Hydraulic Geometry*, AHG, *Leopold et Maddock*, 1953). En effet, la largeur, la profondeur et la vitesse ont une relation exponentielle

avec le débit (Figure 1.5) telle que :

$$W = aQ^b$$

$$h = cQ^f$$

$$u = kQ^m$$

Où les coefficients (a , c et k) et les exposants (b , f et m) sont des paramètres ajustés empiriquement avec

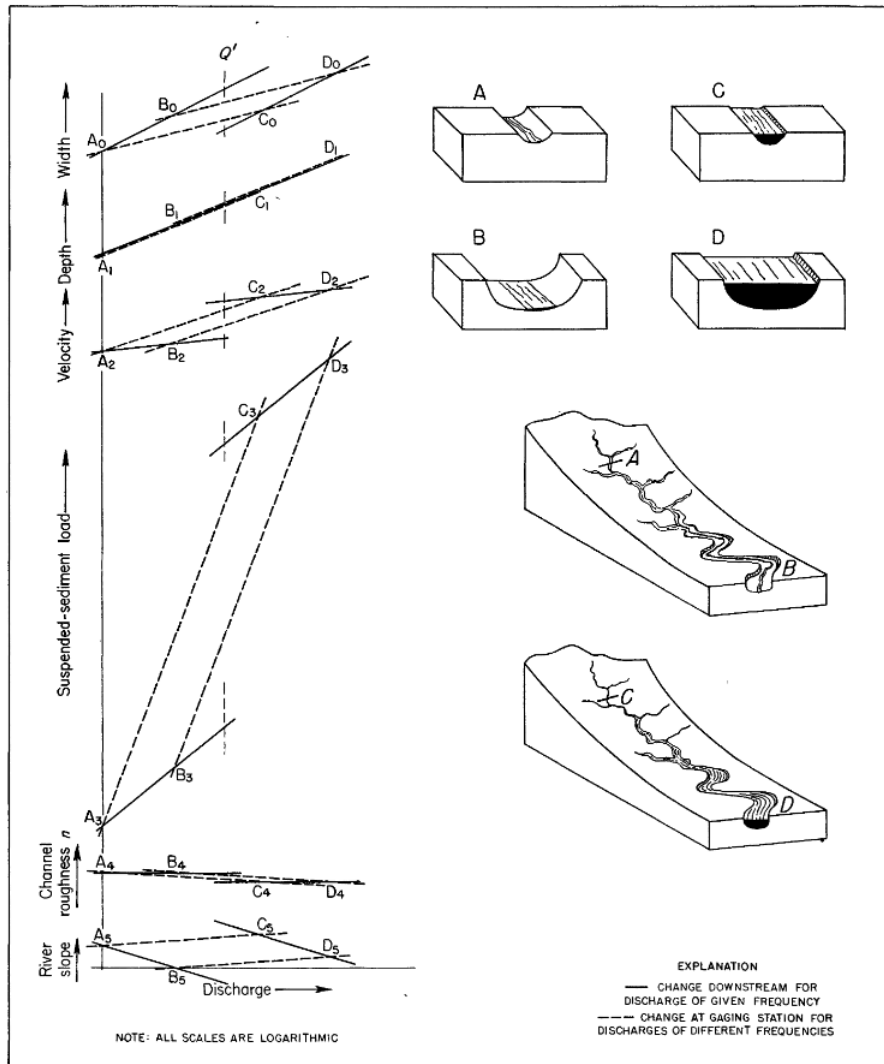


Figure 1.5: Géométrie hydraulique moyenne des cours d'eau exprimée par les relations entre la largeur, la profondeur, la vitesse, la charge en sédiments en suspension, la rugosité et la pente et le débit à deux sections (amont et aval) et à deux périodes (étiage et crue). Les échelles sont en Log (Figure prise de *Leopold et Maddock*, 1953).

$b + f + m = 1$ et $ack = 1$. Les coefficients représentent une estimation des grandeurs hydrauliques pour un débit de $1\text{m}^3/\text{s}$, tandis que les exposants donnent le taux d'augmentation de ces grandeurs avec le débit, indépendamment de la taille du cours d'eau. Ces paramètres dépendent de la section dans laquelle ils sont ajustés et ne sont pas applicables sur tout le tronçon. *Park* (1977) a analysé 139 cours d'eau et a montré que

les exposants de l’AHG ne sont pas reliés aux facteurs climatiques. *Ridenour et Giardino (1994)* ont montré, à travers leur travail, que ces exposants sont fortement liés à la stabilité des berges (présence de végétations et nature des sédiments).

Les équations de l’AHG, avec les paramètres ajustés, permettent d’estimer le débit dans une section à partir de n’importe quelle variable observable (h , W et u).

Dans le même sens, la méthode BaRatin (pour BAYesian RATing curve analysis) permet d’estimer la courbe de tarage et les incertitudes associées à l’aide d’un formalisme bayésien (*Le Coz et al., 2013*). Elle repose sur la définition des contrôles hydrauliques a priori. En effet, les contrôles hydrauliques peuvent se succéder ou s’ajouter les uns aux autres (Figure 1.6). Le niveau d’eau est contrôlé, à un certain débit, par les seuils présents dans le tronçon. Lorsque le débordement intervient, le contrôle hydraulique par le lit majeur s’ajoute à celui par le lit mineur.

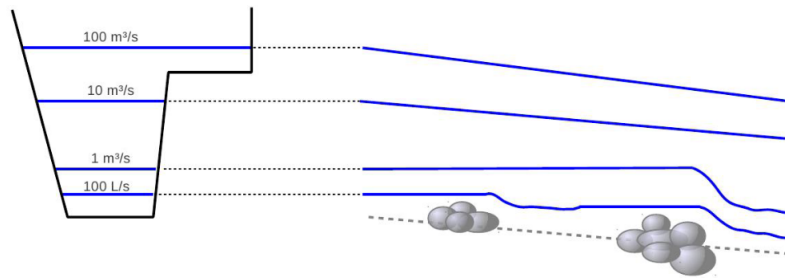


Figure 1.6: Représentation des contrôles hydrauliques pour une station hydrométrique typique : à droite, profil en long du fond et des lignes d’eau pour différentes valeurs de débit ; à gauche, les niveaux d’eau sont reportés sur le profil en travers de la section de la station (source : *Le Coz et al., 2014*).

L’inférence bayésienne permet de croiser les deux sources d’information incertaine (les mesures et l’expertise des contrôles hydrauliques) dans un cadre probabiliste pour estimer la courbe de tarage et les incertitudes associées. Les contrôles hydrauliques peuvent être approchés par une fonction de puissance qui lie la hauteur d’eau avec le débit :

$$Q = a(h - b)^c$$

Le coefficient c est fonction du type de contrôle (par exemple, section de contrôle de type seuil ou déversoir $c = 2.5$, tronçon de contrôle de type Manning-Strickler $c = 1.65$). Les autres coefficients a et b sont à optimiser de façon à reproduire au mieux le couple hauteur-débit jaugé ou modélisé.

La méthode BaRatin nécessite des informations a priori liées aux contrôles hydrauliques. Que faisons-nous dans un cas d’une rivière non-jaugée / non instrumentée ? Quel type de méthodes d’acquisition des données faut-il utiliser dans ce cas ?

Nous définissons une rivière non-jaugée par une rivière non instrumentée pour laquelle on ne dispose pas de données enregistrées dans le passé. D’après *Gleason et Durand (2020)*, nous ne pouvons considérer que les rivières jaugées pour lesquelles les données ne sont pas publiquement disponibles (en cas de conflit politique par exemple) comme des rivières non-jaugées. Dans ce cas, les méthodes de mesure non-intrusives introduites dans la section 1.2.2 (par exemple, GPR, radars Doppler, etc.) ne sont pas applicables et sont considérées des méthodes de “télé-détection locale”. Cette expression est utilisée pour les capteurs qui fonctionnent depuis les berges ou depuis l’avion. De plus, étant donné que la taille des fleuves est grande, les méthodes comme la PIV ne sont pas adéquates vu qu’elles sont utilisées sur des petites surfaces.

À l’échelle du tronçon, des méthodes comme LSPIV (*large-scale particle image velocimetry, Fujita et al., 1998*) ou MLSPIV (*mobile large-scale particle image velocimetry, Kim et al., 2008*) permettent d’avoir des distributions instantanées de la vitesse à la surface libre des plans d’eau d’une superficie aussi grande que

1km². En combinant ces données avec la topographie du fond de la rivière (non visible par les instruments de mesure des vitesses), la LSPIV peut donner des estimations en temps réel et en continu du débit (*Kim et al.*, 2008; *Fujita et al.*, 1998; *Hauet et al.*, 2008).

Par ailleurs, il y a les méthodes de mesures par des satellites qui suivent le même principe des LSPIV, mais en se plaçant plus loin des rivières et en mesurant d’autres variables en relation avec l’hydraulique et la morphologie du tronçon. Ces mesures altimétriques permettent d’observer un grand nombre de fleuves et de rivières sur toute la Terre en évitant le manque de données liée à des conflits politiques. Le satellite permet de “mesurer” à distance des largeurs, des pentes, des altitudes et des formes en plan comme dans le prochain satellite *Surface Water and Ocean Topography* (SWOT).

1.2.4 La télédétection au service des rivières non-jaugées

La croissance technologique a renforcé la capacité des satellites à observer et surveiller le système terrestre à l’échelle mondiale. Plusieurs altimètres Nadir qui observent la cote de la surface libre de l’eau² sont maintenant disponibles pour les applications terrestres (par exemple, JASON-3, Sentinel-3, Sentinel-3B, etc.).

À cet égard, de nombreuses études ont été menées sur l’utilisation de l’altimétrie par satellite pour la surveillance des eaux de surface (*Alsdorf et Lettenmaier*, 2003; *Papa et al.*, 2006; *Crétaux et Birkett*, 2006; *Calmant et Seyler*, 2006; *Crétaux et al.*, 2011). La plupart de ces études se sont focalisées sur l’observation des cotes de surface libre dans les grands fleuves et lacs en utilisant Topex/Poseidon (*Birkett*, 1995, 1998; *de Oliveira Campos et al.*, 2001; *Papa et al.*, 2003; *Kouraev et al.*, 2004) et les missions ENVISAT (*Frappart et al.*, 2006; *Medina et al.*, 2008; *Bartsch et al.*, 2009).

Les observations des cotes de surface libre peuvent également être obtenues par des images Synthetic-Aperture Radar (SAR) des masses d’eau continentales (*Giustarini et al.*, 2011). Bien que les missions altimétriques existantes aient été initialement conçues pour l’océanographie, la prochaine mission SWOT (qui devrait être lancée en fin de 2022), est conçue spécialement pour répondre aux besoins des communautés océanique et hydrologique/hydraulique (*Durand et al.*, 2010; *Pavelisky et al.*, 2014). En ce qui concerne les rivières, SWOT observera la cote, la pente et la largeur de surface libre, avec un niveau de précision sans précédent (pour un outil de télédétection). Contrairement aux satellites traditionnels, qui mesurent l’élévation en un seul point (donnant une ligne de mesures d’élévation lorsque le satellite tourne autour de la Terre), la mission SWOT fournira des mesures d’élévation par larges bandes permettant la mesure simultanée de l’étendue et de l’élévation de l’eau à surface libre avec une fréquence d’observations de 2 à 3 fois sur une période de 21 jours (*Bonnema et al.*, 2016).

L’instrument principal de SWOT est un interféromètre radar en bande Ka (KaRIn), qui utilise deux antennes séparées d’un Nadir de 10 m pour éclairer deux bandes de 50 km sur la surface de la Terre (Figure 1.7). Aux angles d’incidence proches du Nadir, l’eau est plus réfléchissante dans la bande Ka que les surfaces terrestres ; par conséquent, le volume de la puissance renvoyée sera utilisé pour distinguer la partie “eau” et la partie “non-eau” (*Biancamaria et al.*, 2016; *Fjørtoft et al.*, 2013).

Les mesures télédéteectées sont toujours affectées par des erreurs. Les principales sources de ces erreurs sont le bruit thermique des instruments (bruit blanc), les différences dans le chatoiement du signal de retour, les erreurs liées à l’interféromètre, les effets de la troposphère humide et sèche, les effets ionosphériques, la topographie et la végétation (*Biancamaria et al.*, 2016). Le radar fournira des données sous forme de nuage de points sur des pixels “natifs” qui ne sont pas carrés et qui sont $\sim 6 \times (10 \text{ m à } 60 \text{ m})$ (*Biancamaria et al.*, 2016; *Fjørtoft et al.*, 2013). À cette résolution, la précision verticale est faible. En effet, l’écart-type des erreurs verticales des pixels devrait être de l’ordre du mètre (*Domeneghetti et al.*, 2018). Pour obtenir une précision de hauteur centimétrique et atténuer le bruit de l’élévation de la surface libre, on effectue un moyennage spatial sur les pixels d’eau. Par exemple, un calcul de moyenne sur un tronçon de 10 km de longueur et 100 m de largeur entraînent des erreurs de hauteurs et de pentes de l’ordre de 10 cm et 1,7 cm/km (*Biancamaria et al.*, 2016; *Esteban-Fernandez*, 2013; *Frasson et al.*, 2017; *Rodriguez*, 2015). Les variables observées par le

²C’est la hauteur à partir du niveau zéro de la mer, en France, c’est le NGF : Nivellement Général de la France

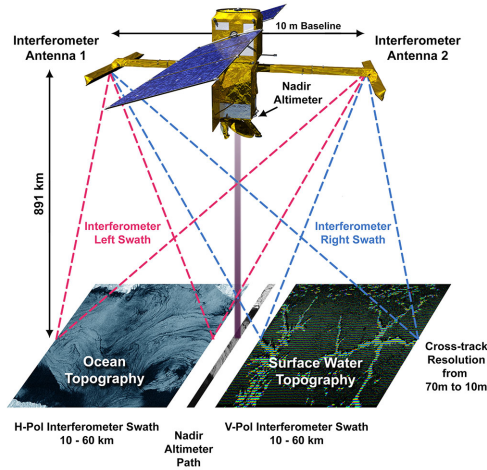


Figure 1.7: Illustration conceptuelle de la méthode des mesures de la mission SWOT, © JPL.

satellite sont toutes moyennées au tronçon. Pour l'estimation de la courbe de tarage, cette relation devient une relation “moyennée” à l'échelle de l'observation télédéteectée (un tronçon de rivière de quelques centaines de mètres à quelques kilomètres) dite **moyennée au tronçon** et non pas une relation estimée à une section précise. Divers degrés de complexité sont envisageables pour la construction de cette relation : modèles hydrauliques inversés ou approches simples liées à la géométrie des rivières (*Leopold et Maddock, 1953; Gleason et al., 2014; Durand et al., 2014; Gejadze et Malaterre, 2017; Oubanas et al., 2018a,b; Larnier et al., 2021*). Nous nous focalisons dans cette thèse uniquement sur les méthodes utilisant des données télédéteectées pour estimer les débits globaux à l'échelle du tronçon. Plus exactement, nous cherchons une méthode pour estimer le débit dans ce type de tronçons non-jaugés et pour lesquels nous ne disposons pas de connaissances *in situ* a priori.

Pour répondre à cette problématique, la section suivante présente l'état de l'art des méthodologies d'estimation des débits dans les rivières non-jaugées, la méthodologie de l'approche que nous proposons dans cette thèse et la démarche suivie pour la tester et la valider sur un tronçon.

1.3 Estimation des courbes de tarage au tronçon dans les rivières non-jaugées

En télédéteectée, la courbe de tarage au tronçon est une courbe qui relie le débit en sortie du tronçon avec la cote d'eau moyenne observable par le satellite. L'indisponibilité des mesures *in situ* de la bathymétrie partielle du cours d'eau et des coefficients de frottement (tronçon intégralement non-jaugé) rend le problème beaucoup plus difficile à inverser. En effet, le satellite visualise uniquement la plus basse ligne d'eau et donc la bathymétrie du lit mineur est partiellement inconnue. Le problème est donc mal posé (“*ill-posed*”) au sens de *Hadamard* (1902). D'où l'intérêt de définir le problème direct et le problème inverse associé.

1.3.1 Problème direct et inverse

Le modèle direct permet de générer des vitesses d'écoulement et des hauteurs d'eau à partir des données de topographie (lit mineur et lit majeur), des coefficients de frottement, du débit d'écoulement et des conditions aux limites. Le problème inverse associé à ce problème direct dans un cas non-jaugé permet de déterminer le débit d'entrée. Ceci consiste de connaître :

- la bathymétrie du lit mineur et du lit majeur ;
- les coefficients de frottement en lit mineur et en lit majeur.

D'un côté, le problème est bathymétrique puisque la bathymétrie du lit mineur n'est connue qu'au-dessus de la plus basse cote de la surface libre observée. D'un autre côté, le problème est frictionnel car il consiste à identifier tous les paramètres de rugosité. D'ailleurs, pour contraindre ce problème, des données peuvent être acquises à l'aide de la télédétection (niveaux d'eau, largeurs, pentes, MNT à haute résolution, etc.) ou des méthodes de mesure des vitesses (par exemple, MLSPIV). Dans le cas de la plupart des sources d'acquisition des données non-intrusives, la bathymétrie du lit mineur reste inconnue. Dans un tel contexte sous-contraint, les outils utilisés pour la résolution du problème inverse hydraulique simple, où la bathymétrie et les paramètres de friction sont obtenus préalablement et de façon indépendante, sont souvent inadaptés ("problème d'équifinalité") (*Gejadze et Malaterre, 2016*). La résolution du problème conjoint par une chaîne d'assimilation nécessite donc de réduire drastiquement la dimension du problème, par une forme ou une autre de régularisation. Le choix donc d'une simplification géométrique uniforme ou non-uniforme avec la résolution et la paramétrisation adéquate du modèle sont nécessaires pour réduire les degrés de liberté et construire la courbe de tarage associée au tronçon.

Quelles simplifications et paramétrisations doit-on choisir pour résoudre le problème hydraulique inverse ? Pour répondre à cette question, la section suivante présente un état de l'art des méthodes et des approches qui s'appuient sur la modélisation hydraulique ou des approches géométriques pour résoudre le problème inverse.

1.3.2 Etat de l'art des méthodes inverses en utilisant des données de télédétection

Plusieurs approches ont été proposées pour estimer le débit à partir de mesures télédéteectées. Ces approches peuvent aller d'un modèle complet de résolution des équations de la conservation de la masse et la quantité de mouvement, à de simples modèles capables de reproduire le débit du tronçon non-jaugé en passant par des équations de Manning-Strickler ou la géométrie hydraulique. Nous pouvons classifier ces approches en plusieurs catégories :

1.3.2.1 Approches avec une paramétrisation uniforme

Dans cette catégorie, nous pouvons trouver des méthodes inverses relativement basiques qui ont été développées dans le cas des tronçons non-jaugés. McFLI (*Mass conserved Flow Law Inversion*) est une méthode basée sur la géomorphologie et les lois de conservation de la masse. Elle suppose que la masse est conservée entre un ensemble de biefs connectés (ou un ensemble de sections dans un tronçon). La résolution du problème inverse en utilisant des données télédéteectées (de type largeurs, pentes et niveaux d'eau) devient simplement une résolution d'une loi d'écoulement (loi de Manning, géométrie hydraulique, etc.).

Durand et al. (2014) ont utilisé les données télédéteectées des niveaux d'eau de la rivière, sa largeur et sa pente pour déterminer le débit moyenné au tronçon de 22.4 km de la Severn. Ils se sont basés sur l'algorithme Metro-Man (*Metropolis Manning*) en utilisant la loi de Manning dans un contexte bayésien. Ils ont effectué plusieurs essais qui ont montré que cet algorithme réussit à estimer le débit à partir des mesures SWOT pour les tronçons de rivière dont les affluents latéraux sont connus (ou pour les tronçons sans affluents latéraux). Lorsque les apports latéraux sont inconnus et sont significatifs pour le bilan de masse du tronçon, ils posent un problème pour l'inversion. En revanche, une connaissance de la bathymétrie moyenne d'un seul bief dans la zone d'étude est suffisante pour surmonter le problème des apports latéraux inconnus. D'autres études, ayant utilisé le même principe sur différents types de données, ont produit des résultats encourageants (par exemple, l'algorithme GaMo, *Garambois et Monnier, 2015*, MFG, *Durand et al., 2016*, etc.).

L'autre loi utilisée dans l'approche de McFLI est celle de la géométrie hydraulique évoquée précédemment sous le nom de l'AHG. Ces relations sont définies à la section. Pour avoir une relation au tronçon, une

amélioration de l’AHG est introduite sous le nom de la géométrie hydraulique “à plusieurs sections” (*At-many-stations hydraulic geometry*, AMHG, *Gleason et Smith*, 2014). L’AMHG soutient que les coefficients et les exposants de l’AHG (a et b , c et f , et k et m) de nombreuses sections transversales d’un tronçon donné sont liés les uns aux autres, suivant une relation log-linéaire. *Gleason et Wang* (2015) ont montré que l’AMHG résulte du fait que les courbes de tarage individuelles et indépendantes passent toutes par les mêmes valeurs de W et Q , données en pratique par les modes spatiaux des moyennes temporelles de W et Q à chaque section transversale. En effet, *Gleason et al.* (2014) ont utilisé uniquement la largeur comme entrée du modèle. Ainsi, les inconnues du modèle sont l’exposant de la loi de puissance et les deux constantes de l’AMHG spécifique au tronçon. Les résultats ont été probants en utilisant un algorithme génétique pour déterminer le débit, ainsi qu’une formulation bayésienne (BAM, *Bayesian AMHG-Manning*) pour inverser le débit en utilisant uniquement la largeur ou la largeur, la hauteur et la pente (*Hagemann et al.*, 2017). *Shen et al.* (2016) ont montré que la relation largeur-débit est faible par rapport à la relation profondeur-débit de l’AMHG. Celle-ci peut émerger de la coévolution des variables morphologiques à l’échelle du tronçon.

1.3.2.2 Approches avec une paramétrisation implicite non-uniforme

L’inversion du débit à l’aide des données télédéteçtées peut être faite avec une paramétrisation non-uniforme implicitement. Cette paramétrisation est un peu complexe et nécessite donc une assimilation des données. *Biancamaria et al.* (2011) et *Yoon et al.* (2012) ont introduit l’assimilation des données et les bases des travaux récents d’*Oubanas et al.* (2018a,b), de *Larnier et al.* (2021), etc.

Le modèle VDA4D (*Variational Data Assimilation 4D*) utilise l’assimilation des données pour estimer le débit sous une incertitude dans la bathymétrie ou la rugosité. Ces approches estiment bien les débits dans les rivières non-jaugées. Toutefois, elles sont coûteuses en temps de calcul, ce qui contraint leur utilisation à grande échelle (*Gleason et Durand*, 2020; *Gejadze et Malaterre*, 2017). *Oubanas et al.* (2018a) ont montré, sous une incertitude dans la bathymétrie ou la rugosité et non pas les deux en même temps, que l’assimilation met à jour la valeur du paramètre correctement, mais quand la bathymétrie et la rugosité sont inconnues, l’assimilation ne peut pas donner de bons résultats. Récemment, *Larnier et al.* (2021) ont proposé une nouvelle méthode d’inversion pour déduire le débit, la bathymétrie effective (au dessous de la plus basse ligne d’eau) et le coefficient de friction correspondant à partir de mesures altimétriques du niveau d’eau, et plus particulièrement des données du prochain satellite SWOT. La méthode est basée sur la combinaison du VDA4D appliquée aux équations classiques de Saint-Venant (en 1D) et quelques hypothèses (écoulement en régime permanent avec un faible nombre de Froude). Les résultats sur trois rivières de ~ 100 km de longueur ont montré que l’algorithme d’inversion fournit des variations spatio-temporelles précises du débit avec une bathymétrie effective et un coefficient de friction correspondant si une bonne valeur moyenne est fournie comme entrée (par exemple une valeur moyenne de débit issue d’une base de données ou d’un modèle hydrologique à grande échelle).

1.3.2.3 Vers des approches avec une paramétrisation explicite non-uniforme

Globalement, l’estimation simultanée de la bathymétrie, de la rugosité et du débit à partir de mesures satellitaires est un problème complexe. La littérature (*Garambois et Monnier*, 2015; *Oubanas et al.*, 2018a; *Schaperow et al.*, 2019) suggère qu’une meilleure connaissance de la bathymétrie pourrait faciliter et améliorer considérablement la déduction de la rugosité et du débit. Pour cela, une paramétrisation explicite de la bathymétrie est nécessaire.

L’irrégularité et la variabilité spatiale de la rivière exigent la recherche d’un modèle simple et non-uniforme. De plus, la plupart des méthodes présentées ci-dessus se limitent au débit de plein bord et ne traitent pas le débordement. Néanmoins, la mission SWOT mesurera, non seulement les variables au niveau du lit mineur du tronçon observable, mais aussi les variables de la plaine inondable (la sinuosité, la surface inondée,

l'étendue du lit majeur, etc.) dont les données sont disponibles à partir d'autres campagnes de télédétection (par exemple, MNT, Orthophotos, etc.). Cela permettra d'avoir plus d'informations sur le tronçon et sa géomorphologie. Pour cela, l'utilisation d'une approche bidimensionnelle dans notre étude semble nécessaire pour arriver à une liaison entre le lit mineur et le débordement afin de modéliser l'inondation, chose qui est peu traitée par la communauté SWOT jusqu'à maintenant.

En effet, *Neal et al.* (2021) ont considéré un cas non-uniforme avec un écoulement graduellement varié (*Gradual Varied Flow*, GVF) pour estimer la bathymétrie de la rivière. Cette méthode s'est révélée plus performante que la méthode de l'équation de Manning. Elle a permis de reconstruire le profil de la surface libre observée par le LiDAR en minimisant le RMSE à des valeurs inférieures à 10 cm. Un modèle simplifié de GVF est couplé avec un modèle d'inondation global (*Global Flood Model*, GFM) développé dans l'étude de *Sampson et al.* (2015). Ce dernier a amélioré de plus en plus les résultats en montrant qu'une représentation inexacte du lit du tronçon, ne tenant pas compte des écoulements non-uniformes, pourrait fausser les calculs d'inondation et de stockage. Ceci est vrai même si les estimations de la bathymétrie ont été moins importantes pour les grands débits.

Par conséquent, quel type de modèle 2D explicite non-uniforme de la bathymétrie pouvons-nous utiliser pour réduire le nombre de degrés de liberté du problème inverse ? Dans la suite, nous allons présenter le contexte et les objectifs de la thèse ainsi que la démarche suivie dans l'élaboration, le test et la validation de l'approche proposée.

1.4 Contexte et objectifs de l'étude

L'objectif de la thèse est de proposer une approche permettant d'estimer le débit transitant dans un tronçon de rivière non-jaugée, en se basant uniquement sur des observations télédéteectées. Ces données comprennent la forme en plan de la rivière, les cotes et les pentes de la surface libre en lit mineur et lit majeur, la bathymétrie partielle du lit mineur (au-dessus de la plus basse ligne d'eau observée) et la bathymétrie du lit majeur. Pour réduire le nombre des degrés de liberté du problème inverse, une simplification de la bathymétrie est indispensable pour la résolution des équations de Saint-Venant en 2D. Ainsi, comment représenter la géométrie des rivières ?

1.4.1 Modèle stationnaire spatialement périodique comme hypothèse de travail

L'une des approches les plus utilisées est l'approche uniforme. Elle consiste à considérer que les paramètres géométriques ne varient pas spatialement dans un tronçon, qui mesure généralement 10 fois la largeur de plein bord (au minimum 1 km). Le tronçon peut regrouper différents faciès, courbures de méandre, seuils et même des ouvrages hydrauliques ou des ponts. Ne pas prendre en compte cette variabilité augmentera l'incertitude du modèle, qui apparaîtra sous la forme d'un biais dans les paramètres frictionnels, réduisant ainsi la robustesse de l'inversion. Notons que ce type de simplifications peut servir à grande échelle (à l'échelle régionale par exemple, *Sichangji et al.*, 2016).

Nous proposons une approche qui repose sur une simplification morphologique de la géométrie des rivières. En effet, la géométrie des cours d'eau naturels présente une variabilité haute fréquence, caractérisée par des alternances de faciès (*Navratil*, 2005) : faciès d'écoulement rapide dans les secteurs rectilignes et peu profonds (radiers ou seuils), faciès d'écoulement lent dans les secteurs plus profonds (mouilles au droit des bancs alternés ou côté intérieur des courbures de méandres, etc.). Cette variabilité engendre une variabilité des grandeurs hydrauliques qui coévoluent à l'échelle du tronçon (*Shen et al.*, 2016; *Mahdade et al.*, 2020). Une étude détaillée sur la structure de variabilité de ces grandeurs hydrauliques est présentée dans le Chapitre 2. Cette variabilité est caractérisée par des alternances de faciès d'écoulement (des unités morphologiques, par exemple, mouille-radier). Nous proposons donc une approche explicite qui représente ces alternances par une

variabilité périodique. Cette approche simplifie la géométrie de la rivière sans “gommer” son aspect physique et naturel à l'aide d'un modèle périodique qui permet de :

- représenter la géométrie équivalente du tronçon par des **fonctions strictement périodiques**, qui possèdent une variabilité similaire, en termes fréquentiels, à celle du tronçon réel, en respectant les covariances entre certaines grandeurs ;
- s'affranchir du besoin d'avoir une condition à la limite aval du tronçon puisque la périodicité ne change pas le long du tronçon ;
- être applicable en 2D en utilisant une périodicité en plan du tronçon ;
- surmonter la difficulté technique de la modélisation des méandres en utilisant les courbes de type “*Sine-generated*” ou Kinoshita (Kinoshita, 1961; Leopold et Langbein, 1966; Parker et al., 1983).

Modéliser la géométrie d'une rivière nécessite plusieurs paramètres selon l'approche choisie. Le choix de cette paramétrisation est détaillé dans le Chapitre 5.

Le modèle géométrique nécessite un solveur des équations de Saint-Venant 2D pour compléter le modèle hydraulique direct. Basilisk (Popinet, 2003) est un solveur des équations de Saint-Venant en 2D. Il est libre et développé par Stéphane Popinet avec des chercheurs de l'Institut Jean le Rond D'Alembert. Il permet de résoudre des équations aux dérivées partielles sur des mailles adaptatives. Développé en langage de programmation C, le solveur offre aux utilisateurs la possibilité de consulter la structure interne, de modifier les codes pour l'améliorer et d'ajouter de nouvelles fonctionnalités. Ce modèle hydraulique est adéquat aux besoins et aux hypothèses de l'approche géométrique périodique.

Ce modèle est utilisé dans un cas stationnaire (régime permanent) et avec une géométrie périodique (non-uniforme) pour inverser les données télédéteectées dans un tronçon non-jaugé. Quelle méthode faut-il choisir pour résoudre le problème inverse, c'est-à-dire pour lier les données observables par télédétection, le modèle hydraulique direct et la courbe de tarage au tronçon ?

Dans cette thèse, nous traitons deux aspects généraux. Le premier concerne les données utilisées, leur acquisition et leur traitement. Nous proposons des méthodes pour acquérir et construire des données avec une bonne précision pour tester les approches d'estimation des débits dans les rivières non-jaugées. Par ailleurs, le second aspect traite le modèle direct, sa paramétrisation, l'approche inverse et l'application des données préalablement préparées pour tester cette approche et valider les résultats.

1.4.2 Démarche à suivre pour tester l'approche périodique sur la Garonne

Pour tester et valider l'approche périodique sur un tronçon réel, il faut avoir des données télédéteectées et des données de validation *in situ*.

D'abord, nous avons choisi la Garonne comme une rivière de test/validation. Ce choix est motivé par la disponibilité des données (courbes de tarage, données télédéteectées de haute résolution, etc.) et par le fait qu'elle compte parmi les tronçons visés par la mission SWOT.

Ensuite, les données SWOT n'étant naturellement pas encore disponibles, plusieurs méthodes permettent de tester l'approche proposée :

- l'utilisation des données semblables à SWOT obtenues sur un site réel, par exemple les données de campagnes AirSWOT (Altenau et al., 2017; Tuozzolo et al., 2019) ;
- la simulation des lignes d'eau à différents débits à l'aide d'un modèle hydraulique, pour ensuite générer des “pseudo-observations” en échantillonnant cette simulation et en ajoutant un bruit de structure similaire à celui affectant les observations SWOT.

Dans cette thèse, nous avons adopté la deuxième approche, à savoir générer des pseudo-observations à l'aide d'une simulation de référence la plus précise possible.

La simulation de référence est une simulation de la crue de juin 2000 sur la Garonne. C'est une simulation débordante qui a atteint un pic de $\sim 3500 \text{ m}^3/\text{s}$. La réalisation de cette simulation nécessite une bathymétrie continue du tronçon, des données hydrologiques (courbes de tarage, hydrogrammes à injecter, etc.) et des coefficients de frottement. Pour cela, il est nécessaire de passer par plusieurs étapes (détaillées dans le Chapitre 3) à savoir :

- construire une bathymétrie du lit mineur et du lit majeur la plus précise possible du tronçon étudié ;
- calibrer les coefficients de frottement (utile aussi pour la validation du modèle inverse).

L'estimation de la courbe de tarage au tronçon nécessite une paramétrisation du modèle périodique direct en spécifiant les connus et les inconnus du modèle. Les paramètres connus sont tout type de paramètres observables par télédétection active ou passive (SWOT ou autre ancienne mission). Le nombre d'inconnues doit être le plus faible possible pour faciliter l'inversion. Une des méthodes d'inversion consiste à chercher un équivalent de la relation hauteur-débit au tronçon. Puisque les données sont bruitées au niveau du pixel, la recherche d'une signature équivalente qui permet de déterminer le débit au tronçon semble être une solution pertinente. Cette signature est une grandeur non-rattachée à une section particulière du tronçon, et qui caractérise un aspect de fonctionnement hydraulique global du tronçon. Elle peut être hydraulique, morphologique et bathymétrique. Parmi les signatures utilisables : la relation surface inondée-cote moyenne $A(\bar{\eta})$; cette surface est calculée directement à partir du nombre des pixels inondés. L'idée de l'algorithme d'inversion (Figure 1.8) est de chercher le meilleur jeu de paramètres inconnus qui minimise l'écart entre les signatures observées et simulées. Quelles sont donc les signatures moyennées au tronçon qu'il faut choisir à partir des données disponibles ?

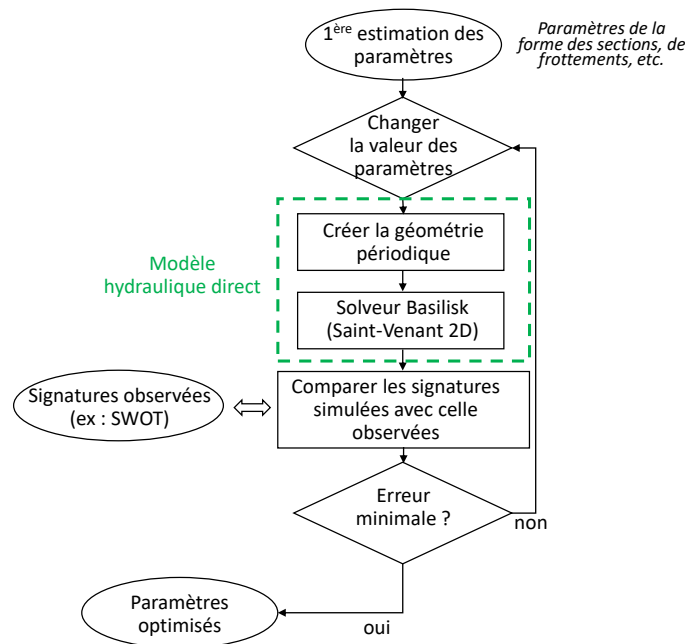


Figure 1.8: Schéma d'optimisation des paramètres inconnus du modèle.

1.5 Plan de la thèse

Pour répondre à la problématique générale de la thèse, à savoir : “Quelle représentation de la variabilité morphologique des rivières non-instrumentées pour construire des courbes de tarage théoriques au tronçon en utilisant des données télédéteectées ?”, la suite du manuscrit est subdivisé en cinq (5) chapitres. Le Chapitre 2, présenté sous la forme d’un article publié dans la revue *Hydrologie and Earth System Sciences* (Mahdade et al., 2020), analyse et caractérise la variabilité morphologique dans les rivières alluviales en introduisant des méthodes d’analyse spectrale. Ces méthodes permettent d’extraire une variabilité structurée de ces faciès (unités morphologiques) sous forme de signaux modulés en amplitudes et en fréquences. La périodicité des faciès morphologiques peut être visualisée comme une simplification de la géométrie du tronçon dans le cas d’une rivière non-instrumentée. Cette géométrie périodique diminue le nombre de degrés de liberté dans le problème inverse.

Avant de présenter en détail le modèle inverse, il faut préparer les données de validation de cette approche sur un tronçon réel (un tronçon de 40 km de la Garonne). Ceci fait l’objet du Chapitre 3, dans lequel nous présentons, tout d’abord, une méthodologie de construction de la bathymétrie continue du tronçon d’étude, en combinant des mesures *in situ* des sections en travers de la bathymétrie avec une teinte multi-spectrale issue des images Orthophotos du même tronçon. Ceci fait l’objet d’un deuxième article publié dans *Remote Sensing* (Le Moine et Mahdade, 2021). La topographie présentée et la simulation de référence de la crue de 2000 constituent l’ensemble du jeu de données pseudo-observé “*in situ*” de la Garonne. Le modèle hydraulique, le schéma numérique et le calage des coefficients de frottement, permettant d’effectuer la simulation de référence de crue, sont présentés dans la seconde partie de ce chapitre. Cette partie fait l’objet d’un article en préparation pour la revue *Geoscientific Model Development*.

Le Chapitre 4 présente une analyse descriptive des signatures hydrauliques, bathymétriques et morphologiques sur le tronçon de la Garonne. Cette analyse permet de déterminer les différentes signatures observées (valeurs uniques ou courbes). De plus, dans ce chapitre, nous proposons quelques réponses aux questions relatives au choix d’une représentation 2D, le seuil de visibilité du niveau de plein bord, etc. Les corrélations entre les variables de la forme en plan avec les variables de la forme en travers analysées dans le Chapitre 2 permettent de réduire le nombre des paramètres inconnus. Les signatures permettent donc d’identifier le reste des inconnus à l’aide des observations. La paramétrisation du modèle direct ainsi que l’algorithme d’inversion sont présentés dans le Chapitre 5 dans lequel nous allons présenter également les résultats préliminaires du cas test réalisé sur un tronçon similaire à la Garonne.

Le dernier chapitre à la fin de ce mémoire proposera des conclusions générales des résultats obtenus, avant de conclure sur des perspectives futures.

Chapitre 2

Caractérisation de la variabilité morphologique des rivières

2.1 Introduction

La morphologie d'un cours d'eau résulte des mécanismes d'érosion, de transport et de dépôt des sédiments. Le style morphologique est déterminé principalement par le type et la quantité de sédiments apportés au lit, bien que modifié par la taille du lit (*Church*, 2006). Les cours d'eau présentent des variations d'amont en aval désignées par le terme de "géométrie hydraulique aval" (*downstream hydraulic geometry*, DHG ; *Knighton*, 2014), dont les éléments les plus caractéristiques sont une diminution progressive de la pente et une augmentation de la largeur de plein bord (c'est le niveau à partir duquel il y a débordement du lit mineur vers la plaine inondable) pour un quantile de débit de référence. Cependant, à cette variabilité basse fréquence spatiale se superpose une variabilité haute fréquence, caractérisée par des alternances de faciès ou des unités morphologiques. Les unités morphologiques fluviales sont des formes de relief identifiables à l'échelle de $\sim 5-10$ fois la largeur de plein bord des rivières. Elles constituent l'une des délimitations fondamentales des reliefs fluviaux.

Le style fluvial d'une rivière peut être défini selon *Knighton* (2014) et *Hey et Thorne* (1986) par cinq degrés de liberté, à savoir : le profil en long (la pente) ; le profil en travers (la forme, hauteur et la largeur au miroir du lit mineur) ; la configuration du lit (séquences mouille-radier, bancs alternés, etc.) ; la rugosité du lit. Une perturbation des forçages externes qui se produit à la suite d'une modification du régime hydrologique ou simplement d'une succession de crues fortes force une évolution de la morphologie d'un équilibre dynamique à un autre. En effet, l'équilibre dynamique est un concept fondé sur le bilan équilibré entre les entrées et sorties en eau et en sédiments du système fluvial (*Navratil*, 2005). Les modifications qui perturbent cet équilibre dynamique sont dues dans la plupart du temps à l'intervention humaine (construction des digues, barrages, etc.). D'après *Knighton* (1981), la forme des sections peut varier à l'échelle de la crue, tandis que la modification de la longueur d'onde des méandres pourra s'observer sur une durée de l'ordre du siècle. À l'échelle de la crue, certains caractères de la morphologie apparaîtront comme mobiles : migration et recoupement de méandres, des mouilles et des radiers vers l'aval, brusques changements des sections en travers. À cette échelle temporelle, d'autres caractéristiques morphologiques se conservent, comme la pente longitudinale moyenne, le gabarit moyen, la longueur d'onde des méandres et des séquences mouille-radier. Dans cette étude, nous ne tenons pas compte de l'évolution temporelle de la morphologie, mais ces informations permettent de comprendre le processus fluvial d'une rivière et de trouver une approche simplificatrice pour le modéliser dans un modèle hydraulique.

Dans les rivières alluviales, les séquences mouille-radier sont les unités morphologiques les plus présentes dans le système fluvial. Elles représentent des faciès d'écoulement rapide dans les secteurs rectilignes et peu

profonds (radiers ou seuils, *riffle*) suivis par des faciès d'écoulement lent dans les secteurs plus profonds (mouilles, *pool* au droit des bancs alternés ou côté intérieur des courbures de méandres, etc.). Pour analyser ce type de faciès et comprendre comment elles se répètent le long d'un tronçon donné, le travail présenté dans la partie 2.2 étudie la variabilité morphologique de 6 tronçons avec des longueurs entre 155m et 495m et des largeurs inférieures à 50m. Ce travail est un *proof-of-concept* d'une méthodologie d'extraction des propriétés quantitatives des alternances des unités morphologiques et d'analyse de leurs variabilités. Cette variabilité peut être modélisée par des fonctions pseudo-périodiques pour donner une description continue des variables d'écoulement et de la géométrie.

2.2 Variabilité des unités morphologique à l'échelle du tronçon

Cette section est sous la forme d'un article publié dans *Hydrology and Earth System Sciences* :

Résumé : La précision des modèles hydrauliques dépend de la qualité des données bathymétriques sur lesquelles ils sont basés, quelle que soit l'échelle à laquelle ils sont appliqués. On sait que la géométrie longitudinale et transversale des cours d'eau naturels varie à l'échelle du réseau hydrographique (par exemple, la pente diminue généralement et la largeur augmente vers l'aval), ce qui permet de paramétrer les principaux paramètres transversaux à l'aide des proxy à grande échelle telles que la superficie de drainage ou le débit de plein bord (une approche appelée géométrie hydraulique en aval). Cependant, la variabilité morphologique haute-fréquence (c'est-à-dire à l'échelle du tronçon de rivière) est présente pour de nombreux types de cours d'eau, avec des conditions d'écoulement variables le long d'un tronçon donné, comme les séquences mouille-radier et les méandres. Pour prendre en compte cette variabilité haute-fréquence de la géométrie dans les modèles hydrauliques, une première étape consiste à concevoir des méthodes robustes pour caractériser les échelles auxquelles elle se produit. Dans cet article, nous introduisons de nouveaux outils d'analyse en ondelettes dans le domaine de la géomorphologie (à savoir, l'extraction de crêtes d'ondelettes) pour identifier la pseudo-périodicité des unités morphologiques alternées d'un point de vue général (en se concentrant sur les séquences mouille-radier) dans six petites rivières françaises. Cette analyse peut être réalisée avec une seule variable (cas univarié) mais aussi avec plusieurs variables (cas multivarié). Dans cette étude, nous choisissons un ensemble de quatre variables décrivant les degrés de liberté de l'écoulement : la vitesse, le rayon hydraulique, la contrainte de cisaillement du lit, et un descripteur de forme en plan qui quantifie la déviation locale du canal par rapport à sa direction moyenne. Enfin, cette méthode est comparée à la technique BDT (*Bedform Differencing Technique*), en calculant la moyenne, la médiane et l'écart-type de leurs espacements longitudinaux. Les deux méthodes montrent un accord dans l'estimation de la longueur d'onde dans tous les tronçons sauf un. La méthode vise à extraire une pseudo-périodicité des formes de morphologies alternées qui permettent une identification objective des unités morphologiques dans une approche continue avec le maintien des corrélations entre les variables (i.e., géométrie hydraulique à plusieurs stations, AMHG) sans qu'il soit nécessaire de définir un seuil préalable pour chaque variable pour caractériser la transition d'une unité à une autre.

Mahdade, M., Le Moine, N., Moussa, R., Navratil, O., and Ribstein, P.: Automatic identification of alternating morphological units in river channels using wavelet analysis and ridge extraction, *Hydrol. Earth Syst. Sci.*, 24, 3513–3537, <https://doi.org/10.5194/hess-24-3513-2020>, 2020.



Automatic identification of alternating morphological units in river channels using wavelet analysis and ridge extraction

Mounir Mahdade¹, Nicolas Le Moine¹, Roger Moussa², Oldrich Navratil³, and Pierre Ribstein¹

¹Sorbonne Université, CNRS, EPHE, Milieux environnementaux, transferts et interaction dans les hydrosystèmes et les sols, METIS, 75005 Paris, France

²INRA, UMR LISAH, 2 Place Pierre Viala, 34060 Montpellier, France

³University of Lyon, Lumire Lyon 2, Department of Geography, CNRS 5600 EVS, Lyon, France

Correspondence: Mounir Mahdade (mounir.mahdade@upmc.fr)

Received: 10 July 2018 – Discussion started: 11 September 2018

Revised: 28 May 2020 – Accepted: 11 June 2020 – Published: 14 July 2020

Abstract. The accuracy of hydraulic models depends on the quality of the bathymetric data they are based on, whatever the scale at which they are applied. The along-stream (longitudinal) and cross-sectional geometry of *natural* rivers is known to vary at the scale of the hydrographic network (e.g., generally decreasing slope, increasing width in the downstream direction), allowing parameterizations of main cross-sectional parameters with large-scale proxies such as drainage area or bankfull discharge (an approach coined downstream hydraulic geometry, DHG). However, higher-frequency morphological variability (i.e., at river reach scale) is known to occur for many stream types, associated with varying flow conditions along a given reach, such as the alternate bars or the pool–riffle sequences and meanders. To consider this high-frequency variability of the geometry in the hydraulic models, a first step is to design robust methods to characterize the scales at which it occurs. In this paper, we introduce new wavelet analysis tools in the field of geomorphic analysis (namely, wavelet ridge extraction) to identify the pseudo-periodicity of alternating morphological units from a general point of view (focusing on pool–riffle sequences) for six small French rivers. This analysis can be performed on a single variable (univariate case) but also on multiple variables (multivariate case). In this study, we choose a set of four variables describing the flow degrees of freedom: velocity, hydraulic radius, bed shear stress, and a planform descriptor that quantifies the local deviation of the channel from its mean direction. Finally, this method is compared with the bedform differencing technique (BDT), by computing the mean, median, and standard deviation of their lon-

gitudinal spacings. The two methods show agreement in the estimation of the wavelength in all reaches except one. The method aims to extract a pseudo-periodicity of the alternating bedforms that allow objective identification of morphological units in a continuous approach with the maintenance of correlations between variables (i.e., at many station hydraulic geometry, AMHG) without the need to define a prior threshold for each variable to characterize the transition from one unit to another.

1 Introduction

Hydraulic modeling is based on the description of river morphology (cross-sectional geometry), and this is the essential input of models despite its scarcity and cost of acquisition. The most important aspect to know is the river bathymetric data at the local scale, detailed and specific to the site and local conditions (Alfieri et al., 2016). This component is essential for accurate modeling of river hydraulics such as flood modeling (e.g., Neal et al., 2015; Trigg et al., 2009), river restoration (e.g., Wheaton et al., 2004), ecohydraulics (e.g., Pasternack and Brown, 2013), environmental modeling, and fluvial process (e.g., Rodríguez et al., 2013). Longitudinal variability in river geometry may have a greater impact on the simulation of the water level than the cross-sectional shapes (Saleh et al., 2013), and it must be taken into account in the hydraulic models. This variability of river geometry at a small scale in the longitudinal and cross-sectional direction yields a variation in flow parameters and is known to occur

for many morphological channel types, each type being characterized by typical morphological units (MUs), e.g., pools, riffles, steps, point bars, and meanders.

1.1 State-of-the-art methods for a quantitative assessment of morphological variability within a reach

Morphological units are topographic forms that shape the river corridor (Wadeson, 1994; Wyrick and Pasternack, 2014). They form alternating and rhythmic undulations continuously varying along the river (Thompson, 2001). This continuity is challenging to represent; for this reason, most of the methods that model these patterns divide the topography into discrete units to analyze them (Kondolf, 1995; Wyrick et al., 2014).

Among the most frequently observed alternating MUs, pools and riffles have been recognized as fundamental geomorphological elements of meandering streams (Krueger and Frothingham, 2007). Pools are located in the outer edge of each meander loop and defined as topographic lows along a longitudinal stream profile with high depth and low velocity (Fig. 1a, b and d), and research has shown that they generally have an asymmetrical cross-section shape. Conversely, riffles are topographic highs with shallow depths and moderate to high velocities located in the straight parts of the reach between adjacent loops (Fig. 1a, c and d), and they have symmetrical cross-section shapes (O'Neill and Abrahams, 1984; Knighton, 1981).

For many years, many researchers have been trying to develop techniques to identify MUs and especially pools and riffles using hydraulic variables or topographic ones, or both (Table 1). In 1D identification, some studies used bed topography only to determine the characteristics of MUs. Richards (1976a) proposed the zero-crossing method, which fits a regression line to the longitudinal profile of the bed elevation and defines pools as points that have negative residuals and riffles as points with positive residuals. O'Neill and Abrahams (1984) developed the bedform differencing technique (BDT) as a refinement of Richards' methodology. This one uses bed elevations measured at a fixed interval along the channel to calculate the bed elevation difference series between local extrema (maximum and minimum) of the bed profile. The BDT introduces a tolerance value (T); it is the minimum absolute value of the cumulative elevation change required for the identification of a pool or riffle. The value of T is based on the standard deviation of the bed elevation difference series and eliminates the erroneous classification of small undulations in the bed profile. Knighton (1981) proposed the Areal Difference Asymmetry Index to identify the location of pools and riffles by their symmetrical or asymmetrical areas. This index is defined as the ratio of the difference between the area of the right and the left of the channel centerline on the total cross-sectional area.

On the other hand, some studies focused only on hydraulic parameters to identify MUs. For example, Yang (1971) proposed an identification of pools and riffles using the energy gradient and affirmed that the fundamental difference between riffles and pools is the difference in energy gradients. Also, Jowett (1993) proposed a classification criterion with Froude number and velocity–depth ratio to distinguish between pools, runs, and riffles.

All these methods handle topographic or hydraulic parameters separately. Recently, however, several researchers have improved MU identification through the use of the covariance of several parameters in a multidimensional approach. Schweizer et al. (2007) used a joint depth and velocity distribution to predict pools, runs, and riffles without the knowledge of the river bathymetry. Hauer et al. (2009) used a functional linkage between depth-averaged velocity, water depth, and bottom shear stress to describe and quantify six different hydro-morphological units (riffle, fast run, run, pool, backwater, and shallow water) using a conceptual Mesohabitat Evaluation Model (MEM) under various flow conditions. These methods use digital elevation models (DEMs) to extract more information about MUs. For this purpose, Milne and Sear (1997) began with depth to define pool–riffle sequences using ArcGis tools and DEMs to model the geometry of river channels based on field-surveyed cross sections on a 3D basis. But, by choosing depth alone, the difference between two bedforms with the same depth becomes difficult to know. By contrast, it is easy with different bed slopes and bed roughness that yield different velocities and shear stresses (Wyrick et al., 2014). So to overcome this and take into account the lateral variation of rivers, Wyrick et al. (2014) proposed a new method for the objective identification and mapping of landforms at the morphological unit scale. They used spatial grids of depth and velocity at low flow estimated using a 2D hydrodynamic model and an expert classification scheme that determine the number and the nomenclature of MUs and the range of base flow depth and velocity of each type.

Brown and Pasternack (2017) chose two variables: the minimum bed elevation and the channel top width across several flow discharges. They calculated the geomorphic covariance structure (GCS); it is a bivariate spatial relationship amongst or between standardized and possibly detrended variables along a river corridor. They found that there is a positive correlation between these two variables. Also, they used an autocorrelation function and power spectral density to prove a quasi-periodic pattern of wide and shallow or narrow and deep cross sections along the river. This pioneering work and other studies (e.g., Richards, 1976b; Carling and Orr, 2000) proved that a single longitudinal cycle might contain a pool with a narrow and deep cross section, a riffle with a wide and shallow cross section, in addition to transitional forms. The work that we introduce in this paper aims to present a spectral method that extracts this pseudo-periodicity from a river to characterize the alternating MUs,

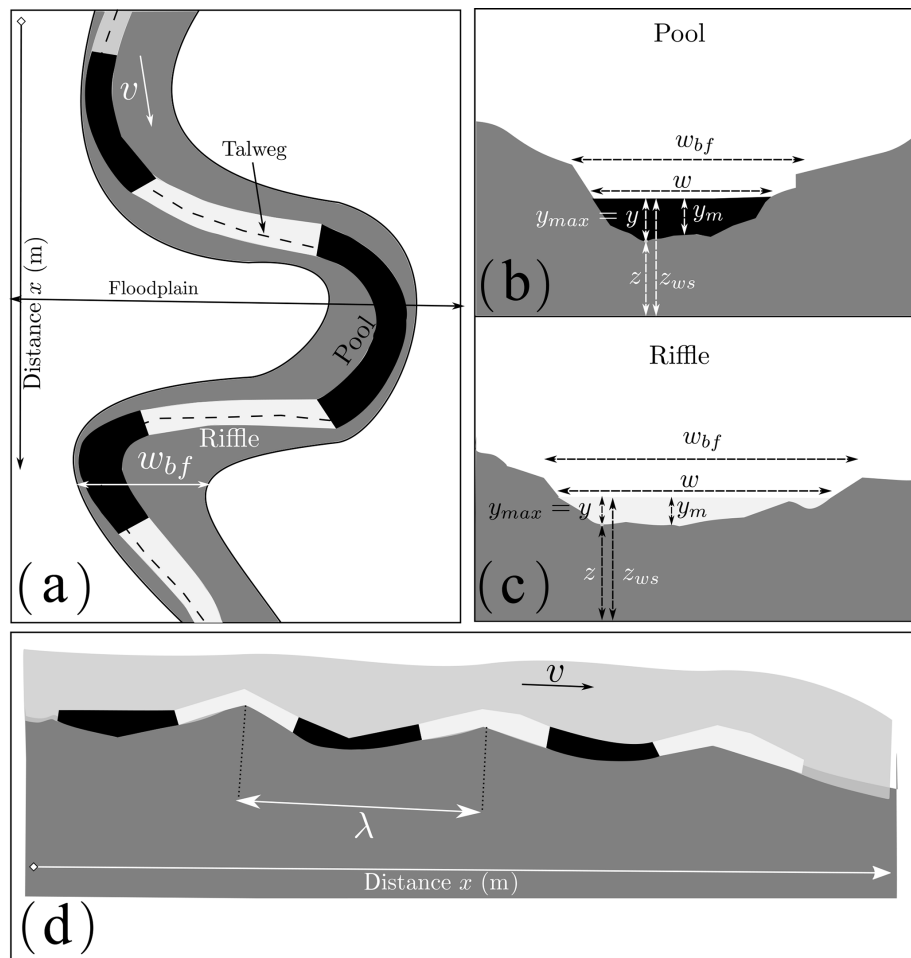


Figure 1. Different views of pool–riffle sequences. **(a)** Plan view pattern that includes bankfull width w_{bf} , floodplain extent, talweg line, velocity v , pools and riffles, and channel direction (planform); **(b)** cross-sectional view of a pool with a section width w and a steeper water depth y calculated from the talweg elevation, which is the deepest part of the bottom, and $y = y_{max} = z_{ws} - z$, with z the bed elevation, z_{ws} the water surface elevation, and y_m the mean water depth; **(c)** cross-sectional view of a riffle with a shallower water depth y , higher bed elevation z and high bankfull width w_{bf} ; **(d)** longitudinal profile that makes it possible to see the water surface, the bed slope, the pools and riffles, and the wavelength λ calculated between two successive riffles or pools.

and especially pool–riffle sequences, and to identify the key parameter (the wavelength) that characterizes the scale of variability of the river topography. This information can be further used to build a synthetic river such as the River-Builder (Pasternack and Zhang, 2020) or the channel builder for simulating the river morphology of Legleiter (2014).

Some of the methods presented in the literature have shown limits in calculating the wavelengths of pool–riffle sequences. Others have given results that are often difficult to interpret in terms of bedform amplitude. This amplitude, which varies according to each bedform, involves the use of the pseudo-period. A few methods are developed to extract this pseudo-period from alternating MU rivers. We, therefore, choose to work with wavelet analysis that estimates the local variability strength of a signal and extracts the signal amplitude and wavelength. In this study, we apply continu-

ous wavelet transform (CWT) to calculate the wavelength λ and the dimensionless wavelength spacing λ^* (longitudinal spacing) which is

$$\lambda^* = \frac{\lambda}{w_{bf}}, \quad (1)$$

where w_{bf} is bankfull width.

In reality, the longitudinal spacing λ^* has several definitions. Some authors have defined the wavelength λ as the distance between riffle crests (e.g., Harvey, 1975; Hogan, 1986) or the distance from the bottom of successive pools (e.g., Keller and Melhorn, 1973, 1978). Other authors have chosen channel width w (e.g., Richards, 1976a, b; Dury, 1983) instead of bankfull channel width w_{bf} (e.g., Leopold et al., 1964). These differences raise questions about the selection of these ratios and their dependence on geometric or hydraulic parameters. Moreover, the majority of researchers use

Table 1. Review of some methods of morphological units' identification (variable used and MU types).

Methods	Variables	MUs	References
Control-point method	Energy gradient	Pools and riffles	Yang (1971)
Zero-crossing method	Bed topography	Pools and riffles	Richards (1976a), Milne (1982)
Areal difference asymmetry index	Cross-section area	Pools and riffles	Knighton (1981)
Power spectral analysis	Bed topography	Pools and riffles	Nordin (1971), Box and Jenkins (1976)
Bedform differencing technique (BDT)	Bed topography	Pools and riffles	O'Neill and Abrahams (1984)
Hydraulic characteristics classification	Froude number	Pools, runs, and riffles	Jowett (1993)
3D identification	Water depth	Pools and riffles	Milne and Sear (1997)
Schweizer's method	Water depth and velocities	Pools, runs, and riffles	Schweizer et al. (2007)
MEM Model	Water depth, velocity, and bottom shear stress	Pool, riffle, run, fast run, shallow water, and backwater	Hauer et al. (2009, 2011)
Wyrick's method	Water depth and velocity	Pools, riffles, runs, and glides	Wyrick and Pasternack (2014), Wyrick et al. (2014)
Brown and Pasternack method	Minimum bed elevation and channel top width	Pools and riffles	Brown and Pasternack (2017)

the average channel width instead of the bankfull width because both give a similar pool–riffle spacing interval. Here, we are working with w_{bf} and with a new automatic wavelength calculation method that uses the whole covariance structure of a set of hydraulically independent variables without the need for ad hoc thresholding of these variables.

Some researchers have investigated the variability of longitudinal spacing depending on geometric or hydraulic parameters. Rosgen (2001) developed an empirical relationship between the ratio of pool-to-pool spacing/bankfull width and the channel slope expressed as a percentage based on a negative power function of slope S :

$$\lambda^* = 8.2513S^{-0.9799}. \quad (2)$$

In addition, Montgomery et al. (1995) showed that there is an influence of large woody debris (LWD) on channel morphology that leads to a relation between LWD and longitudinal spacing in a pool–riffle sequence, and they found that 82% of pools were formed by LWD or other obstructions, and increased numbers of obstructions led to a decrease in the pool–riffle spacing. Moreover, research has linked variation in spacing to channel characteristics, including gradient (Gregory et al., 1994). Also, Harvey (1975) showed that pool–riffle spacing correlated strongly with discharges between the mean-annual flood and a 5-year recurrence interval (Thompson, 2001). Recently, Wyrick and Pasternack

(2014) measured the spacing of six different morphological units using a tool in ArcGIS. Therefore, the definition of the characteristics and the measurement methods allowed us to expect some variation from one study to another in the estimated relationship between longitudinal spacing and bankfull width (Richards, 1976a; O'Neill and Abrahams, 1984; Gregory et al., 1994; Knighton, 2014). Aside from the interval $[5w_{bf}, 7w_{bf}]$ defined by Leopold et al. (1964) and the interval $[2w_{bf}, 4w_{bf}]$ defined by Montgomery et al. (1995) in forested streams, other values of the longitudinal spacing exist, such as the Carling and Orr (2000) interval, which is $[3w_{bf}, 7.5w_{bf}]$ and decreases to $[3w_{bf}, 6w_{bf}]$ as sinuosity increases (Clifford, 1993; Carling and Orr, 2000).

1.2 Study objectives

The studies that used wavelet analysis in the geomorphological field consist in extracting components of a given spatial series (e.g., $w(x)$, $v(x)$), but they are not specifically designed to identify pseudo-periodic components in a univariate case, let alone in a multivariate case. For this reason, we introduce an automatic procedure called wavelet ridge extraction defined by Lilly and Olhede (2011) and used in this study to extract the longitudinal spacing of the alternating MUs.

The objective is to extract some quantitative properties of these alternating morphological units, such as the mean and the median of their longitudinal spacing, with a continuous vision of the topography instead of a discrete classification. For that, we focus on two numerical criteria computed at reach scale: the distribution of spacings between morphological units (mean, median, etc.) and the evaluation of correlations between all geometrical and flow variables. We use in this work four classical variables (velocity, hydraulic radius, bottom shear stress, and the local channel direction angle) because they respond directly to morphodynamic processes (flow convergence routing or meander migration), and they are independent hydraulic degrees of freedom.

In this study, we focus mainly on alternating alluvial channels, especially pool–riffle sequences, even though the method presented here could be used to analyze any morphology characterized by alternating topographic forms. We first present the dataset of six river reaches in France used for this analysis (Sect. 2). In Sect. 3, we present the wavelet ridge extraction method to identify pool–riffle sequences in the univariate and multivariate cases with the four variables. Section 4 presents results and compares them with the BDT developed by O’Neill and Abrahams (1984) to determine whether they yield the same results in terms of spacing. We choose this method instead of threshold methods because the latter require ad hoc thresholding/parameter range definition from independent calibration data, which was not possible in our case.

The rationale behind this approach is to provide a continuous description of geometric and flow patterns along a reach using the wavelength, a description that could be subsequently used to create a synthetic river as in the River-Builder (Brown et al., 2014).

2 Dataset and study reaches

Six reaches of small French rivers are used in this study (Navratil, 2005; Navratil et al., 2006): the Graulade at St Sylvain Montaignut (1), the Semme at Droux (2), the Olivet at Beaumont (3), the Ozanne at Tirzay lès Bonneval (4), the Avenelles at Boissy-le-Châtel Les Avenelles (5), and the Orgeval at Boissy-le-Châtel Le Theil (6) (Fig. 2). These reaches contain mainly pool–riffle sequences. They have slopes between 0.002 and 0.013 m m⁻¹ (estimated from the talweg elevation, which is the lowest point in the section), mobile gravel beds, stable banks, and well-defined floodplains along at least one side of the channel (Navratil et al., 2006). These reaches are located in the Loire River basin (four reaches) and the Seine River basin (two reaches), and their length ranges from 155 to 495 m (Table 2). All reaches are located at or near the stream gauging stations of the French national hydrometric network. Long-term (about 20 years) hydrological records are available for most reaches.

The bankfull widths vary from 4 to 12 m, with an average value of about 9 m.

Cross sections were surveyed along the river reaches at the level of hydraulic controls and morphological breaks to describe the major variations in terms of width, height, and slope in the main channel and the floodplain and at the level of pool–riffle sequences. Cross sections and water surface profile measurements were surveyed in 2002–2004, covering the main channel and floodplain and using an electronic, digital, total-station theodolite. Water surface profiles were measured at different flow discharges (Navratil et al., 2006). Using this dataset, we solely rely on measurements at the lowest surveyed discharge in the development of the method because it is the discharge through which we can visualize the variability of the bathymetry (alternating morphological units). We select four spatial series:

1. velocity $v(x)$;
2. hydraulic radius $R_h(x) = \frac{A(x)}{P(x)} \approx \frac{A(x)}{w(x)}$ with $A(x)$ the cross-sectional area and $P(x)$ the wetted perimeter;
3. bed shear stress $\tau_b(x) = (\rho g)n^2 v(x)^2 R_h(x)^{-1/3}$ with ρ water density (1000 kg m⁻³) and n Manning’s roughness coefficient;
4. local channel direction angle (planform) $\theta(x)$.

All descriptors are derived from in situ observations taken from Navratil et al. (2006), except the calibrated estimates of Manning’s roughness coefficient n . These values were estimated by Navratil et al. (2006) using a 1D open channel steady and step backwater model, FLUVIA (Baume and Poirson, 1984). However, we use these values to compute the bed shear stress $\tau_b(x)$, along the reach: even if it partly relies on calibration, it is a more robust way of computing τ_b here than through the finite differentiation of the total head function $\frac{v(x)^2}{2g} + z(x)$ between adjacent cross sections to get the energy slope J , given the typical number and spacing of surveyed cross sections for each reach in the dataset.

The fourth variable chosen is related to the channel planform: we define $\theta(x)$ as the local angular deviation of the channel direction from a lower-frequency curve. There are many possible definitions of this low-frequency behavior, such as parametric splines or Bezier curves; in order to avoid over-parameterization, we define this low-frequency planform as a constant curvature curve, i.e., the best-fitting arc-circle (Fig. 3), a choice suitable for all six reaches studied. Since θ is signed, it is expected to have a pseudo-periodicity, which is approximately twice as slow as other 1D variables. Indeed, a large positive value of θ indicates a counterclockwise deviation from the low-frequency direction, while a large negative value of the same amplitude indicates a clockwise deviation. From a hydraulic perspective, both deviations have the same effect since they are symmetrical with respect to the low-frequency direction. For this reason, we chose to analyze the variable $\cos(\theta(x))$.

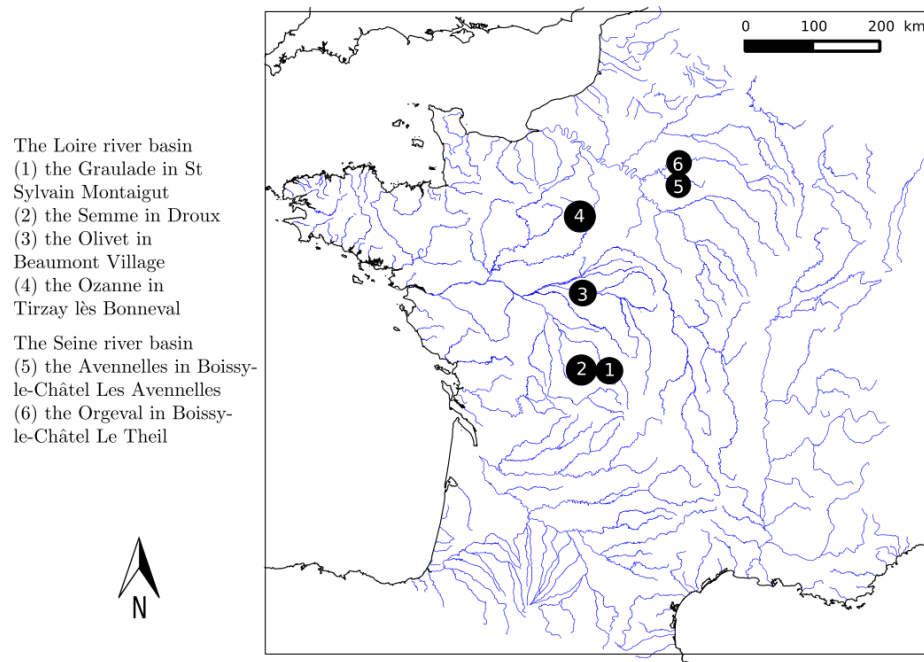


Figure 2. The locations of the study reaches in France.

Table 2. Characteristics of the six reaches and their catchment. The bankfull width w_{bf} is taken from the study of Navratil et al. (2006), and the average width w_m and the standard deviation $\sigma(w)$ are calculated for minimum discharge Q_{min} .

Reach	(1) Graulade	(2) Semme	(3) Olivet	(4) Ozanne	(5) Avenelles	(6) Orgeval
Reach length L (m)	160	177	495	319	155	318
Number of cross sections	14	32	66	26	25	36
Reach gradient S ($m\ m^{-1}$)	0.0125	0.0044	0.0018	0.0024	0.0060	0.0047
Bankfull width w_{bf} (m)	4	12	6	12	9	10
Average width w_m (m)	2.8	9.3	4.7	7.0	3.3	6.1
Standard deviation $\sigma(w)$ (m)	0.4	1.9	0.9	1.1	0.9	1.0
Surveyed flow discharges ($m^3\ s^{-1}$)	0.22 and 1.26	1.85 and 2.41	0.18; 1.13; 1.72 and 1.99	0.19; 0.33; 0.8 and 11.5	0.15	0.21
Min discharge Q_{min} ($m^3\ s^{-1}$)	0.22	1.85	0.18	0.19	0.15	0.21

3 Wavelet method

Classical mathematical methods, such as Fourier analysis, extract the wavelengths in the frequency domain for stationary signals, but can also be used for non-stationary signals using an “evolutive” methodology based on spectral estimators (Thomson, 1982; Pasternack and Hinnov, 2003). Wavelet transform standardly does the same for non-stationary signals: analyzing a signal basically consists in looking for the local similarity between the signal and a given waveform (the wavelet). In this paper, we use the continuous wavelet transform with the Morlet wavelet (Gabor, 1946) (Fig. 4) ap-

plied to spatial series instead of time series, so periods and frequencies in time series are replaced by wavelengths (in m) and wavenumbers (in $rad\ m^{-1}$). The choice of the Morlet wavelet is justified by the analytical properties in its derivation and its flexibility due to the exponential form (see Appendix B).

The wavelet transform uses a whole family of “daughter” wavelets generated by scaling and translating the mother wavelet ψ ; the value of the transform at location x and scale s is the scalar product of the signal and this daughter wavelet $\psi_{s,x}$.

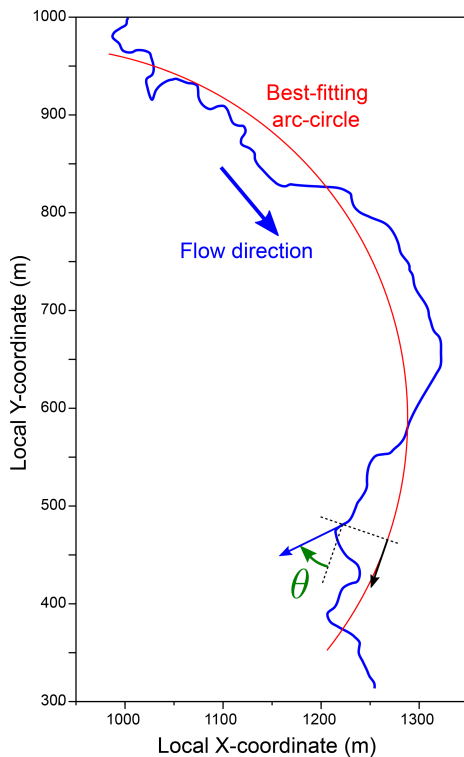


Figure 3. Definition of θ , the local angular deviation of the channel direction from a lower-frequency behavior. Here this low-frequency platform is defined as an arc-circle (illustration on the Olivet River reach). It is worth noting that θ is signed: at the location indicated in the figure, θ is negative.

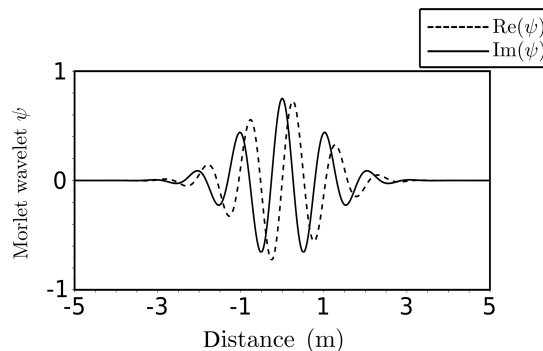


Figure 4. The wavelet Morlet mother function; the plot shows the real and imaginary parts of the wavelets in the space domain (distance).

Wavelet analysis is prevalent in many fields, such as fluid mechanics (e.g., Schneider and Vasilyev, 2010; Higuchi et al., 1994; Katul et al., 1994; Katul and Parlange, 1995b, a), meteorology (e.g., Kumar and Foufoula-Georgiou, 1993; Kumar, 1996), geophysics (e.g., Ng and Chan, 2012; Grinsted et al., 2004), hydrology (e.g., Rossi et al., 2011; Schaeffli et al., 2007; Nourani et al., 2014), and geomorphology (Gangodagamage et al., 2007; Lashermes and Foufoula-Georgiou,

2007; McKean et al., 2009). In the literature of the alternating bedforms' identification, McKean et al. (2009) used a derivative of a Gaussian wavelet (DOG) of order 6 to investigate the spatial patterns (pools and riffles) of channel morphology and salmon spawning using a 1D elevation profile of the channel bed morphology.

In this study, we use another application of the wavelet analysis called the wavelet ridge extraction method (Mallat, 1999; Lilly and Olhede, 2010). This analysis is based on the existence of special space/wavenumber curves, called wavelet ridge curves or simply ridges (Lilly and Olhede, 2010), where the signal concentrates most of its energy (Carmona et al., 1999; Ozkurt and Savaci, 2005). Along such a curve, the signal can be approximated by a single component modulated in both amplitude and frequency. So, the rationale behind the method is that the existence of alternating morphological units along a reach (such as pool–riffle sequences) could be translated into a pseudo-periodicity in geometric and flow variables. Hence, identifying these bedforms requires identification of a local wavenumber $K(x)$ and phase $\Phi(x)$ for each variable, a task that can be performed by wavelet analysis and especially wavelet ridge extraction (Mallat, 1999; Lilly and Olhede, 2010).

3.1 Wavelet analysis and ridge extraction

Few methods in the literature have been trying to identify river characteristics with wavelets. For example, Gangodagamage et al. (2007) used wavelet transform modulus maxima (WTMM, Muzy et al., 1993) in a fractal analysis to extract multiscale statistical properties of a corridor width. Procedures such as the WTMM consist in extracting components of the signal, but they are not specifically designed to identify pseudo-periodic components in a univariate case, let alone in a multivariate case.

In the present study, we test a new wavelet ridge analysis on spatial series with the Morlet mother basis function represented in Fig. 4. Its expression is

$$\psi(\eta) = \pi^{-\frac{1}{4}} e^{i\beta\eta} e^{-\frac{\eta^2}{2}}, \quad (3)$$

with ψ the mother wavelet function that depends on the dimensionless “position” parameter η and β the dimensionless frequency, here taken to be 6, as recommended by Torrence and Compo (1998). Starting with this wavelet mother, a family $\psi_{s,x}$ called wavelet daughters is obtained by translating and scaling ψ .

$$\psi_{s,x}(\eta) = \frac{1}{\sqrt{s}} \psi\left(\frac{\eta-x}{s}\right), \quad x \in \mathbb{R}, s > 0, \quad (4)$$

with x the translation offset, which represents a position at which the signal is analyzed, and s the dilation or scale factor. If $s > 1$, the daughter wavelet has a frequency lower than the mother wavelet, whereas if $s < 1$, a wavelet with a frequency higher than the mother wavelet is generated.

Given a spatial series $f(\eta)$, its continuous wavelet transform $i\mathcal{W}[f](x, s)$ depending on the wavelet ψ is a function of two variables where

$$\mathcal{W}[f]: \mathbb{R} \times \mathbb{R}_+^* \rightarrow \mathbb{C},$$

$$(x, s) \mapsto \frac{1}{\sqrt{s}} \int_{-\infty}^{+\infty} f(\eta) \psi^* \left(\frac{\eta - x}{s} \right) d\eta. \quad (5)$$

(*) indicates the complex conjugate. This complex function can also be written as

$$\mathcal{W}[f](x, s) = R(x, s) e^{i\phi(x, s)}, \quad (6)$$

where R is the absolute value (modulus) and ϕ is the phase (argument) at position x with the scale s .

$$R(x, s) = |\mathcal{W}[f](x, s)| \quad (7)$$

$$\phi(x, s) = \text{Im}(\ln \mathcal{W}[f](x, s)) \quad (8)$$

where Im is the imaginary part of $\ln(\mathcal{W}[f](x, s))$. To maintain the nomenclature in the spatial definition and facilitate the extraction of wavelengths, we choose the angular wavenumber (in rad m^{-1}) $k = \frac{2\pi}{\lambda}$ instead of the scale factor. We associate a wavelength $\lambda = 2\pi\alpha s$ with the scale parameter s , where α is the Fourier factor associated with the wavelet, and

$$\alpha = \frac{2}{\beta + \sqrt{2 + \beta^2}}, \quad (9)$$

$$s = \frac{1}{\alpha k}. \quad (10)$$

Thus, the wavelet transform of the function $f(x)$ is defined in the space wavenumber as

$$\mathcal{W}[f]: \mathbb{R} \times \mathbb{R}_+^* \rightarrow \mathbb{C},$$

$$(x, k) \mapsto \sqrt{\alpha k} \int_{-\infty}^{+\infty} f(\eta) \psi^* (\alpha k(\eta - x)) d\eta. \quad (11)$$

Except for the channel angle, all input variables are always positive and may substantially vary in magnitude, so we perform the wavelet transform on the Neperian logarithm of these variables. The whole analysis is performed in a simple *Scilab* script, using the functions that compute the wavelet transform $\mathcal{W}[f]$ and are provided by Torrence and Compo (1998) (<https://atoc.colorado.edu/research/wavelets/>, 11 June 2020). We followed the procedure in Appendix B to compute $\frac{\partial \phi}{\partial x}$ and extract the curves that satisfy Eqs. (12) and (13).

The complex wavelet transform can be classically visualized using a scalogram, i.e., a colored map of the modulus $R(x, k)$ in the (x, k) plane (Fig. 5, bottom). The wavelet analysis neglects parts of the signal at both extremities of the series: this is *the cone of influence* (Torrence and Compo, 1998) that is the region of the wavelet spectrum in which

edge effects become important. However, as explained previously, the complex transform also yields a phase $\phi(x, k)$ in rad (Eq. 8); it can also be plotted in the same plane (Fig. 5, top). In our study, we search for space/wavenumber curves mainly using phase information, i.e., search for phase ridges as opposed to amplitude ridges (Lilly and Olhede, 2010).

In Sect. 3.2, we give a rigorous definition of wavelet ridge points and curves in a univariate case (i.e., a single spatial series). Then, in Sect. 3.3, we generalize the definition to the multivariate, i.e., when the series consists of several correlated variables.

3.2 Univariate case

In the univariate case, we choose a single variable f (velocity, hydraulic radius, bed shear stress, or local channel direction angle). For the wavelet $\psi(\eta)$, the ridge point of $\mathcal{W}[f](x, k)$ is a space–wavenumber pair (x, k) satisfying the *phase ridge point conditions* (Lilly and Olhede, 2010):

$$\frac{\partial}{\partial x} \text{Im}(\ln \mathcal{W}[f](x, k)) - k = 0, \quad (12)$$

or, according to the definition of the phase (Eq. 8):

$$\frac{\partial \phi}{\partial x} \Big|_{(x, k)} - k = 0. \quad (13)$$

This condition states that the rate of change of the transform phase at scale k exactly matches k at location x ; from this condition, the instantaneous frequency of the signal can be derived (Lilly and Olhede, 2008, 2010). The sets of points satisfying the condition form a parametric curve (ridge curve) noted $(x, K(x))$ implicitly defined by

$$\frac{\partial \phi}{\partial x} \Big|_{(x, K(x))} - K(x) = 0. \quad (14)$$

This property is illustrated in Fig. 5, where a ridge curve is superposed both on the scalogram and on the phase map.

There may be several curves that verify Eq. (14); in practice, we choose curves that cross the domain of the wavelet transform (from one cone of influence to another) continuously and belong to the region where the maximum power of the wavelet is. This curve $K(x)$ also represents the local wavenumber, which is defined on a support $\ell < L$ named assessed length, with L the total reach length.

The phase function Φ is then obtained by evaluating the function $\phi(x, k)$ along the curve $(x, K(x))$, in thick black in Fig. B1a.

$$\Phi(x) = \phi(x, K(x)) \quad (15)$$

In the end, we can extract the wavelength function of pool–riffle sequences, which corresponds to a pseudo-period function of the signal f , and which is

$$\lambda(x) = \frac{2\pi}{K(x)}. \quad (16)$$

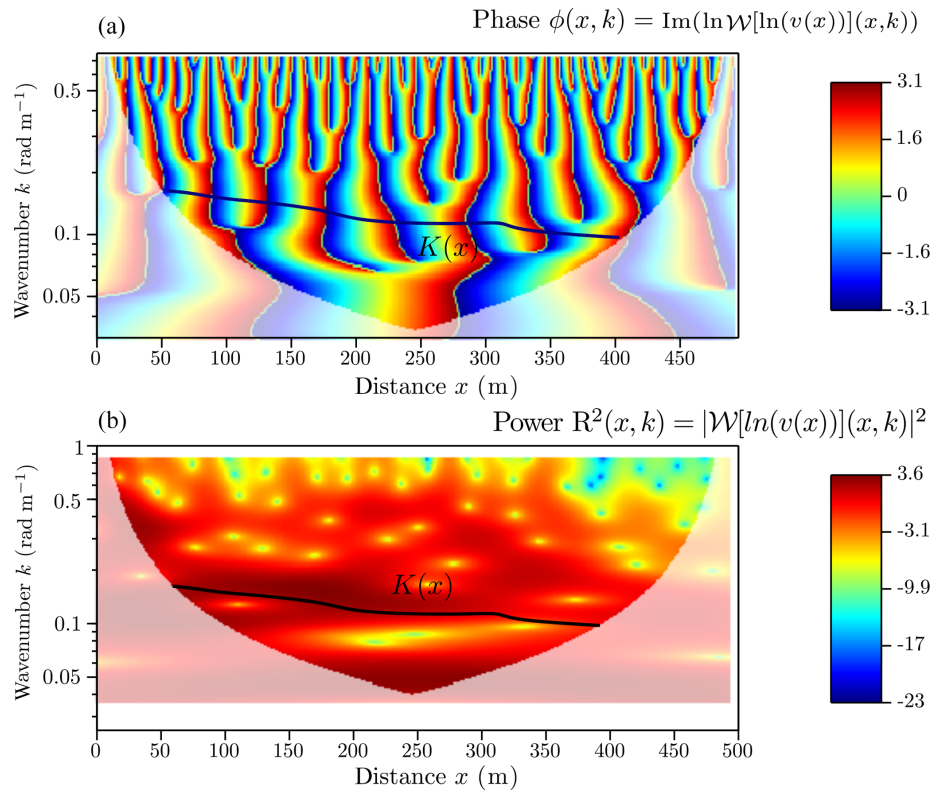


Figure 5. (a) The phase function from which we get the function $K(x)$; (b) the power of the wavelet with the region where there is maximum variability depicted by the black curve $K(x)$ (ridge curve). These two figures are represented in a wavenumber/distance space for the Olivet River, and the wavelet transform is performed on the logarithm of the velocity. The part of the figure with low opacity shows the cone of influence, which is ignored in this study (edge effects are more important for short wavelengths than for long wavelengths).

Also, the shape’s amplitude A_m , with which pools and riffles vary, is corrected by a coefficient $\sqrt{\frac{1}{\alpha K(x)}}$. This correction comes from the inversion of the direct transformation equation (Eq. 11) which holds the coefficient $\sqrt{\alpha K(x)}$.

$$\begin{aligned}
 A_m(x) &= |\mathcal{W}[f](x, K(x))| \sqrt{\frac{1}{\alpha K(x)}} \\
 &= R(x, K(x)) \sqrt{\frac{1}{\alpha K(x)}} \quad (17)
 \end{aligned}$$

The signal is locally similar to a sinusoid f_{mod} of wavenumber K in rad m^{-1} , which models the variability of f . We can define the pseudo-periodic variable as presented in Fig. 6 with

$$f_{\text{mod}}(x) = A_m(x) \cos(\Phi(x)) = A_m(x) \cos(\phi(x, K(x))). \quad (18)$$

In the example below (Fig. 6), the modeled velocity function follows the variability of the observed velocity; it is a pseudo-periodic, continuous function that approximates the first-order variability of this hydraulic parameter across pool–riffle sequences. The statistics of the $K(x)$ function can be translated into statistics of longitudinal spacings of al-

ternating bedforms, e.g., mean spacing λ_{mean}^* , median spacing $\lambda_{\text{median}}^*$ or spacing standard deviation $\sigma(\lambda^*)$. In Fig. 6 we would find $\lambda_{\text{mean}}^* \approx 8.7$, $\lambda_{\text{median}}^* \approx 9.12$ and $\sigma(\lambda^*) \approx 0.79$ if we were to analyze velocity only. The pseudo-periodicity of v_{mod} results in the identification of six pools (white) and seven riffles (gray).

In the next section, we extend the definition of phase ridge points and ridges to the case where several variables are sampled along the reach, all of them potentially correlated and embedding information about the pseudo-periodicity of the channel’s hydraulic behavior.

3.3 Multivariate case

The multivariate case is the extension of the univariate case to a set of N real-valued signals. We use the coevolution of more than one variable to extract the wavelength of the reach and therefore identify the pool–riffle sequences. We start by computing the wavelet transform for each variable $i = 1 \dots N$ and extract their phase functions $\phi_i(x, k)$. According to the previous section, univariate ridge curves $K_i(x)$ would be defined by

$$\left. \frac{\partial \phi_i}{\partial x} \right|_{(x, K_i(x))} - K_i(x) = 0. \quad (19)$$

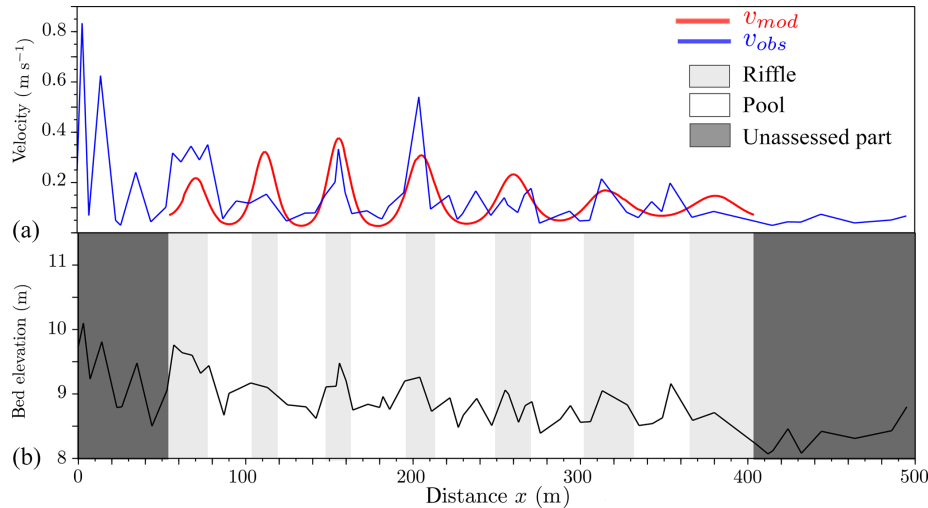


Figure 6. (a) Variation of the modeled function f_{mod} which represents the pseudo-periodic variable (e.g., the velocity of the Olivet River) compared to the observed one. This pseudo-periodicity results in the identification of pools (white) and riffles (gray) in panel (b). The unstudied part is due to the cone of influence of the wavelet method.

But then the local wavenumber would be specific to a given variable. On the other hand, the multivariate case requires one to determine a common wavenumber between all the variables such that

$$\left. \frac{\partial \phi_i}{\partial x} \right|_{(x, K(x))} - K(x) \approx 0 \forall i. \quad (20)$$

The identification of a “master” ridge point/curve is now a minimization problem. We define it as a local minimum of the squared norm of the vector $\left(\left. \frac{\partial \phi_1}{\partial x} \right|_{(x,k)} - k, \left. \frac{\partial \phi_2}{\partial x} \right|_{(x,k)} - k, \dots, \left. \frac{\partial \phi_N}{\partial x} \right|_{(x,k)} - k \right)$:

$$E(x, k) = \sum_{i=1}^N \left(\left. \frac{\partial \phi_i}{\partial x} \right|_{(x,k)} - k \right)^2. \quad (21)$$

This minimum is calculated by searching for the wavenumbers and positions where the derivatives (Eq. 22) of this quantity satisfy

$$\frac{\partial E(x, k)}{\partial k} = \sum_{i=1}^N \left(\left. \frac{\partial^2 \phi_i}{\partial k \partial x} \right|_{(x,k)} - 1 \right) \left(\left. \frac{\partial \phi_i}{\partial x} \right|_{(x,k)} - k \right) = 0$$

and

$$\begin{aligned} \frac{\partial^2 E(x, k)}{\partial k^2} = & \sum_{i=1}^N \left[\left. \frac{\partial^3 \phi_i}{\partial k^2 \partial x} \right|_{(x,k)} \left(\left. \frac{\partial \phi_i}{\partial x} \right|_{(x,k)} - k \right) \right. \\ & \left. + \left(\left. \frac{\partial^2 \phi_i}{\partial k \partial x} \right|_{(x,k)} - 1 \right)^2 \right] > 0. \end{aligned} \quad (22)$$

The procedure is applied to a set of variables $[v, R_h, \tau_b, \theta]$ and the goal is to seek for the common wavenumber between

all these variables. In Fig. 7 we illustrate the result of this procedure applied to the Olivet River for all four variables. A unique wavenumber is extracted, which represents a co-evolution of all these variables.

As a result, the phase shift of every variable is calculated by

$$\Phi_i(x) = \phi_i(x, K(x)). \quad (23)$$

This ridge curve $K(x)$ is common between all variables, yet Φ_i varies according to each variable. Therefore, each one can be represented as a pseudo-periodic function $f_{i,mod}$ with the pair $(K(x), \Phi_i(x))$.

In our case, after calculating the phase and amplitude, we modeled each variable as in Eq. (24) and represented them in Fig. 8.

$$\begin{aligned} f_{i,mod}(x) &= A_{i,m}(x) \cos(\Phi_i(x)) \\ &= A_{i,m}(x) \cos(\phi_i(x, K(x))) \end{aligned} \quad (24)$$

The amplitude shape of the modeled variable is calculated by the same way in the univariate case:

$$A_{i,m}(x) = |\mathcal{W}[f_i](x, K(x))| \sqrt{\frac{1}{\alpha K(x)}}. \quad (25)$$

The results in Fig. 8 show that a common pseudo-period has been successfully identified and allows consistent pseudo-periodic representation of all four variables.

Figure 9 shows the correlations between these variables: it can be seen that the pseudo-periodic reconstruction preserves the correlation structure between the three flow variables; an anti-correlated hydraulic radius with bed shear stress and velocity and a strong correlation between bed shear stress and

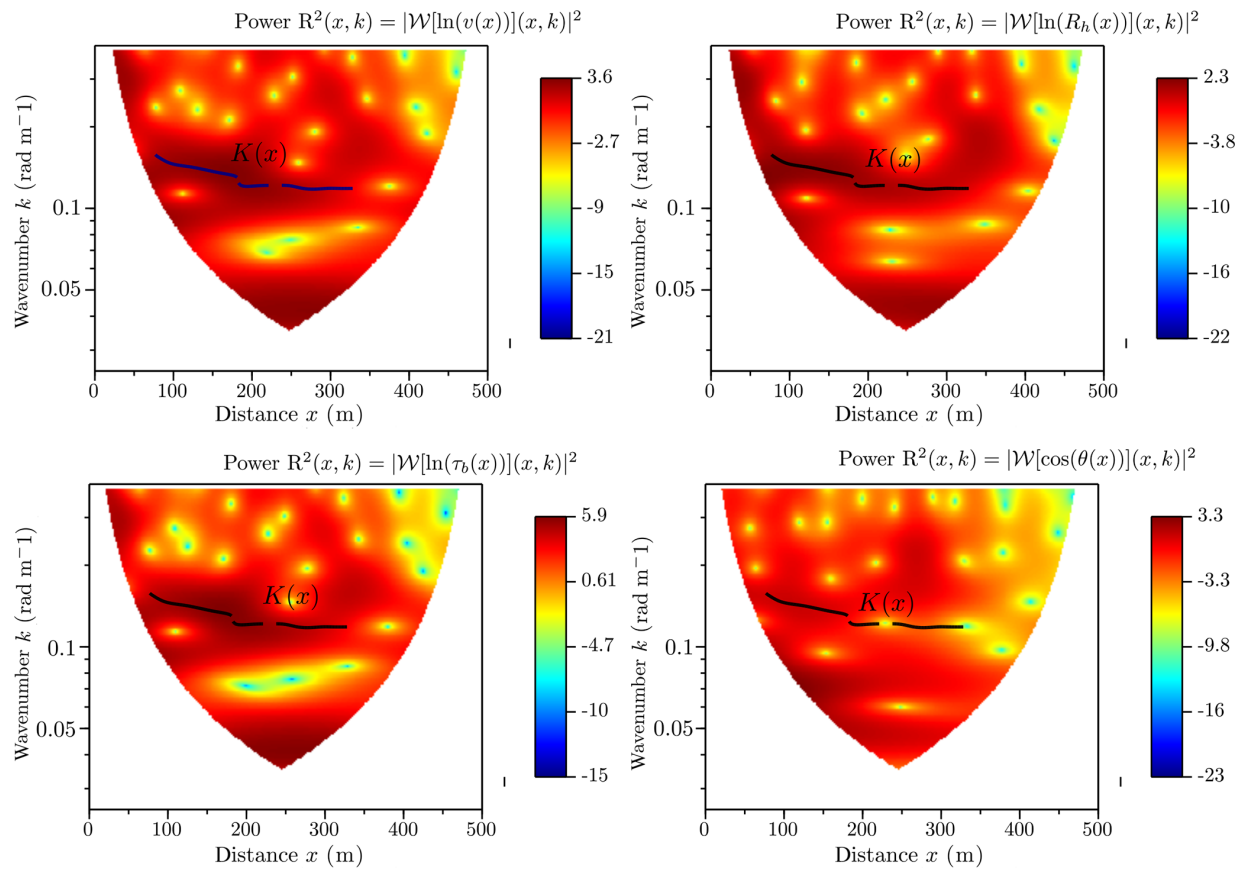


Figure 7. Power of the wavelet of the four variables: velocity, hydraulic radius, bed shear stress, and local channel direction angle. The black curve $K(x)$ is the extracted ridge curve of the Olivet River in the multivariate case.

velocity. However, concerning the angle, the results show a small phase shift, which is corrected in the following x positions. But generally, a deviation (clockwise or counter-clockwise) from the average direction of the channel (i.e., $\cos(\theta)$ much smaller than 1) is associated with a low hydraulic radius and large values of τ_b and v , a consistent characterization of a riffle. These pseudo-periodic functions give us an identification of reach features: pools (in white) and riffles (in gray).

As already mentioned in Sect. 3.2 (univariate analysis), the statistics of the $K(x)$ function can be translated into statistics of local wavelength $\lambda(x) = \frac{2\pi}{K(x)}$, which can, in turn, be interpreted as statistics of longitudinal spacings of alternating bedforms, e.g., mean spacing λ_{mean}^* , median spacing $\lambda_{\text{median}}^*$ or spacing standard deviation $\sigma(\lambda^*)$. In the example of the Olivet River (Fig. 9) $\lambda_{\text{mean}}^* \approx 8.16$, $\lambda_{\text{median}}^* \approx 8.62$ and $\sigma(\lambda^*) \approx 0.70$. The pseudo-periodicity of the set $[v_{\text{mod}}, R_{h,\text{mod}}, \tau_{b,\text{mod}}, \cos(\theta)_{\text{mod}}]$ results in the identification of five pools and five riffles.

4 Results

In this section, we present the results of the analysis on the six reaches presented in Sect. 2. We compare the univariate to the multivariate approach and also the multivariate to the benchmark method. First, the methods are compared in terms of the statistics (mean, median, etc.) they yield. Second, we present the benchmark method called the BDT and compare its results for the six reaches with the multivariate case.

4.1 Univariate vs. multivariate

First, both approaches are employed on all reaches to extract statistics such as the mean, median, and standard deviation wavelengths of morphological units (pool–riffle sequences). The wavelet method extracts the wavelength for an assessed length ℓ (which is the $K(x)$ support in Figs. 6 and 9) that is generally small compared to the total length of the reach. Consequently, we have results that are valuable only for the lengths shown in Table 3. In this table, we give the values of the assessed lengths for each approach, including the variables used in it. These values generally depend on the number of alternating bedforms and also on the total length of

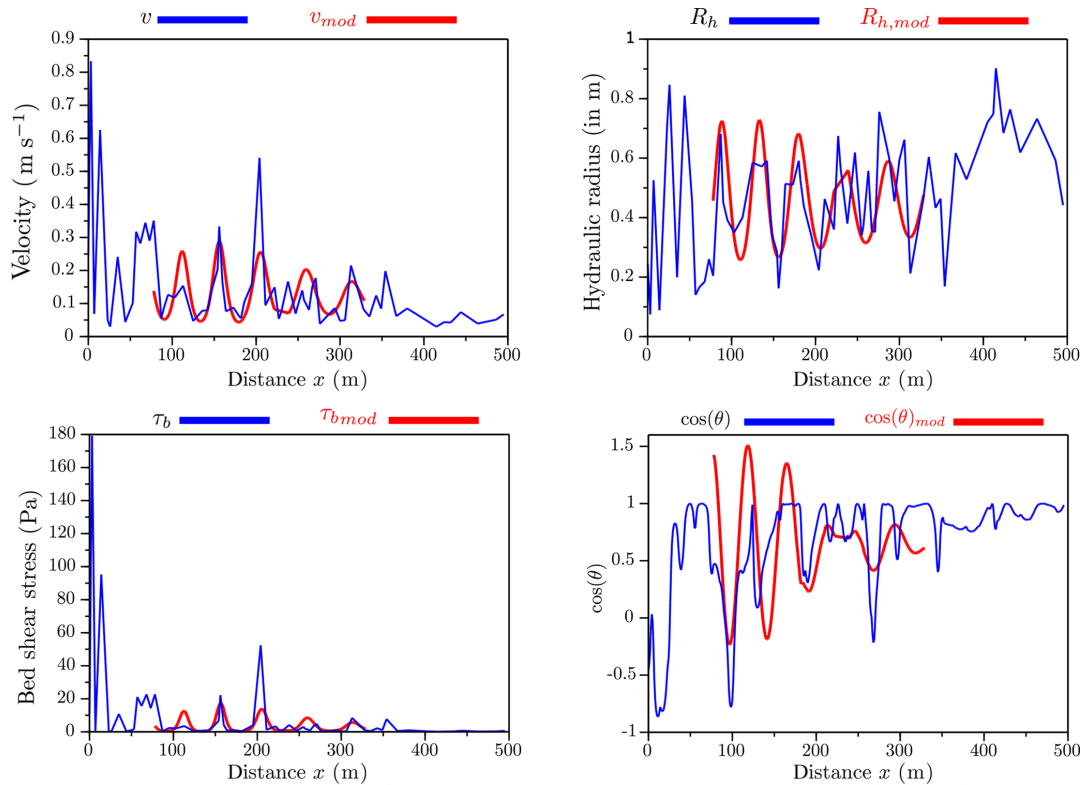


Figure 8. Variation of the modeled function $f_{i,mod}$, which represents the pseudo-periodic variable (in red) for the velocity, the hydraulic radius, the bed shear stress, and the local channel direction angle of the Olivet River compared to the observed ones (in blue).

the reach. The greater the number of alternating bedforms and the reach length are, the greater the assessed length is.

Moreover, the multivariate approach takes into account all the variables and therefore looks for a single pseudo-periodicity between the four variables, and then we are going to have a pseudo-periodicity that represents the reach and not the chosen variable.

Figure S1 in the Supplement shows the comparison between the univariate and multivariate results for the six reaches (from 1 to 6) using the four variables (velocity, hydraulic radius, bed shear stress, and cosine of local channel direction angle). Table 4 gives some statistics on both approaches. Longitudinal spacing is calculated using the wavelengths extracted automatically by the wavelet ridge method from $K(x)$.

We compare the methods in terms of longitudinal spacing (λ^*). In each reach, there seems to be one variable which drives the wavelength identified in the multivariate approach:

- in the Graulade River, the longitudinal spacing identified using the multivariate approach matches closely the one associated with the hydraulic radius (in the mean and the median with a deviation of $0.05w_{bf}$) and also with the local channel direction angle (in the median with a deviation of $0.06w_{bf}$);

- in the Semme River, it matches those of the local channel direction angle (in the mean and the median with a deviation of $0.14w_{bf}$ and $0.12w_{bf}$ consecutively);
- in the Olivet River, it matches the bed shear stress (in the mean with a deviation of $0.25w_{bf}$) and the velocity (in the median with a deviation of $0.5w_{bf}$);
- in the Ozanne River, it matches those of the hydraulic radius and the velocity (in the mean and the median with a deviation less than $0.6w_{bf}$);
- in the Avenelles, it matches those of the velocity, hydraulic radius, and bed shear stress (in the mean with a deviation less than $0.15w_{bf}$);
- in the Orgeval River, it matches those of the hydraulic radius (in the mean with a deviation of $0.28w_{bf}$ and the median with $0.06w_{bf}$) and also with the local channel direction angle (in the mean with a deviation of $0.23w_{bf}$ and in the median with $0.11w_{bf}$).

Consequently, the multivariate estimates of λ^* compare with univariate estimates in a similar way:

- the distribution of λ^* in the multivariate case is included in the envelope of univariate distributions;

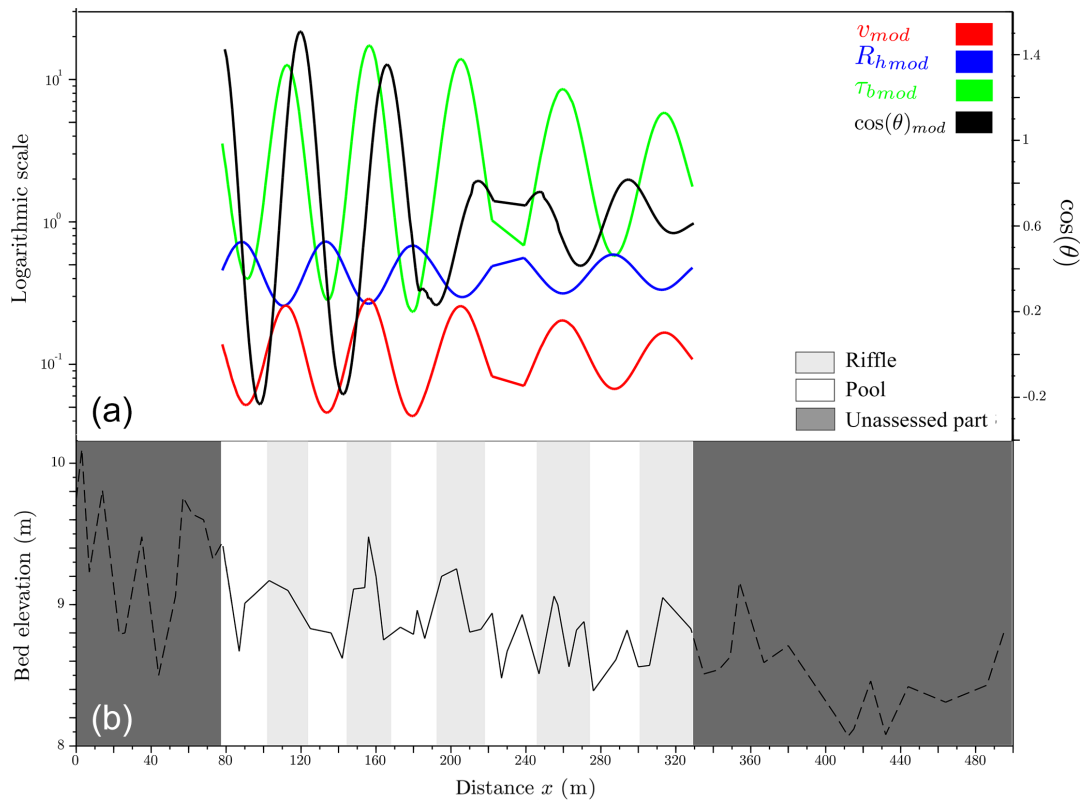


Figure 9. (a) Correlation between the modeled functions $f_{i,mod}$, which represents the pseudo-periodic variables (velocity in red, hydraulic radius in blue, bed shear stress in green, and local channel direction angle in black) of the Olivet River. These pseudo-periodic functions result in the identification of pools (white) and riffles (gray) in panel (b).

Table 3. The assessed length provided by the wavelet analysis for all reaches in the univariate case using the velocity, hydraulic radius, bed shear stress, or local channel direction angle and in the multivariate case using all these four variables.

Reaches	Reach length (m)	Assessed length ℓ (m) (univariate)				Assessed length ℓ (m) (Multivariate) ($v, R_h, \tau_b, \cos(\theta)$)
		Velocity	Hydraulic radius	Bed shear stress	Local channel direction angle	
(1) Graulade	160	88	67	72	102	67
(2) Semme	177	87	70	89	110	37
(3) Olivet	495	349	366	363	365	251
(4) Ozanne	319	215	157	151	125	77
(5) Avenelles	155	76	70	79	64	60
(6) Orgeval	318	142	200	163	140	158

– the dispersion of this multivariate distribution, measured by $\sigma(\lambda^*)$, is always close to the minimum value that can be achieved by any of the univariate distributions.

Hence, the multivariate method improves the identification of the wavelength: it is less sensitive to a local high-frequency variation of a given variable if this variation is not associated with a variation of the other variables. However, there is

no direct way of validating the estimates from these raw results: a way of doing so would be to build a synthetic, equivalent periodic geometry parameterized by the identified wavelength to verify that it yields, for example, a similar reach-average rating curve. This will be the subject of further work.

In the following section, we compare the wavelet method with a benchmark method using talweg elevation.

Table 4. Summary of results for all reaches in the univariate case using the velocity, hydraulic radius, bed shear stress, or local channel direction angle and in the multivariate case using all these four variables. For each variable, we compute the mean, median, and standard deviation σ of the wavelength and the longitudinal spacing. This one (λ^*) is calculated by $\frac{\lambda}{w_{bf}}$, and w_{bf} is taken from Table 2. Bold values represent the median or the mean wavelengths that are similar in both the univariate case and the multivariate case in a reach.

			(1) Graulade	(2) Semme	(3) Olivet	(4) Ozanne	(5) Avenelles	(6) Orgeval	
Univariate	Velocity	λ (m)	Mean	23.47	13.86	52.20	37.18	27.32	51.66
			median	24.17	13.93	54.74	36.74	27.17	51.29
			$\sigma(\lambda)$	1.69	0.53	7.75	2.97	1.60	1.70
		λ^* (m)	Mean	5.87	1.15	8.70	3.10	3.03	5.17
			median	6.04	1.16	9.12	3.06	3.02	5.13
			$\sigma(\lambda^*)$	0.42	0.04	1.29	0.25	0.18	0.17
	Hydraulic radius	λ (m)	Mean	21.74	39.28	47.19	37.73	25.72	45.46
			median	21.41	39.43	46.60	38.47	25.47	48.23
			$\sigma(\lambda)$	0.71	1.19	4.74	2.40	0.66	8.73
		λ^* (m)	Mean	5.43	3.27	7.86	3.14	2.86	4.55
			median	5.35	3.29	7.76	3.20	2.83	4.82
			$\sigma(\lambda^*)$	0.18	0.10	0.79	0.20	0.07	0.87
Bed shear stress	λ (m)	Mean	26.07	32.29	47.47	36.43	27.47	51.70	
		median	25.92	32.66	45.54	36.30	27.95	51.26	
		$\sigma(\lambda)$	1.12	1.68	5.36	1.70	0.73	1.54	
	λ^* (m)	Mean	6.52	2.69	7.91	3.04	3.05	5.17	
		median	6.48	2.72	7.59	3.02	3.11	5.13	
		$\sigma(\lambda^*)$	0.28	0.14	0.89	0.14	0.09	0.15	
Cosine of local channel direction angle	λ (m)	Mean	21.14	23.45	40.87	66.31	28.79	50.58	
		median	21.32	23.30	39.44	62.98	28.73	49.93	
		$\sigma(\lambda)$	0.75	0.95	3.57	7.49	1.47	4.35	
	λ^* (m)	Mean	5.28	1.95	6.81	5.52	3.20	5.06	
		median	5.33	1.94	6.57	5.25	3.19	4.99	
		$\sigma(\lambda^*)$	0.19	0.08	0.60	0.62	0.16	0.43	
Multivariate ($v, R_h, \tau_b, \cos(\theta)$)	λ (m)	Mean	21.54	21.74	48.98	43.89	26.59	48.29	
		median	21.55	21.84	51.70	43.49	26.54	48.78	
		$\sigma(\lambda)$	0.38	0.85	4.22	0.98	0.40	3.42	
	λ^* (m)	Mean	5.38	1.81	8.16	3.66	2.95	4.83	
		median	5.39	1.82	8.62	3.62	2.95	4.88	
		$\sigma(\lambda^*)$	0.09	0.07	0.70	0.08	0.04	0.34	

4.2 Comparison with benchmark method

In this section, we compare our method's results with a selected benchmark method from the literature (i.e., BDT). This method shows good results in the identification of these bedforms according to some studies (e.g., Frothingham and Brown, 2002; Krueger and Frothingham, 2007).

The technique of O'Neill and Abrahams (1984) (BDT) uses a tolerance value (T), which defines the minimum absolute value needed to identify a pool or a riffle (Krueger and Frothingham, 2007). It is calculated using the standard deviation SD of the series of bed elevation differences from upstream to downstream for each reach and corrected by a coefficient chosen according to the reach. For this, we test several tolerance values, and for the Graulade (1), Ozanne (4), Avenelles (5), and Orgeval (6) reaches, we find the same re-

sults. We choose to check one tolerance value for each reach with $T = SD$. This method identifies pool and riffle positions by assigning a crest as a riffle and a bottom as a pool, and therefore the computation of the wavelengths becomes a little difficult. So, we choose to calculate a series of pool–pool and riffle–riffle spacings, their medians, and their standard deviations and then calculate their averages.

This procedure is applied to all rivers, and the results are depicted in Fig. 10. Table 5 presents statistics of the BDT and displays a comparison between these two types of morphological unit identification and mostly the identification of an average wavelength of the reach.

Figure 10 shows the BDT results on all reaches; this method relies only on topography to determine the positions of pools and riffles. Moreover, it also uses a threshold T (tolerance), but the technique does not need a calibration reach

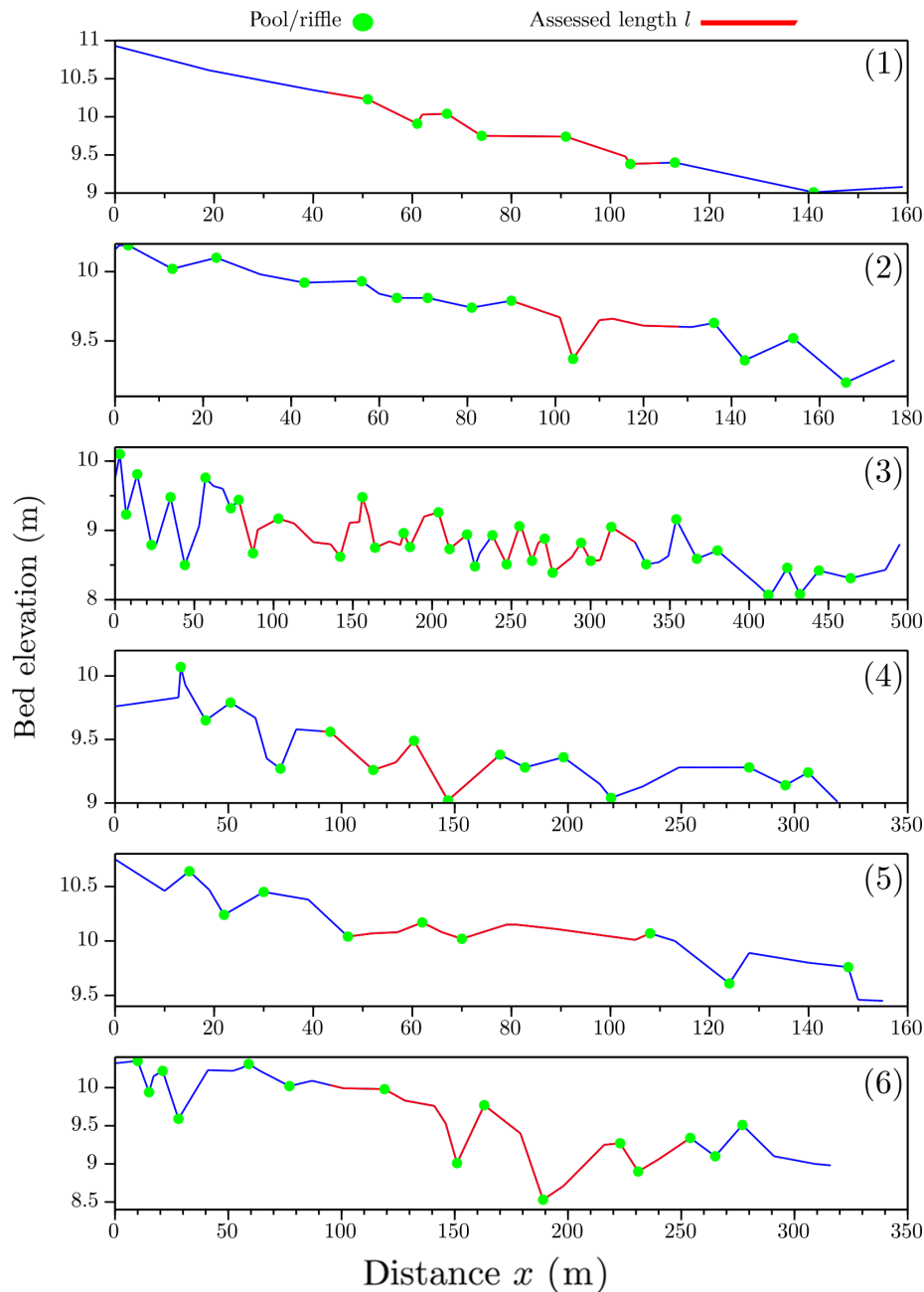


Figure 10. Results of the BDT method using a tolerance equal to the standard deviation on the total length (blue) and the assessed one (red) for all reaches (1 to 6). Round points are pools or riffles: pools are high, and riffles are low points.

or field investigation to know how to set this threshold. In this figure, round points are pools or riffles, and we can calculate the wavelengths and longitudinal spacing of each reach using positions of these points.

The work of the wavelet analysis is done on the assessed length ℓ . However, the BDT method works on the total length of the reach. This comparison is made to determine how effectively the wavelength extracted by the wavelet analysis can represent the entire reach even if an entire part is left unassessed.

For the wavelet method (Fig. 9), the wavelength extraction is among its objectives, while the BDT does not directly calculate the wavelength. It is computed by averaging the pool-to-pool and riffle-to-riffle distances. To compare these two methods, we use only the longitudinal spacing (λ^*) as a criterion.

In Table 5, we present the results of the BDT on the total length L and the assessed length ℓ of all reaches. By using the total length L , the longitudinal spacings found with the BDT are close to the ones found with the wavelet analysis

Table 5. Results of BDT and the multivariate wavelet methods for all reaches. $\lambda_{\text{meth,mean}}$ is the mean wavelength using one of the methods (for BDT, it is used on the total length and the assessed one ℓ). $\lambda_{\text{meth,median}}$ is the median, and $\lambda_{\text{meth,mean}}^*$ is the mean longitudinal spacing, $\lambda_{\text{meth,median}}^*$ is the median, and $\sigma(\lambda_{\text{meth,median}})$ is the standard deviation of the longitudinal spacing. (–) means that we find only one longitudinal spacing, which is the mean and the median, so there is no standard deviation.

Reaches	(1) Graulade	(2) Semme	(3) Olivet	(4) Ozanne	(5) Avenelles	(6) Orgeval
Total length L (m)	160.0	177.0	495.0	319.0	155.0	318.0
Assessed length ℓ (m)	67.0	37.0	251.0	77.0	60.0	158.0
$\lambda_{\text{wav,mean}}$ (m)	21.54	21.74	48.98	43.89	26.59	48.29
$\lambda_{\text{BDT,mean}}$ (m)	23.67	25.33	24.94	41.12	33.63	39.90
$\lambda_{\ell\text{BDT,mean}}$ (m)	20.75	–	28.00	35.00	–	46.00
$\lambda_{\text{wav,mean}}^*$	5.38	1.81	8.16	3.66	2.95	4.83
$\lambda_{\text{BDT,mean}}^*$	5.92	2.11	4.16	3.43	3.74	3.99
$\lambda_{\ell\text{BDT,mean}}^*$	5.19	–	4.66	2.92	–	4.60
$\lambda_{\text{wav,median}}$ (m)	21.55	21.84	51.70	43.49	26.54	48.78
$\lambda_{\text{BDT,median}}$ (m)	26.00	21.25	21.75	36.50	30.50	39.00
$\lambda_{\ell\text{BDT,median}}$ (m)	20.75	–	23.50	35.00	–	46.00
$\lambda_{\text{wav,median}}^*$	5.39	1.81	8.62	3.62	2.95	4.88
$\lambda_{\text{BDT,median}}^*$	6.5	1.77	3.63	3.04	3.39	3.90
$\lambda_{\ell\text{BDT,median}}^*$	5.19	–	3.92	2.92	–	4.60
$\sigma(\lambda_{\text{wav,mean}}^*)$	0.09	0.07	0.70	0.08	0.04	0.34
$\sigma(\lambda_{\text{BDT,mean}}^*)$	2.06	0.82	1.83	1.56	1.71	1.91
$\sigma(\lambda_{\ell\text{BDT,mean}}^*)$	2.21	–	2.30	–	–	0.71

(deviation less than 1 times the bankfull width for the median) in all the reaches except the Olivet (deviation of $4w_{\text{bf}}$). Over the assessed length ℓ , we find very similar results with deviations less than 1 times the bankfull width. However, the shortening of the length ($\ell < L$) reduces the number of pools and riffles identified – Graulade (1) and Avenelles (5) – and therefore introduces bias. This indicates that a length greater than two cycles (pool–riffle) is always required to produce a pseudo-periodicity of the reach by both methods, a condition which is not fulfilled for all reaches of our dataset. But for the other rivers except for the Olivet (3) and Orgeval (6) reaches, there is not much improvement if we replace the total length with the assessed one. In this comparison, we found that the wavelengths extracted by the multivariate wavelet analysis are generally included in the variance intervals of the wavelengths found by the BDT. This conclusion is verified in all reaches except the Olivet River (3), where there is a big difference between the longitudinal spacings found by the BDT and by wavelets. This difference is due to the choice of the tolerance value, which is low in our case to the point of not filtering out the high-frequency variability of bed elevation and therefore gives a lower periodicity compared to the wavelets.

5 Discussion

In this study, we consider the BDT method to be a benchmark method. We do not consider a specific method to be the “true” one, and we only apply these methods to have a general idea on the uncertainties in the identification of morphological units. This method was chosen not because it is the “best” method for pool–riffle identification, but because it does not use thresholds (except for the tolerance T , which does not depend on field data). It means that it does not require a preliminary calibration of thresholds on velocity, hydraulic radius, etc. on an independent reach (e.g., Wyrick and Pasternack, 2014; Hauer et al., 2009). These thresholds vary from one reach to another and according to the characteristics of each river. For this reason, we did not compare our method with threshold methods on this dataset. In contrast, the results of the longitudinal spacing intervals will be compared with the literature.

For a long time, researchers have found a common interval of longitudinal spacings that vary between 5 and 7 times the channel width (Leopold et al., 1964; Keller, 1972; Richards, 1976a; Gregory et al., 1994). Keller (1972) found that the median is less and varies between 3 and 5 times the channel width. O’Neill and Abrahams (1984), using the BDT method, found the same results but with a median close to 3 times the channel width, and this value can vary according to the tolerance T . Carling and Orr (2000) found lower values

than before at about $3w$. Recent studies (e.g., Wyrick and Pasternack, 2014) have calculated the longitudinal spacing of six morphological units using 2D identification methods. The averages of these pool and riffle spacings are, respectively, 3.3 and 4.3 times the channel width, which is less than the commonly accepted values of $5-7w$.

In this study, the longitudinal spacing varies in the mean and the median from ~ 1.8 to 8.6 times the bankfull width, supporting the conclusion of Carling and Orr (2000) that pools are spaced approximately 3 to 7 times the channel width. However, the quoted longitudinal spacing relationships should be considered in the context that the bankfull width and spacing distance are inherently variable even for short length reaches. To illustrate this inherent variability, we found the example of Keller and Melhorn (1978), where the pool–pool spacing values ranged from 1.5 to 23.3 channel widths, with an overall mean of 5.9 (Gregory et al., 1994; Knighton, 2014). This variability in longitudinal spacing is probably related to a short assessed length, a small number of cross sections surveyed, or other factors such as geology, bank characteristics (cohesion), the grain size of the river bed, or artificial channel modifications.

We worked with a dataset that contains cross sections spaced 0.46 to 2.9 times the bankfull width. Other studies have used much shorter spacings (e.g., Pasternack and Arroyo, 2018; Legleiter, 2014) to identify morphological units. Of course, the larger the number of cross sections, the more robust the identified correlations are. Also, we worked with irregularly spaced cross sections, which normally lead to biases in the results. Despite this, the “biased” placement does not impair the overall methodology. This methodology has provided good results in terms of longitudinal spacings, and therefore it can be applied for shorter cross-section spacings to clearly identify these alternate morphologies. The short lengths we found raise questions about the *naturality* of the rivers. In our case, the rivers are subject to artificial modifications (e.g., bridges, weirs) and rehabilitations, which will have a significant impact on the hydro-morphological parameters (width, depth, meandering, etc.). This can have a very important impact on the identification of pseudo-periods.

The wavelet ridge analysis is powerful in identifying pseudo-periods, amplitude, and phase while preserving the correlations between parameters. We can thus identify alternating morphological units more objectively in terms of frequency/wavenumber.

This wavelength can be used to represent the variability of the bathymetry in hydraulic models in cases where we do not have full access to the geometry of the channel (e.g., remote sensing data such as the overcoming Surface Water and Ocean Topography Mission) and the morphology can be modeled by pseudo-periodic functions. Furthermore, it can be implemented in synthetic geometry generators (e.g., River Builder, Pasternack and Zhang, 2020) where the bathymetry and sinuosity wavelengths extracted by the wavelets can be used to model meandering rivers with alternating morpholo-

gies. Ultimately, hydraulic modeling will be the true test of the potential of a pseudo-periodic equivalent geometry (e.g., for simulating a reach-average rating curve).

On the other hand, it presents drawbacks compared to other methods. First, the cone of influence ignores a large part of the river and sometimes biases the results in the case of small total lengths (the Graulade (1) and Semme (2) reaches). Similarly for reach length and number of morphological units as for the number of cross sections, the larger it is, the more robust the results are, and the smaller the relative portion of “unassessed length” is. Still, the method remains a powerful tool for non-stationary analysis. Another problem is the amplitude, which is sometimes overestimated in some regions of the topography. We visualized this in several cases in our study since we used the Neperian logarithm to avoid negative values, and therefore the inverse function (exponential) gives slightly larger values. However, this does not bias the identified wavelength of the reach.

6 Conclusions

In this study, we present an automatic procedure based on wavelet ridge extraction to identify some characteristics of alternating morphological units (MUs), such as their longitudinal spacing and amplitude. The method does not rely on any a priori thresholds to identify MU sequences. It was applied to six rivers with a maximum length of 500 m. We chose to work with classical hydro-morphological variables (velocity, hydraulic radius, bed shear stress) in addition to the planform channel direction angle that evaluates the impact of river sinuosity in the determination of the wavelength.

As a result, identified wavelengths are consistent with the values of the literature (mean in $3-7w_{bf}$). The use of a multivariate approach yields more robust results than the univariate approaches by ensuring a consistent covariance of flow variables in the pseudo-periodic behavior.

Given the short length of several reaches, the relatively small number of cross sections for each reach, and the possible impacts of artificial modifications, this paper is mainly a proof-of-concept of the wavelet approach. We foresee many perspectives for this work, such as the possibility of extending the work to other rivers with other types of MUs or other longer reaches with a large number of cross sections.

Appendix A: List of symbols

$A(x)$	Cross-sectional area (m^2)
$A_{i,\text{mod}}$	Signal amplitude of the shape of the modeled variable number i
A_m	Signal amplitude of the shape
$\cos(\theta)$	Cosine of the local channel direction angle
$\cos(\theta)_{\text{mod}}$	Modeled cosine of the local channel direction angle
f	Space series function (m)
f_i	Measured space series function number i
$f_{i,\text{mod}}$	Modeled space series function number i with multivariate wavelet analysis
f_{mod}	Modeled variable with the univariate wavelet analysis
g	Acceleration due to gravity and its value is 9.81 m s^{-2}
$k(x)$	Wavenumber (rad m^{-1})
$K(x)$	Local wavenumber that corresponds to the maximum variance of the signal (rad m^{-1})
$K_i(x)$	Local wavenumber that corresponds to the maximum variance of the variable i (rad m^{-1})
ℓ	$K(x)$ support (m)
L	Total reach length (m)
N	Number of total chosen variables
n	Manning's roughness coefficient
$P(x)$	Wetted perimeter (m)
Q_{min}	Minimum discharge modeled ($\text{m}^3 \text{ s}^{-1}$)
$R(x, s)$ or $R(x, k)$ or R	Absolute value or modulus of the wavelet transform at a position x and with a scale s or wavenumber k
R_h	Hydraulic radius (m)
$R_{h,\text{mod}}$	Modeled hydraulic radius (m)
s	Dilation or scale factor
S	Reach slope (m m^{-1})
SD	The standard deviation of the bed elevation difference series (m)
T	Tolerance value used in BDT method (m)
$v(x)$	Velocity (m s^{-1})
v_{mod}	Modeled velocity (m s^{-1})
v_{obs}	Measured velocity (m s^{-1})
w	Reach width (m)
w_{bf}	Reach bankfull width (m)
w_m	Mean bankfull width (m)
x	Translation factor in the wavelet transform or the abscissa position (m)
$y = y_{\text{max}}$	Water depth measured from the talweg elevation $y = z_{\text{ws}} - z$ (m)
y_m	Mean depth (m)
z	Measured bed elevation or talweg elevation (m)
z_{ws}	Water surface elevation measured from the ONGF (m)
α	Fourier factor associated with the wavelet (m rad^{-1})
β	Dimensionless frequency is taken to be 6 recommended by Torrence and Compo (1998)
λ	Reach wavelength (m)
λ^*	Typical pool (riffle) spacing or dimensionless reach wavelength
ρ	Water density (997 kg m^{-3})
θ	local channel direction angle in rad
$\sigma(w)$	The standard deviation of the width along the reach (m)
$\sigma()$	Standard deviation
$\tau_b(x)$	Bed shear stress in the x abscissa (Pa)
$\tau_{b,\text{mod}}(x)$	Modeled bed shear stress in the x abscissa (Pa)
Φ	Corresponding phase at the position x and the wavenumber K with $\Phi(x) = \phi(x, K(x))$ (rad)
Φ_i	Phase at the position x and the wavenumber K for the variable number i
$\phi(x, s)$ or $\phi(x, k)$ or ϕ	Phase or argument at a position x and with a scale s or wavenumber k (rad)
ϕ_i	The phase of the variable number i

ψ	Mother wavelet function
$\psi_{s,x}$	Daughter wavelet function
η	Dimensionless position parameter
\mathbb{C}	Complex numbers
\mathbb{R}	Real numbers
\mathbb{R}_+^*	Positive real numbers
$\mathcal{W}[f](x, s)$	Continuous wavelet transform of $f(x)$ with the wavelet ψ

Appendix B: Mathematical calculus for the wavelet transform

(B4)

B1 The univariate case

The conjugate form of the mother wavelet is

$$\psi(\eta) = \pi^{-\frac{1}{4}} e^{-i\beta\eta - \frac{\eta^2}{2}}. \tag{B1}$$

Its derivative depending on the mute variable η is

$$\psi'^*(\eta) = -\pi^{-\frac{1}{4}} (i\beta + \eta) e^{-i\beta\eta - \frac{\eta^2}{2}} \tag{B2}$$

$$= -(i\beta + \eta) \psi^*(\eta). \tag{B3}$$

In Sect. 3.1, η is a mute integration variable, and x appears only in the argument $\alpha k(\xi - x)$ of the function ψ^* . By applying the derivation formula of a composite function, the derivative of the wavelet transform is expressed by

$$\begin{aligned} \frac{\partial}{\partial x} \mathcal{W}[f(x)](x, k) &= \sqrt{\alpha k} \int_{-\infty}^{+\infty} f(\eta) \frac{\partial}{\partial x} [\psi^*(\alpha k(\eta - x))] d\eta \\ &= \sqrt{\alpha k} \int_{-\infty}^{+\infty} f(\eta) \cdot (-\alpha k) \cdot \psi'^*(\alpha k(\eta - x)) d\eta \\ &= (\alpha k) \sqrt{\alpha k} \int_{-\infty}^{+\infty} f(\eta) \cdot (i\beta + \alpha k(\eta - x)) \cdot \psi^*(\alpha k(\eta - x)) d\eta \\ &= (\alpha k) \sqrt{\alpha k} \int_{-\infty}^{+\infty} [(i\beta - \alpha kx) f(\eta) + \alpha k\eta f(\eta)] \cdot \psi^*(\alpha k(\eta - x)) d\eta \\ &= (\alpha k)(i\beta - \alpha kx) \mathcal{W}[f(x)](x, k) + (\alpha k)^2 \mathcal{W}[xf(x)](x, k). \end{aligned}$$

On the other hand, we have

$$\begin{aligned} \frac{\partial}{\partial x} \mathcal{W}[f(x)](x, k) &= \frac{\partial}{\partial x} \left(R(x, k) e^{i\phi(x, k)} \right) \\ &= \left[\frac{1}{R} \frac{\partial R}{\partial x} + i \frac{\partial \phi}{\partial x} \right] R e^{i\phi}, \\ \frac{\partial}{\partial x} \operatorname{Re}(\ln \mathcal{W}[f(x)](x, k)) &= \frac{1}{R} \frac{\partial R}{\partial x} \\ &= \operatorname{Re} \left(\frac{1}{\mathcal{W}[f(x)](x, k)} \frac{\partial}{\partial x} \mathcal{W}[f(x)](x, k) \right), \\ \frac{\partial}{\partial x} \operatorname{Im}(\ln \mathcal{W}[f(x)](x, k)) &= \frac{\partial \phi}{\partial x} \\ &= \operatorname{Im} \left(\frac{1}{\mathcal{W}[f(x)](x, k)} \frac{\partial}{\partial x} \mathcal{W}[f(x)](x, k) \right). \end{aligned}$$

Finally,

$$\begin{aligned} \frac{\partial \phi}{\partial x} &= \operatorname{Im} \left\{ (\alpha k)(i\beta - \alpha kx) + (\alpha k)^2 \frac{\mathcal{W}[xf(x)](x, k)}{\mathcal{W}[f(x)](x, k)} \right\}, \\ \frac{\partial \phi}{\partial x} &= (\alpha k)\beta + (\alpha k)^2 \operatorname{Im} \left\{ \frac{\mathcal{W}[xf(x)](x, k)}{\mathcal{W}[f(x)](x, k)} \right\}. \end{aligned}$$

The previous expression numerically avoids the derivative of the function $\phi(x, k)$, which varies quickly for large wavenumbers.

B2 The multivariate case

In the multivariate case, we should resolve Eq. (20), which contains three derivatives to compute. The first one is already done in the univariate case:

$$\frac{\partial \phi_i(x, k)}{\partial x} = (\alpha k)\beta + (\alpha k)^2 \operatorname{Im} \left\{ \frac{\mathcal{W}[x f_i(x)](x, k)}{\mathcal{W}[f_i(x)](x, k)} \right\}. \tag{B5}$$

The second one is the computation of $\frac{\partial^2 \phi_i(x, k)}{\partial k \partial x}$:

$$\begin{aligned} \frac{\partial^2 \phi_i(x, k)}{\partial k \partial x} &= (\alpha\beta) + 2\alpha^2 k \operatorname{Im} \left\{ \frac{\mathcal{W}[x f_i(x)](x, k)}{\mathcal{W}[f_i(x)](x, k)} \right\} \\ &+ (\alpha k)^2 \operatorname{Im} \left\{ \frac{\partial}{\partial k} \left(\frac{\mathcal{W}[x f_i(x)](x, k)}{\mathcal{W}[f_i(x)](x, k)} \right) \right\}. \end{aligned} \tag{B6}$$

For that, we should develop $\frac{\partial}{\partial k} \left(\frac{\mathcal{W}[x f_i(x)](x, k)}{\mathcal{W}[f_i(x)](x, k)} \right)$:

$$\begin{aligned} \frac{\partial}{\partial k} \left(\frac{\mathcal{W}[x f_i(x)](x, k)}{\mathcal{W}[f_i(x)](x, k)} \right) &= \frac{1}{\mathcal{W}[f_i(x)](x, k)} \\ \frac{\partial \mathcal{W}[x f_i(x)](x, k)}{\partial k} &- \frac{\mathcal{W}[x f_i(x)](x, k)}{\mathcal{W}[f_i(x)](x, k)^2} \\ \frac{\partial \mathcal{W}[f_i(x)](x, k)}{\partial k}. \end{aligned} \tag{B7}$$

We calculate each derivative:

$$\begin{aligned} \frac{\partial \mathcal{W}[f_i(x)](x, k)}{\partial k} &= \left(\frac{1}{\sqrt{k}} \frac{\partial \sqrt{k}}{\partial k} \right) \mathcal{W}[f_i(x)](x, k) \\ &+ \sqrt{\alpha k} \int_{-\infty}^{+\infty} f(\eta) \frac{\partial}{\partial k} [\psi^*(\alpha k(\eta - x))] d\eta \\ &= \left(\frac{1}{2k} \right) \mathcal{W}[f_i(x)](x, k) + \sqrt{\alpha k} \int_{-\infty}^{+\infty} f(\eta) \alpha(\eta - x) \psi'^*(\alpha k(\eta - x)) d\eta \\ &= \left(\frac{1}{2k} \right) \mathcal{W}[f_i(x)](x, k) + \sqrt{\alpha k} \int_{-\infty}^{+\infty} f(\eta) \alpha(\eta - x) (i\beta + \alpha k(\eta - x)) \psi^* \\ &(\alpha k(\eta - x)) d\eta = \left(\frac{1}{2k} \right) \mathcal{W}[f_i(x)](x, k) \\ &+ \sqrt{\alpha k} \int_{-\infty}^{+\infty} \left[(i\beta - \alpha^2 kx^2) \right. \end{aligned}$$

$$\begin{aligned}
 &+ (-i\beta\alpha + 2\alpha^2 kx) \eta \\
 &- (\alpha^2 k) \eta^2 \int f(\eta) \psi^*(\alpha k(\eta - x)) d\eta \\
 &= \left(\frac{1}{2k} + i\beta\alpha x - \alpha^2 kx^2 \right) \mathcal{W}[f_i(x)](x, k) \\
 &+ (2\alpha^2 kx - i\beta\alpha) \mathcal{W}[x f_i(x)](x, k) \\
 &- (\alpha^2 k) \mathcal{W}[x^2 f_i(x)](x, k).
 \end{aligned}$$

and

$$\begin{aligned}
 \frac{\partial^2 \mathcal{W}[x^p f_i(x)](x, k)}{\partial k^2} &= \left(-\frac{1}{2k^2} - \alpha^2 x^2 \right) \\
 &\mathcal{W}[x^p f_i(x)](x, k) + (2\alpha^2 x) \\
 &\mathcal{W}[x^{p+1} f_i(x)](x, k) - (\alpha^2) \\
 &\mathcal{W}[x^{p+2} f_i(x)](x, k) \\
 &+ \left(\frac{1}{2k} + i\beta\alpha x - \alpha^2 kx^2 \right) \\
 &\frac{\partial \mathcal{W}[x^p f_i(x)](x, k)}{\partial k} \\
 &+ (2\alpha^2 kx - i\beta\alpha) \\
 &\frac{\partial \mathcal{W}[x^{p+1} f_i(x)](x, k)}{\partial k} \\
 &- (\alpha^2 k) \frac{\partial \mathcal{W}[x^{p+2} f_i(x)](x, k)}{\partial k}.
 \end{aligned}$$

We find a general formulation with $p = 0 \dots N - 2$:

$$\begin{aligned}
 \frac{\partial \mathcal{W}[f_i(x)](x, k)}{\partial k} &= \left(\frac{1}{2k} + i\beta\alpha x - 2\alpha^2 kx^2 \right) \\
 &\mathcal{W}[x^p f_i(x)](x, k) \\
 &+ (2\alpha^2 kx - i\beta\alpha) \mathcal{W}[x^{p+1} f_i(x)](x, k) \\
 &- (\alpha^2 k) \mathcal{W}[x^{p+2} f_i(x)](x, k). \tag{B8}
 \end{aligned}$$

The third one is the computation of $\frac{\partial^3 \phi_i(x, k)}{\partial k^2 \partial x}$:

$$\begin{aligned}
 \frac{\partial^3 \phi_i(x, k)}{\partial k^2 \partial x} &= 2\alpha^2 \text{Im} \left\{ \frac{\mathcal{W}[x f_i(x)](x, k)}{\mathcal{W}[f_i(x)](x, k)} \right\} \\
 &+ 4\alpha^2 k \text{Im} \left\{ \frac{\partial}{\partial k} \left(\frac{\mathcal{W}[x f_i(x)](x, k)}{\mathcal{W}[f_i(x)](x, k)} \right) \right\} \\
 &+ (\alpha k)^2 \text{Im} \left\{ \frac{\partial^2}{\partial k^2} \left(\frac{\mathcal{W}[x f_i(x)](x, k)}{\mathcal{W}[f_i(x)](x, k)} \right) \right\}, \tag{B9}
 \end{aligned}$$

with

$$\begin{aligned}
 \frac{\partial^2}{\partial k^2} \left(\frac{\mathcal{W}[x f_i(x)](x, k)}{\mathcal{W}[f_i(x)](x, k)} \right) &= \frac{1}{\mathcal{W}[f_i(x)](x, k)} \\
 &\frac{\partial^2 \mathcal{W}[x f_i(x)](x, k)}{\partial k^2} \\
 &- \frac{\mathcal{W}[x f_i(x)](x, k)}{\mathcal{W}[f_i(x)](x, k)^2} \\
 &\frac{\partial^2 \mathcal{W}[f_i(x)](x, k)}{\partial k^2} \\
 &- \frac{2}{\mathcal{W}[f_i(x)](x, k)^2} \\
 &\frac{\partial \mathcal{W}[f_i(x)](x, k)}{\partial k} \\
 &\frac{\partial \mathcal{W}[x f_i(x)](x, k)}{\partial k}
 \end{aligned}$$

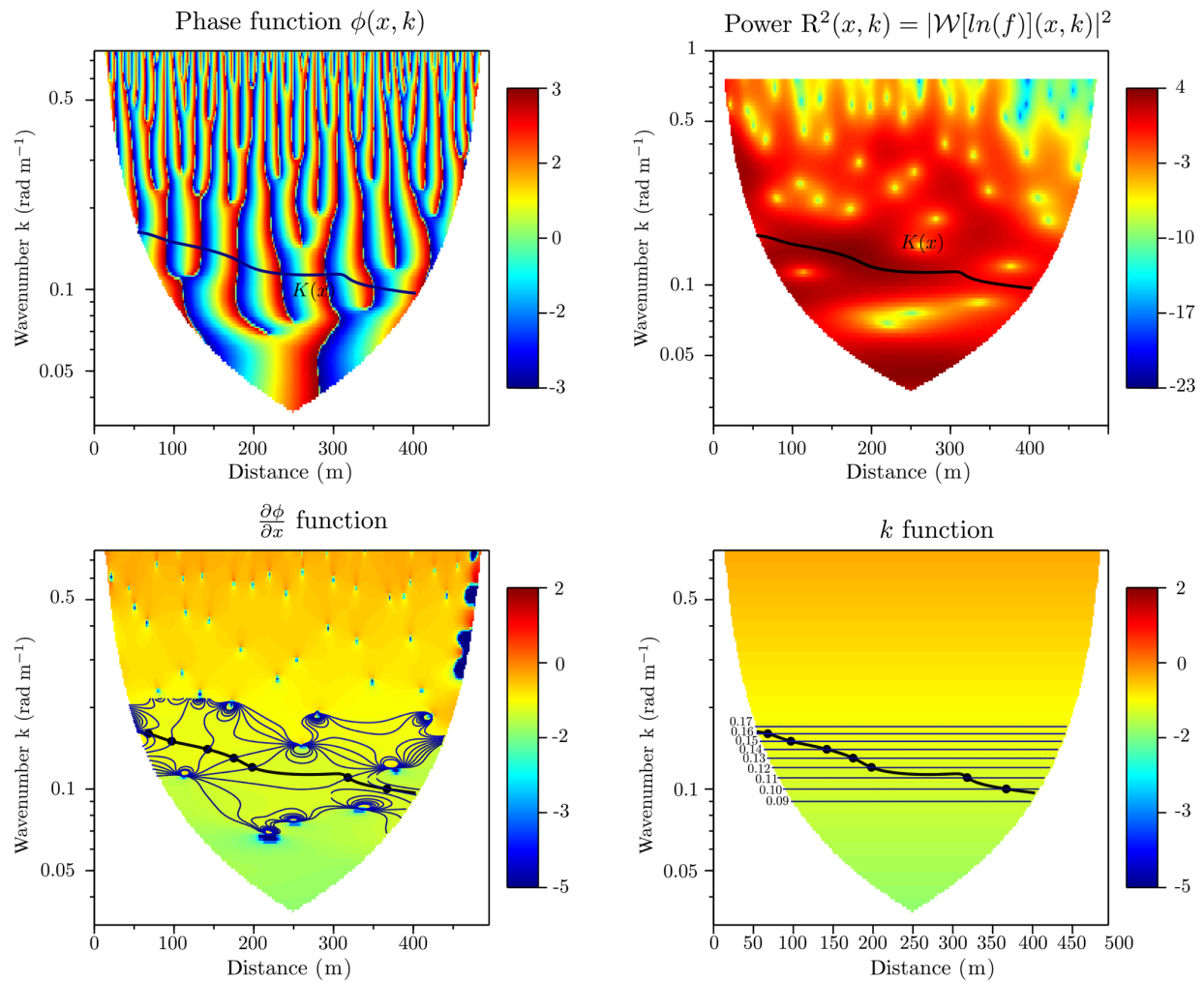


Figure B1. Steps of determining the local wavenumber $K(x)$ using the wavelet univariate ridge analysis of the velocity of the Olivet (3) reach, represented in the four panels. **(a)** The phase function $\phi(x, k)$; **(b)** the power's cone of influence of the wavelet to characterize the region where there is a maximum variability of the velocity in the Neperian logarithm; **(c)** the function $\frac{\partial \phi}{\partial x}$; **(d)** the function k .

Code availability. The script is available upon request from the authors.

Supplement. The supplement related to this article is available online at: <https://doi.org/10.5194/hess-24-3513-2020-supplement>.

Author contributions. NLM, MM, and RM designed the study. ON provided the dataset. MM and NLM built the methodology and analyzed the results. MM wrote the paper with contributions from NLM, PR, and RM.

Competing interests. The authors declare that they have no conflict of interest.

Acknowledgements. The authors acknowledge financial support from the French National Space Agency (Centre National d'Etudes Spatiales, CNES) and Sorbonne University through the PhD grant of Mounir Mahdade. We would also like to thank Gregory B. Pasternack, the second anonymous reviewer, and the editor for the valuable suggestions and comments that greatly helped improve this paper.

Financial support. This research has been supported by the Centre National d'Etudes Spatiales (CNES; grant no. CNES5100018525) and Sorbonne University (grant no. C18/1749).

Review statement. This paper was edited by Theresa Blume and reviewed by Gregory Pasternack and one anonymous referee.

References

- Alfieri, L., Feyen, L., Salamon, P., Thielen, J., Bianchi, A., Dottori, F., and Burek, P.: Modelling the socio-economic impact of river floods in Europe, *Nat. Hazards Earth Syst. Sci.*, 16, 1401–1411, <https://doi.org/10.5194/nhess-16-1401-2016>, 2016.
- Baume, J.-P. and Poirson, M.: Modélisation numérique d'un écoulement permanent dans un réseau hydraulique maillé à surface libre, en régime fluvial, *La Houille Blanche*, 1–2, 95–100, 1984.
- Box, G. E. and Jenkins, G. M.: *Time series analysis: Forecasting and control* Holden-Day, San Francisco, California, 1976.
- Brown, R. A. and Pasternack, G. B.: Bed and width oscillations form coherent patterns in a partially confined, regulated gravel-cobble-bedded river adjusting to anthropogenic disturbances, *Earth Surf. Dynam.*, 5, 1–20, <https://doi.org/10.5194/esurf-5-1-2017>, 2017.
- Brown, R. A., Pasternack, G. B., and Wallender, W. W.: Synthetic river valleys: Creating prescribed topography for form–process inquiry and river rehabilitation design, *Geomorphology*, 214, 40–55, 2014.
- Carling, P. A. and Orr, H. G.: Morphology of riffle-pool sequences in the River Severn, England, *Earth Surf. Proc. Land.*, 25, 369–384, 2000.
- Carmona, R. A., Hwang, W. L., and Torrèsani, B.: Multiridge detection and time–frequency reconstruction, *IEEE T. Sig. Process.*, 47, 480–492, 1999.
- Clifford, N. J.: Formation of riffle–pool sequences: field evidence for an autogenetic process, *Sediment. Geol.*, 85, 39–51, 1993.
- Dury, G.: Osage-type underfitness on the River Severn near Shrewsbury, Shropshire, England, *Background to palaeohydrology*, Wiley, Chichester, 399–412, 1983.
- Frothingham, K. M. and Brown, N.: Objective identification of pools and riffles in a human-modified stream system, *Mid. Stat. Geogr.*, 35, 52–60, 2002.
- Gabor, D.: Theory of communication. Part 1: The analysis of information, *J. Inst. Elect. Eng. Pt. III*, 93, 429–441, 1946.
- Gangodagamage, C., Barnes, E., and Fofoula-Georgiou, E.: Scaling in river corridor widths depicts organization in valley morphology, *Geomorphology*, 91, 198–215, 2007.
- Gregory, K., Gurnell, A., Hill, C., and Tooth, S.: Stability of the pool-riffle sequence in changing river channels, *River Res. Appl.*, 9, 35–43, 1994.
- Grinsted, A., Moore, J. C., and Jevrejeva, S.: Application of the cross wavelet transform and wavelet coherence to geophysical time series, *Nonlin. Processes Geophys.*, 11, 561–566, <https://doi.org/10.5194/npg-11-561-2004>, 2004.
- Harvey, A.: Some aspects of the relations between channel characteristics and riffle spacing in meandering streams, *Am. J. Sci.*, 275, 470–478, 1975.
- Hauer, C., Mandlbürger, G., and Habersack, H.: Hydraulically related hydro-morphological units: description based on a new conceptual mesohabitat evaluation model (MEM) using LiDAR data as geometric input, *River Res. Appl.*, 25, 29–47, 2009.
- Hauer, C., Unfer, G., Tritthart, M., Formann, E., and Habersack, H.: Variability of mesohabitat characteristics in riffle-pool reaches: Testing an integrative evaluation concept (FGC) for MEM-application, *River Res. Appl.*, 27, 403–430, 2011.
- Higuchi, H., Lewalle, J., and Crane, P.: On the structure of a two-dimensional wake behind a pair of flat plates, *Phys. Fluids*, 6, 297–305, 1994.
- Hogan, D. L.: *Channel morphology of unlogged, logged, and debris torrented streams in the Queen Charlotte Islands*, Ministry of Forests and Lands, British Columbia, Canada, 1986.
- Jowett, I. G.: A method for objectively identifying pool, run, and riffle habitats from physical measurements, *N. Zeal. J. Mar. Freshw. Res.*, 27, 241–248, 1993.
- Katul, G. G. and Parlange, M. B.: Analysis of land surface heat fluxes using the orthonormal wavelet approach, *Water Resour. Res.*, 31, 2743–2749, 1995a.
- Katul, G. G. and Parlange, M. B.: The spatial structure of turbulence at production wavenumbers using orthonormal wavelets, *Bound.-Lay. Meteorol.*, 75, 81–108, 1995b.
- Katul, G. G., Parlange, M. B., and Chu, C. R.: Intermittency, local isotropy, and non-Gaussian statistics in atmospheric surface layer turbulence, *Phys. Fluids*, 6, 2480–2492, 1994.
- Keller, E.: Development of alluvial stream channels: a five-stage model, *Geol. Soc. Am. Bull.*, 83, 1531–1536, 1972.
- Keller, E. and Melhorn, W.: *Bedforms and fluvial processes in alluvial stream channels: selected observations*, edited by: Morisawa,

- M., *Fluvial Geomorphology*, St. Univ. New York, Binghamton, USA, 253–283, 1973.
- Keller, E. and Melhorn, W.: Rhythmic spacing and origin of pools and riffles, *Geol. Soc. Am. Bull.*, 89, 723–730, 1978.
- Knighton, A.: Asymmetry of river channel cross-sections: Part I. Quantitative indices, *Earth Surf. Proc. Land.*, 6, 581–588, 1981.
- Knighton, D.: *Fluvial forms and processes: a new perspective*, Routledge, New York, USA, 2014.
- Kondolf, G. M.: Geomorphological stream channel classification in aquatic habitat restoration: uses and limitations, *Aquat. Conserv.: Mar. Freshw. Ecosyst.* 5, 127–141, 1995.
- Krueger, A. and Frothingham, K.: Application and comparison of geomorphological and hydrological pool and riffle quantification methods, *Geogr. Bull.*, 48, 85–95, 2007.
- Kumar, P.: Role of coherent structures in the stochastic-dynamic variability of precipitation, *J. Geophys. Res.-Atmos.*, 101, 26393–26404, 1996.
- Kumar, P. and Foufoula-Georgiou, E.: A multicomponent decomposition of spatial rainfall fields: 1. Segregation of large- and small-scale features using wavelet transforms, *Water Resour. Res.*, 29, 2515–2532, 1993.
- Lashermes, B. and Foufoula-Georgiou, E.: Area and width functions of river networks: New results on multifractal properties, *Water Resour. Res.*, 43, W09405, <https://doi.org/10.1029/2006WR005329>, 2007.
- Legleiter, C. J.: A geostatistical framework for quantifying the reach-scale spatial structure of river morphology: 1. Variogram models, related metrics, and relation to channel form, *Geomorphology*, 205, 65–84, 2014.
- Leopold, L., Wolman, M., and Miller, J.: *Fluvial processes in geomorphology*, Freeman, San Francisco, 1964.
- Lilly, J. M. and Olhede, S. C.: Higher-order properties of analytic wavelets, *IEEE T. Sig. Process.*, 57, 146–160, 2008.
- Lilly, J. M. and Olhede, S. C.: On the analytic wavelet transform, *IEEE T. Inform. Theory*, 56, 4135–4156, 2010.
- Lilly, J. M. and Olhede, S. C.: Analysis of modulated multivariate oscillations, *IEEE T. Sig. Process.*, 60, 600–612, 2011.
- Mallat, S.: *A wavelet tour of signal processing*, Elsevier, San Diego, CA, USA, 1999.
- McKean, J., Nagel, D., Tonina, D., Bailey, P., Wright, C. W., Bohn, C., and Nayegandhi, A.: Remote sensing of channels and riparian zones with a narrow-beam aquatic-terrestrial LIDAR, *Remote Sens.*, 1, 1065–1096, 2009.
- Milne, J.: Bed-material size and the riffle-pool sequence, *Sedimentology*, 29, 267–278, 1982.
- Milne, J. and Sear, D.: Modelling river channel topography using GIS, *Int. J. Geogr. Inform. Sci.*, 11, 499–519, 1997.
- Montgomery, D. R., Buffington, J. M., Smith, R. D., Schmidt, K. M., and Pess, G.: Pool spacing in forest channels, *Water Resour. Res.*, 31, 1097–1105, 1995.
- Muzy, J.-F., Bacry, E., and Arneodo, A.: Multifractal formalism for fractal signals: The structure-function approach versus the wavelet-transform modulus-maxima method, *Phys. Rev. E*, 47, 875–884, 1993.
- Navratil, O.: Débit de pleins bords et géométrie hydraulique: une description synthétique de la morphologie des cours d'eau pour relier le bassin versant et les habitats aquatiques, Institut National Polytechnique de Grenoble, Grenoble, France, 2005.
- Navratil, O., Albert, M., Herouin, E., and Gresillon, J.: Determination of bankfull discharge magnitude and frequency: comparison of methods on 16 gravel-bed river reaches, *Earth Surf. Proc. Land.*, 31, 1345–1363, 2006.
- Neal, J. C., Odoni, N. A., Trigg, M. A., Freer, J. E., Garcia-Pintado, J., Mason, D. C., Wood, M., and Bates, P. D.: Efficient incorporation of channel cross-section geometry uncertainty into regional and global scale flood inundation models, *J. Hydrol.*, 529, 169–183, 2015.
- Ng, E. K. and Chan, J. C.: Geophysical applications of partial wavelet coherence and multiple wavelet coherence, *J. Atmos. Ocean. Tech.*, 29, 1845–1853, 2012.
- Nordin, C. F.: *Statistical properties of dune profiles*, in: vol. 562, US Government Printing Office, Washington, USA, 1971.
- Nourani, V., Baghanam, A. H., Adamowski, J., and Kisi, O.: Applications of hybrid wavelet – Artificial Intelligence models in hydrology: A review, *J. Hydrol.*, 514, 358–377, 2014.
- O'Neill, M. P. and Abrahams, A. D.: Objective identification of pools and riffles, *Water Resour. Res.*, 20, 921–926, 1984.
- Ozkurt, N. and Savaci, F. A.: Determination of wavelet ridges of nonstationary signals by singular value decomposition, *IEEE T. Circuit. Syst. II*, 52, 480–485, 2005.
- Pasternack, G. and Arroyo, R.: RiverBuilder: River Generation for Given Data Sets, R package version 0.1, available at: <https://cran.r-project.org/web/packages/RiverBuilder/index.html>, (last access: 11 June 2020), 2018.
- Pasternack, G. and Zhang, M.: *River Builder User's Manual For Version 1.0.0*, University of California, Davis, CA, available at: <https://github.com/RiverBuilder/RiverBuilder>, last access: 11 June 2020.
- Pasternack, G. B. and Brown, R. A.: 20 Ecohydraulic Design of Riffle-Pool Relief and Morphological Unit Geometry in Support of Regulated Gravel-Bed River Rehabilitation, in: *Ecohydraulics*, Wiley Online Library, p. 337, 2013.
- Pasternack, G. B. and Hinnov, L. A.: Hydrometeorological controls on water level in a vegetated Chesapeake Bay tidal freshwater delta, *Estuar. Coast. Shelf Sci.*, 58, 367–387, 2003.
- Richards, K.: The morphology of riffle-pool sequences, *Earth Surf. Proc. Land.*, 1, 71–88, 1976a.
- Richards, K.: Channel width and the riffle-pool sequence, *Geol. Soc. Am. Bull.*, 87, 883–890, 1976b.
- Rodríguez, J. F., García, C. M., and García, M. H.: Three-dimensional flow in centered pool-riffle sequences, *Water Resour. Res.*, 49, 202–215, 2013.
- Rosgen, D. L.: The cross-vane, w-weir and j-hook vane structures . . . their description, design and application for stream stabilization and river restoration, in: *Designing successful stream and wetland restoration projects*, edited by: Hays, D. F., Proceedings of the 2001 Wetlands Engineering and River Restoration, 27–31 August 2001, Reno, NV, USA, Am. Soc. Civil Eng., Reston, VA, USA, 1–22, [https://doi.org/10.1061/40581\(2001\)72](https://doi.org/10.1061/40581(2001)72), 2001.
- Rossi, A., Massei, N., and Laignel, B.: A synthesis of the time-scale variability of commonly used climate indices using continuous wavelet transform, *Global Planet. Change*, 78, 1–13, 2011.
- Saleh, F., Ducharne, A., Flipo, N., Oudin, L., and Ledoux, E.: Impact of river bed morphology on discharge and water levels simulated by a 1D Saint-Venant hydraulic model at regional scale, *J. Hydrol.*, 476, 169–177, 2013.

- Schaeffli, B., Maraun, D., and Holschneider, M.: What drives high flow events in the Swiss Alps? Recent developments in wavelet spectral analysis and their application to hydrology, *Adv. Water Resour.*, 30, 2511–2525, 2007.
- Schneider, K. and Vasilyev, O. V.: Wavelet methods in computational fluid dynamics, *Annu. Rev. Fluid Mech.*, 42, 473–503, 2010.
- Schweizer, S., Borsuk, M. E., Jowett, I., and Reichert, P.: Predicting joint frequency distributions of depth and velocity for instream habitat assessment, *River Res. Appl.*, 23, 287–302, 2007.
- Thompson, D. M.: Random controls on semi-rhythmic spacing of pools and riffles in constriction-dominated rivers, *Earth Surf. Proc. Land.*, 26, 1195–1212, 2001.
- Thomson, D. J.: Spectrum estimation and harmonic analysis, *Proc. IEEE*, 70, 1055–1096, 1982.
- Torrence, C. and Compo, G. P.: A practical guide to wavelet analysis, *B. Am. Meteorol. Soc.*, 79, 61–78, 1998.
- Trigg, M. A., Wilson, M. D., Bates, P. D., Horritt, M. S., Alsdorf, D. E., Forsberg, B. R., and Vega, M. C.: Amazon flood wave hydraulics, *J. Hydrol.*, 374, 92–105, 2009.
- Wadeson, R.: A geomorphological approach to the identification and classification of instream flow environments, *S. Afr. J. Aquat. Sci.*, 20, 38–61, 1994.
- Wheaton, J. M., Pasternack, G. B., and Merz, J. E.: Spawning habitat rehabilitation – I. Conceptual approach and methods, *Int. J. River Basin Manage.*, 2, 3–20, 2004.
- Wyrick, J. R. and Pasternack, G.: Geospatial organization of fluvial landforms in a gravel–cobble river: beyond the riffle–pool couplet, *Geomorphology*, 213, 48–65, 2014.
- Wyrick, J. R., Senter, A. E., and Pasternack, G. B.: Revealing the natural complexity of fluvial morphology through 2D hydrodynamic delineation of river landforms, *Geomorphology*, 210, 14–22, 2014.
- Yang, C. T.: Formation of Riffles and Pools, *Water Resour. Res.*, 7, 1567–1574, <https://doi.org/10.1029/WR007i006p01567>, 1971.

2.3 Conclusion du chapitre

Dans cette étude, nous présentons quelques caractéristiques des tronçons alluviaux avec des alternances des unités morphologiques de type mouille-radier. Nous avons proposé un outil d'extraction de ces caractéristiques en utilisant une approche multivariée des ondelettes.

La longueur d'onde des unités morphologiques extraite est une longueur d'onde qui caractérise le tronçon. Ceci est plus précis qu'une extraction arbitraire des espacements moyens entre les mouilles, les radiers ou la moyenne des deux. Ces longueurs d'onde ainsi que leurs amplitudes permettent la caractérisation de la morphologie des tronçons. Ces paramètres varient en fonction de la position de la mouille ou du radier et aussi en fonction de la forme en plan de la ligne de thalweg. De plus, ils coévoluent avec les variables d'écoulement (par exemple, vitesse), de la géométrie longitudinale (par exemple, rayon hydraulique) et transversale (par exemple, angle avec la direction du thalweg) et des frottements (par exemple, contrainte de cisaillement). Par conséquent, une variabilité au niveau de la structure morphologique peut entraîner une variabilité dans la courbe de tarage du tronçon.

La longueur d'onde moyenne au tronçon ainsi que l'amplitude peuvent représenter la variabilité de la bathymétrie dans les modèles hydrauliques dans le cas où l'accès complet à la géométrie du lit mineur est impossible (le problème à résoudre dans cette thèse). Dans ce cas, la morphologie du cours d'eau peut être modélisée par des fonctions pseudo-périodiques.

Cette étude peut être appliquée sur n'importe quel type d'alternances de faciès. Les données utilisées concernent des tronçons avec des largeurs inférieures à 50 m. Ces tronçons ne rentrent pas dans le gabarit des rivières observées par SWOT. En revanche, cette caractérisation reste valable pour les rivières avec des largeurs supérieures à 50 m. Ainsi, nous avons choisi un tronçon de la Garonne. Elle compte parmi les rivières visées par la mission SWOT afin d'élaborer des algorithmes d'estimation du débit (*Durand et al., 2016; Garambois et Monnier, 2015; Yoon et al., 2016; Oubanas et al., 2018a*).

Avant de paramétrer le modèle bathymétrique 2D (cette partie est détaillée dans le Chapitre 5). Il faut préparer le jeu de données pour tester et valider l'approche.

Dans le chapitre suivant, nous allons présenter une simulation de référence de crue sur la Garonne. À travers cette simulation, nous allons présenter la méthodologie d'élaboration de la topographie du tronçon ainsi que le calage des coefficients de frottement.

Chapitre 3

Production d’une simulation de référence sur la Garonne

3.1 Introduction

La simplification de la géométrie/bathymétrie est une solution parmi d’autres qui permet la réduction des degrés de liberté du problème inverse (bathymétrie du lit mineur, frottements et courbe de tarage inconnus). Dans cette étude, nous proposons une simplification non-uniforme périodique de la géométrie dans un plan bidimensionnel. Ce modèle prend en compte les interactions entre le lit mineur et la plaine inondable et produit des courbes de tarage non limitées au débit de plein bord. De plus, cette approche permet d’utiliser des données télédéteectées de la plaine inondable dans le but de contraindre le problème inverse.

Cette méthodologie nécessite de choisir une rivière avec des données *in situ* disponibles et des données similaires aux données SWOT afin de tester le modèle et de valider les résultats. Nous avons donc choisi un tronçon de la Garonne pour lequel ses données *in situ* sont disponibles. En effet, cette rivière fait partie de l’ensemble des tronçons sur lesquels la communauté SWOT teste les approches d’estimation des débits (Durand *et al.*, 2016; Garambois et Monnier, 2015; Yoon *et al.*, 2016; Oubanas *et al.*, 2018a).

Le tronçon étudié fait 40 km de longueur avec une largeur moyenne de 150m et une profondeur maximale de 5-6 m aux faibles débits. Le tronçon se trouve entre le pont des Catalans et le pont de Verdun-sur-Garonne. Le détail du tronçon et des données disponibles sont présentés dans la partie 3.2. Comme introduit dans le Chapitre 1, nous avons choisi de faire une simulation de référence à haute résolution d’un événement de crue important. Ensuite, nous générerons des pseudo-observations en échantillonnant cette simulation de référence.

Etant donné que l’inversion se fera en 2D, nous avons choisi de simuler une crue de référence qui demande :

- une bathymétrie continue du lit mineur et de la plaine inondable ;
- un calage des coefficients de frottement en lit mineur et lit majeur ;
- des données liées au calage et à la simulation de la crue de 2000 (hydrogramme de crue à la station de jaugeage aval, hydrogrammes d’entrée, etc.).

Comment peut-on donc élaborer une bathymétrie du lit mineur et lit majeur avec une haute résolution d’un tronçon ? quel type de modèle hydraulique et de schéma associé faut-il choisir ? et comment caler les coefficients de frottement dans ce cas ?

Le but de la partie 3.3 est de construire une bathymétrie 2D du tronçon de la Garonne. Cette topographie continue est utilisée, d’un côté, comme une bathymétrie “observée” afin de la comparer avec la bathymétrie périodique approchée, et d’un autre côté, comme un élément très important dans la simulation de référence.

La partie 3.4 présente la simulation de la crue de 2000 en calibrant les coefficients de frottement du lit mineur et du lit majeur. Ces coefficients sont utilisés pour valider les frottements inversés par l’approche périodique. Cette simulation joue un rôle très important dans le test de l’algorithme d’inversion. D’une part, nous considérons que les résultats de cette simulation sont des pseudo-observables télédéteçtés, auxquelles il faudrait appliquer un bruit ayant la même structure que le bruit de mesure de l’instrument SWOT (cette phase de bruitage n’a pas pu être réalisée faute de temps). Ils permettent le calcul de signatures “moyennées” au tronçon, qui seront définies au Chapitre 4, et qui seront comparées ensuite avec les signatures estimées par l’approche périodique selon une méthodologie définie au Chapitre 5.

3.2 La Garonne comme tronçon d’étude

Cette étude est réalisée sur le bassin de la Garonne situé dans le sud-ouest de la France, étant l’un des principaux systèmes fluviaux du pays. Le bassin versant de la Garonne (Figure 3.1) a une superficie de 55000 km². Il traverse trois grandes unités morphostructurales du sud-ouest de la France. À l’amont, ils existent deux ensembles montagnards distincts : les Pyrénées centrales (franco-espagnoles) et le sud-ouest du Massif Central. Au centre et à l’aval, il y a l’un des principaux bassins sédimentaires français : le Bassin Aquitain. La Garonne s’écoule des Pyrénées jusqu’à l’Océan Atlantique sur une longueur totale de 647 km. Les affluents de rive gauche issus des Pyrénées sont essentiellement des petits cours d’eau organisés sur les cônes (Baïze, Gers, Gimone, Save, etc.). Pourtant, les affluents de rive droite sont majoritairement issus du Massif Central et représentent souvent les apports les plus importants, qui modifient le régime de la Garonne (par exemple, Tarn et Lot) (David, 2016).

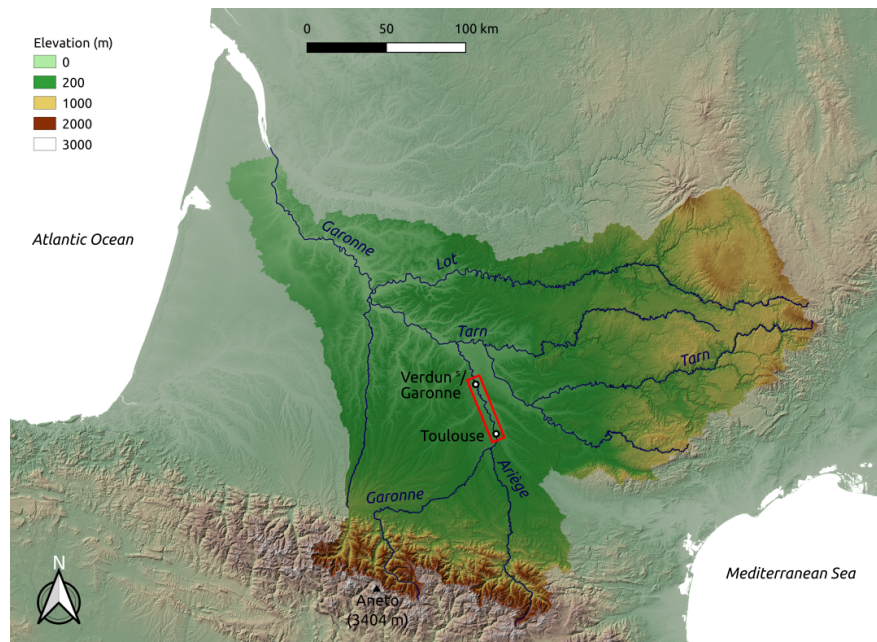


Figure 3.1: Bassin versant de la Garonne. Le domaine d’étude est la partie encadrée en rouge.

On peut diviser la zone en Garonne montagnarde, moyenne et maritime. La Garonne montagnarde correspond à la partie amont du bassin versant jusqu’à la confluence de l’Ariège à Toulouse-Portet, alimentée par des affluents pyrénéens. Elle est marquée par une pente forte et un chenal confiné instable. La Garonne moyenne (Toulousaine), entre la confluence de l’Ariège (Toulouse) et la confluence du Dropt à Castets-en-Dorthe, se caractérise par une vallée plus large et un chenal instable moins confiné. Ainsi, la Garonne

maritime, à l'aval de Castets-en-Dorthe, correspond à la partie du fleuve sous influence de la dynamique de la marée, qui forme l'estuaire de la Gironde (75 km de longueur), après sa confluence avec la Dordogne (David, 2016).

Le tronçon d'étude de cette thèse se situe dans la Garonne moyenne, dite Garonne toulousaine qui correspond à un linéaire de 40 km de longueur, compris entre le pont des Catalans à Toulouse et le pont de Verdun-sur-Garonne. Le choix du pont des Catalans au lieu de la confluence de l'Ariège à Toulouse-Portet est fait pour éviter tous les ouvrages hydrauliques entre la station Portet-sur-Garonne et le pont des Catalans. Ce chenal présente un tracé sinueux à méandrique. La pente est douce et varie entre 0.0009 m/m et 0.001 m/m. La largeur moyenne est ~ 150 m et la profondeur moyenne est ~ 4 m (Steiger et Corenblit, 2000). La morphologie de la Garonne Toulousaine se caractérise principalement par des affleurements molassique miocène et s'apparente à un *bedrock* et par une dominance des graviers et des galets (Sawage et al., 2003; Jantzi et al., 2020) avec une structure mouille-radier présente dans le chenal. Contrairement au centre de Toulouse où les berges sont stabilisées par des digues de terre revêtues de dalles de béton, le tronçon d'étude se caractérise par des berges relativement naturelles puisqu'elles font l'objet d'aménagements ponctuels. Il existe de nombreuses singularités (ponts, déversoirs, etc.) et aussi de grands seuils. Ces singularités influencent localement le régime hydraulique du fleuve.

Concernant l'hydrologie du bassin versant, le débit du fleuve est très variable. En effet, le débit le plus élevé, pendant la crue de 1875, a été estimé à $7000 \text{ m}^3/\text{s}$, et le débit en période d'étiage est de $40 \text{ m}^3/\text{s}$. Le bassin a un régime pluvio-nival avec un débit moyen annuel supérieur à $200 \text{ m}^3/\text{s}$. Il existe trois stations de mesure associées à ce tronçon. La station de Toulouse-Portet à Portet-sur-Garonne (bassin versant de 9980 km^2), la station de Toulouse-Pont-neuf (bassin versant légèrement supérieur) et la station de Verdun-sur-Garonne (bassin versant de 13700 km^2). Dans la station Toulouse-Portet, les débits sont enregistrés depuis 1910 et les hauteurs d'eau depuis 1980. La station Verdun-sur-Garonne contient des données de débit depuis 1972 et de hauteur d'eau depuis 1980. La station Toulouse-Pont-neuf enregistre les hauteurs d'eau depuis 1855.

Les données topographiques *in situ* disponibles du tronçon étudié sont des profils en long et en travers de 120 sections collectées entre 1987 et 2002. Ces données sont utilisées dans la construction de la bathymétrie continue 2D du tronçon. La partie 3.3 présente la création de la bathymétrie du lit mineur du tronçon en combinant des images aériennes avec les données *in situ*.

Concernant les données hydrologiques, les hydrogrammes des deux crues de 2000 et 2002 sont disponibles à l'aval à Verdun-sur-Garonne. Les hydrogrammes à injecter en amont ne sont pas disponibles au niveau tous les affluents et donc une modélisation hydrologique à ce niveau est nécessaire. La cartographie des crues et le calage des coefficients de frottement sont faits à l'aide du modèle hydraulique Basilisk. Ce modèle avec son avantage de raffinement adaptatif des mailles permet de réduire le temps de calcul dans les simulations. Le calage du coefficient de frottement en lit mineur est effectué à l'aide d'une courbe de tarage disponible à Verdun-sur-Garonne à l'échelle des hautes-eaux (HE) ainsi que la dynamique de l'événement non-débordeant de juin 2002. Nous proposons également une méthodologie de calage du coefficient de frottement en lit majeur adaptée au modèle hydraulique Basilisk. Ce calage est fait à l'aide des données des repères et des laisses de la crue majeure de juin 2000 ainsi que sa dynamique de propagation entre Toulouse et Verdun-sur-Garonne.

La présentation de la méthodologie de production de cette simulation de référence se fait donc en deux temps dans le mémoire, sous la forme de deux articles en anglais :

- dans la section 3.3, nous présentons la méthode de construction de la bathymétrie sur la base de profils en long levés *in situ* et d'images aériennes dans le domaine visible et proche infrarouge,
- dans la section 3.4, nous présentons la méthodologie de calage du modèle hydraulique 2D Basilisk. Cette étude ayant vocation à être présentée de façon autonome et décorrélée des aspects inversion, elle est principalement axée sur les développements algorithmiques dans Basilisk (injections de débits et parallélisation) et la Garonne est présentée comme un cas test. La présentation est donc inverse

à la démarche de la thèse, puisque c'est la production de la simulation de référence qui a motivé les développements dans Basilisk.

3.3 Construction de la bathymétrie

Cette section est sous la forme d'un article publié dans *Remote Sensing* :

Résumé : La bathymétrie est un élément clé de la modélisation des systèmes fluviaux pour la cartographie des inondations, la géomorphologie ou la caractérisation des habitats fluviaux. Les pratiques standards reposent sur l'interpolation de mesures de profondeurs *in situ* obtenues à l'aide de GPS différentiels ou de relevés de stations totales, tandis que des techniques plus avancées font appel au LiDAR bathymétrique ou à des sondages acoustiques. Cependant, ces techniques actives à haute résolution ne sont pas si faciles à appliquer sur de grandes surfaces. Les méthodes alternatives utilisant l'imagerie optique passive présentent un compromis intéressant : elles reposent sur le fait que les longueurs d'onde composant le rayonnement solaire ne sont pas atténuées au même rythme dans l'eau. Sous certaines hypothèses, le logarithme du rapport des radiances dans deux bandes spectrales est corrélé linéairement avec la profondeur. Dans cette étude, nous allons au-delà de ces méthodes de rapport en définissant une teinte multi-spectrale qui conserve toutes les informations spectrales. Etant données n bandes coregistrées, cet invariant spectral se situe sur la $(n - 2)$ -sphère incluse dans \mathbb{R}^{n-1} , désignée \mathbb{S}^{n-2} et nommée "hypersphère des teintes". Il peut être considéré comme une généralisation de la "roue des couleurs" RVB (\mathbb{S}^1) dans des dimensions supérieures. Nous utilisons cette cartographie pour identifier une relation teinte-profondeur dans un tronçon de 35 km de la Garonne, en utilisant des images aériennes à haute résolution (0.50 m) dans quatre bandes et des données provenant de 120 sections transversales étudiées. La distribution de la teinte multi-spectrale sur les pixels du fleuve est modélisée comme un mélange de deux composantes : une composante représente la distribution de la teinte du substrat, tandis que l'autre représente la distribution de la teinte de l'eau profonde ; les paramètres sont ajustés de telle sorte que la probabilité d'appartenance à la composante "profonde" est corrélée avec la profondeur.

Le Moine, N.; Mahdade, M. A Preliminary Assessment of a Newly-Defined Multispectral Hue Space for Retrieving River Depth with Optical Imagery and In Situ Calibration Data. *Remote Sens.* 2021, 13, 4435. <https://doi.org/10.3390/rs13214435>

Article

A Preliminary Assessment of a Newly-Defined Multispectral Hue Space for Retrieving River Depth with Optical Imagery and In Situ Calibration Data

Nicolas Le Moine *  and Mounir Mahdade

Sorbonne Université, CNRS, EPHE, UMR 7619 METIS, 4 Place Jussieu, Box 105, 75005 Paris, France; mounir.mahdade@sorbonne-universite.fr

* Correspondence: nicolas.le_moine@sorbonne-universite.fr; Tel.: +33-144-276-326

Abstract: Bathymetry is a key element in the modeling of river systems for flood mapping, geomorphology, or stream habitat characterization. Standard practices rely on the interpolation of in situ depth measurements obtained with differential GPS or total station surveys, while more advanced techniques involve bathymetric LiDAR or acoustic soundings. However, these high-resolution active techniques are not so easily applied over large areas. Alternative methods using passive optical imagery present an interesting trade-off: they rely on the fact that wavelengths composing solar radiation are not attenuated at the same rates in water. Under certain assumptions, the logarithm of the ratio of radiances in two spectral bands is linearly correlated with depth. In this study, we go beyond these ratio methods in defining a multispectral hue that retains all spectral information. Given n coregistered bands, this spectral invariant lies on the $(n - 2)$ -sphere embedded in \mathbb{R}^{n-1} , denoted \mathbb{S}^{n-2} and tagged ‘hue hypersphere’. It can be seen as a generalization of the RGB ‘color wheel’ (\mathbb{S}^1) in higher dimensions. We use this mapping to identify a hue-depth relation in a 35 km reach of the Garonne River, using high resolution (0.50 m) airborne imagery in four bands and data from 120 surveyed cross-sections. The distribution of multispectral hue over river pixels is modeled as a mixture of two components: one component represents the distribution of substrate hue, while the other represents the distribution of ‘deep water’ hue; parameters are fitted such that membership probability for the ‘deep’ component correlates with depth.

Keywords: bathymetry; depth; rivers; multispectral hue; spectral invariant; directional statistics; Fisher–Bingham–Kent distribution; expectation–maximization algorithm



Citation: Le Moine, N.; Mahdade, M. A Preliminary Assessment of a Newly-Defined Multispectral Hue Space for Retrieving River Depth with Optical Imagery and In Situ Calibration Data. *Remote Sens.* **2021**, *13*, 4435. <https://doi.org/10.3390/rs13214435>

Academic Editor: Andrzej Stateczny

Received: 21 September 2021

Accepted: 29 October 2021

Published: 4 November 2021

Publisher’s Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Characterizing river channel bathymetry by average parameters (e.g., average depth, cross-section area, hydraulic radius) or sparsely distributed cross-sections may be sufficient for certain applications; however, a precise description is needed for applications in geomorphology or ecology, where local flow parameters are known to vary at the scale of a few times the full bank width in the streamwise direction with alternating morphological units such as pools, runs, and riffles [1,2].

Dense and precise bathymetric survey requires active measurement techniques such as bathymetric LiDAR using water-penetrating wavelengths in the visible spectrum [3,4], or acoustic soundings [5]. Unfortunately, these techniques rely on an artificial illuminating source (either electromagnetic or acoustic) and thus have a high operating cost and are very difficult to use over large areas.

A very interesting alternative is offered by passive optical methods, which have been used in marine, coastal, and more recently river settings since the pioneering work of Lyzenga [6]. Here, natural sunlight is the illuminating source and depth is retrieved by analyzing the radiance signal received at the sensor. According to Marcus and Fonstad [7],

this simple method may actually be ‘the only viable method for measuring, monitoring and mapping a large suite of in-channel river parameters continuously at sub-meter resolution’.

Following this introduction, Section 2 of this paper will provide a quick overview of the principles behind remotely-sensed estimation of river bathymetry using spectral ratios between bands in multispectral images. Then, in Section 3, we will temporarily move away from this specific application and propose a formal definition of a more general multispectral hue. Our procedure for estimating depth using this multispectral hue will be the core of Section 4, and results will be presented in Section 5. Then, we will summarize our findings and propose perspectives in Section 6.

2. Principles of Remotely-Sensed Estimation of River Bathymetry Using Passive Optical Measurements

As illustrated in Figure 1, the total upwelling spectral radiance (in $W \cdot m^{-2} \cdot sr^{-1}$) over water in a given spectral band with central wavelength λ can be split into four components (e.g., Legleiter et al. [8]):

$$L_{\uparrow}(\lambda) = L_b(\lambda) + L_c(\lambda) + L_s(\lambda) + L_p(\lambda) \quad (1)$$

- L_b is the radiance originating from the Lambertian (diffuse) reflection from the bed substrate;
- L_c is the *volume* radiance of the water column of depth h , which is basically sunlight that has been backscattered upwards before reaching the bottom;
- L_s is the surface radiance due to specular reflections at the air–water interface. L_s can make up a large fraction of L_{\uparrow} for certain geometries or viewing angles (‘sun glints’);
- L_p is a path radiance due to atmospheric scattering.

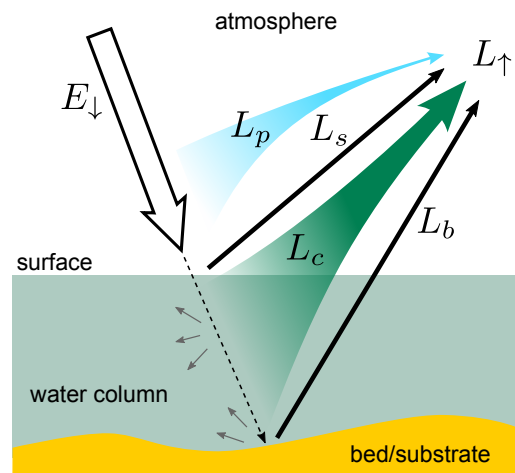


Figure 1. Decomposition of total measured radiance over water (adapted from Campbell, 1996 [9]). E_{\downarrow} is the downwelling solar irradiance ($W \cdot m^{-2}$), L_b , L_c , L_s and L_p are the bed, water column, surface and path radiances, respectively ($W \cdot m^{-2} \cdot sr^{-1}$).

Providing that we can account for both atmospheric path radiance and surface radiance, Philpot [10] derived the following expression for the total radiance measured at a given pixel of the sensor:

$$\begin{aligned} L_{\uparrow}(\mathbf{x}, \lambda) &\approx L_b(\mathbf{x}, \lambda) + L_c(\mathbf{x}, \lambda) \\ &\approx \underbrace{\left[(R_b(\mathbf{x}, \lambda) - R_{c,\infty}(\lambda)) e^{-K(\lambda)h(\mathbf{x})} + R_{c,\infty}(\lambda) \right]}_{R_{\text{mix}}(\mathbf{x}, h, \lambda) \text{ 'mixture' reflectance}} C_0 T_a(\lambda) E_{\downarrow}(\lambda) \end{aligned} \quad (2)$$

The rightmost term $E_{\downarrow}(\lambda)$ is the downwelling solar irradiance in the spectral band (in $W \cdot m^{-2}$), while the product $C_0 T_a$ (in sr^{-1}) accounts for both atmospheric and air–water interface transmission (C_0 can be considered roughly independent of wavelength). These terms are then multiplied by a reflectance term (brackets) which, according to Philpot, can be written as a mixture between two ‘end-members’:

- The first end-member is the reflectance of bed/substrate, R_b , which depends on both location \mathbf{x} and wavelength λ ,
- The second end-member is the reflectance $R_{c,\infty}$ that would be measured over an ‘infinitely deep’ water column.

The weight of the bed reflectance R_b in the mixture reflectance R_{mix} is an exponentially decreasing function of depth, with an attenuation coefficient $K(\lambda)$, which depends on the wavelength (Beer–Lambert law).

While the solar irradiance $E_{\downarrow}(\lambda)$ depends on the time of the day and on cloudiness, taking the ratio of solar irradiances in two different spectral bands cancels these variations. Hence, it is convenient to take the ratio:

$$\frac{L_{\uparrow}(\mathbf{x}, \lambda_i)}{L_{\uparrow}(\mathbf{x}, \lambda_j)} = \frac{\left[(R_b(\mathbf{x}, \lambda_i) - R_{c,\infty}(\lambda_i)) e^{-K(\lambda_i)h(\mathbf{x})} + R_{c,\infty}(\lambda_i) \right] C_0 T_a(\lambda_i) E_{\downarrow}(\lambda_i)}{\left[(R_b(\mathbf{x}, \lambda_j) - R_{c,\infty}(\lambda_j)) e^{-K(\lambda_j)h(\mathbf{x})} + R_{c,\infty}(\lambda_j) \right] C_0 T_a(\lambda_j) E_{\downarrow}(\lambda_j)} \quad (3)$$

Two common assumptions are then made to proceed with this expression:

1. First, the reflectance $R_{c,\infty}$ of deep water is assumed to be small compared to the bed reflectance R_b (for any substrate and any wavelength). Deep water reflectance is certainly low if the water is clear, but this assumption breaks down in the presence of sediment, dissolved organic matter, algae, etc. Furthermore, the bed might have a very low reflectance. Quartz sand might be highly reflective, but mud and rocks (especially rocks coated with a microbial film) can be quite dark;
2. Second, the ratio of bed reflectances at wavelengths λ_i and λ_j , $\frac{R_b(\mathbf{x}, \lambda_i)}{R_b(\mathbf{x}, \lambda_j)}$ is approximated as uniform in space.

With these assumptions, Equation (3) can be simplified and boils down to a linear relationship between depth and the logarithm of the spectral ratio:

$$X_{ij} = \ln \frac{L_{\uparrow}(\mathbf{x}, \lambda_i)}{L_{\uparrow}(\mathbf{x}, \lambda_j)} \approx \underbrace{(K(\lambda_j) - K(\lambda_i)) h}_{\text{slope}(\lambda_i, \lambda_j)} + \underbrace{\ln \left(\frac{R_b(\lambda_i)}{R_b(\lambda_j)} \right) + \ln \frac{C_0 T_a(\lambda_i) E_{\downarrow}(\lambda_i)}{C_0 T_a(\lambda_j) E_{\downarrow}(\lambda_j)}}_{\text{intercept}(\lambda_i, \lambda_j)} \quad (4)$$

Slope and intercept depend on optical properties of the atmosphere, water, and substrate at each wavelength but, if they can be considered invariant across the image, Equation (4) provides a predictor for depth. It has been used in numerous studies (see Shah et al. [11] for a recent review), either with the calibration of slope and intercept against in situ depth measurement, or without calibration; in the latter case, attenuation coefficients $K(\lambda)$ have to be modeled physically [12,13].

For a multispectral image with n coregistered bands, $n(n - 1)$ spectral ratios can be computed, but only $(n - 1)$ are independent (keeping those between *adjacent* bands for instance) since

$$X_{ik} = \ln \frac{L_{\uparrow}(\lambda_i)}{L_{\uparrow}(\lambda_k)} = \ln \left(\frac{L_{\uparrow}(\lambda_i)}{L_{\uparrow}(\lambda_j)} \frac{L_{\uparrow}(\lambda_j)}{L_{\uparrow}(\lambda_k)} \right) = X_{ij} + X_{jk} \quad (5)$$

Hence, a ratio–depth relation can be obtained by identifying a single, optimal pair of bands [8], or using several log-transformed ratios given by several pairs [14]. The use of multiple ratios (or other predictors) will increase model flexibility, though attention must be paid to multicollinearity [15]. In fact, retaining all spectral information from the $(n - 1)$ independent ratios amounts to defining a multispectral ‘hue’, or ‘pure color’, independent

of illumination conditions. The definition of such a spectral invariant is presented in Section 3.

3. Beyond Spectral Ratios: A Definition of Multispectral Hue

3.1. Rationale for the Definition of Multispectral Hue

3.1.1. RGB Case: The Color Wheel

Let us start with an example that will be familiar to many: many computer programs offer a color definition tool that relies on the kind of ‘color wheel’ shown in Figure 2. In this tool, a hue or ‘pure color’ can be selected along a circle, with an angle ranging between 0 and 360°. Two other operations can then be performed to obtain the final red, green and blue components:

1. The pure color can be partially *desaturated*, i.e., mixed with a white component. Such a mixing can occur for example with specular reflections, which will appear nearly white under white illumination if the refractive index of the material is slowly varying with wavelength (this is the case for water, or for an object coated with varnish).
2. The overall *brightness* or *value* of the color can be decreased. Physically speaking, this other operation is similar to either reducing the intensity of the light source, or changing the orientation of an object’s surface normals so that less light is diffused towards the sensor (‘shaded face’ effect).

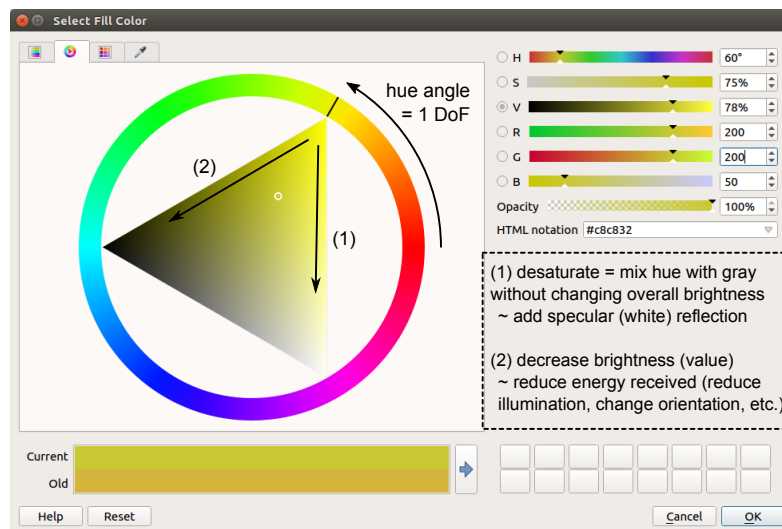


Figure 2. Screenshot of the color wheel tool in QGIS. The hue or ‘pure color’ is selected along the circle with a single angular degree of freedom (DoF). The two other operations that can be performed are (1) desaturation or (2) decrease in brightness/value.

3.1.2. Notations

In this section and the following ones, we will use bold symbols to denote points, vectors or matrices in a multidimensional space. Denoting \mathbf{v}^T the transpose of \mathbf{v} , $[1, 1, 1]$ denotes a row vector while $[1, 1, 1]^T$ denotes a column vector.

3.1.3. The Reason Why Multispectral Hue Should Have $(n - 2)$ Degrees of Freedom with n Bands

Defining the multispectral hue of a pixel rigorously amounts to reversing the two operations described above, that is, *resaturating* the color and *enhancing/normalizing* its brightness. Because of these two steps, multispectral hue should have only $(n - 2)$ degrees of freedom. This property is obviously verified with $n = 3$ bands and the resulting color wheel, which provides only one angular degree of freedom ($n - 2 = 1$).

We start our general definition of multispectral hue with the rationale of Montoliu et al. [16]. Given a pixel whose values in the n available bands form a vector $\mathbf{C} = [C_1, C_2, \dots, C_n]^T$ of digital counts, they first define a vector \mathbf{c} with entries

$$c_i = C_i - \min\{C_1, C_2, \dots, C_n\} = C_i - \min_k\{C_k\} \quad (6)$$

This operation removes the same value from all bands: in fact, it amounts to removing the *largest possible white component* at each pixel while preserving the positivity of all digital counts. In an image taken under natural sunlight, it will help to remove highlights or ‘sun glints’, even though they are not strictly white due to the wavelength dependency of refractive indices and variations in solar radiation [17]. The second step is actually a brightness normalization: the vector \mathbf{c} is divided by the sum of its entries, yielding a vector invariant $\mathbf{L}^{(M)}$ with entries:

$$L_i^{(M)} = \frac{c_i}{\sum_{j=1}^n c_j} = \frac{C_i - \min_k\{C_k\}}{\sum_{j=1}^n (C_j - \min_k\{C_k\})} \quad (7)$$

In order to illustrate the effect of these operations, we apply them to the RGB image displayed in Figure 3: it shows a set of plain color wooden toys with different shapes, under natural sunlight. The shapes are fitted so as to cast shadows on each other, each hue is then seen with a great variety of illuminations due to the varied orientations of surface normals as well as casted shadows.

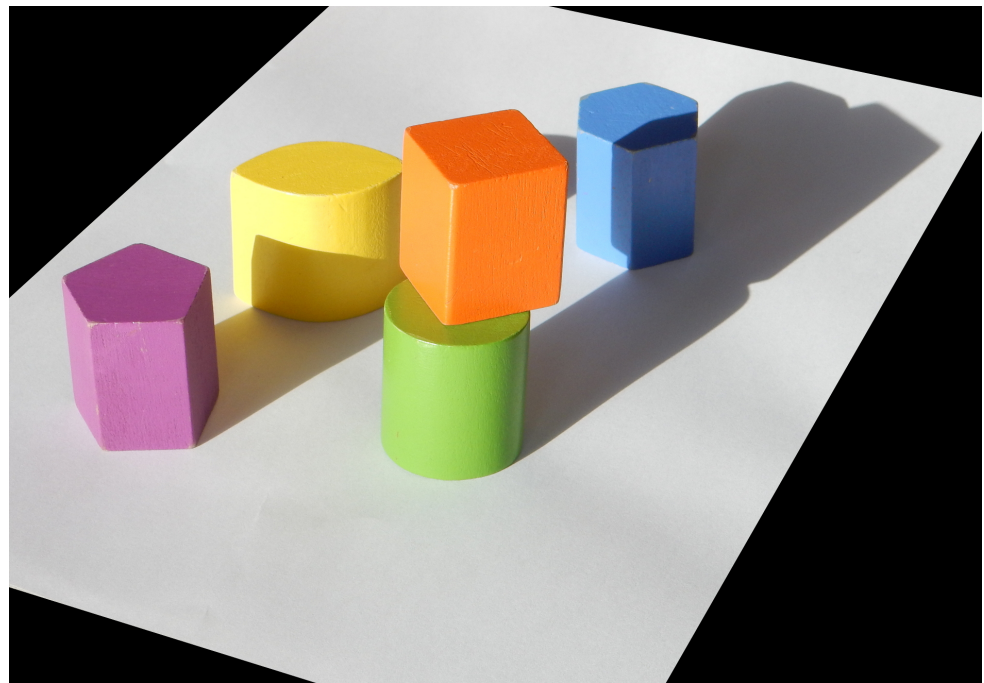


Figure 3. Katell’s wooden toys under natural sunlight. Note the specular reflection on the front edge of the orange cube, for example, which appears almost white.

Figure 4 shows both the image resulting from the transformation and the location of transformed pixel values in the RGB space. It can be seen that the spectral invariant $\mathbf{L}^{(M)}$ always lies on a triangle-shaped ‘wire-frame’, consisting of three connected rods: the frame is three-dimensional but, just as the color wheel, it basically provides only one degree of freedom. This rather complicated shape results from the choice of subtracting the pixel-wise *minimum* of all bands in the first step: $\mathbf{L}^{(M)}$ has always at least one entry equal to zero, but it is not always the same. As a result, it always lies on one of the faces of the unit cube. The normalization step yields another property: the entries sum to 1. If we

define the ‘white’ vector $\mathbf{w} = [1, 1, \dots, 1]^T$ having all entries equal to 1, the normalization can be written as a scalar product:

$$\sum_j L_j^{(M)} = 1 \iff \mathbf{L}^{(M)} \cdot \mathbf{w} = 1 = \text{constant} \quad (8)$$

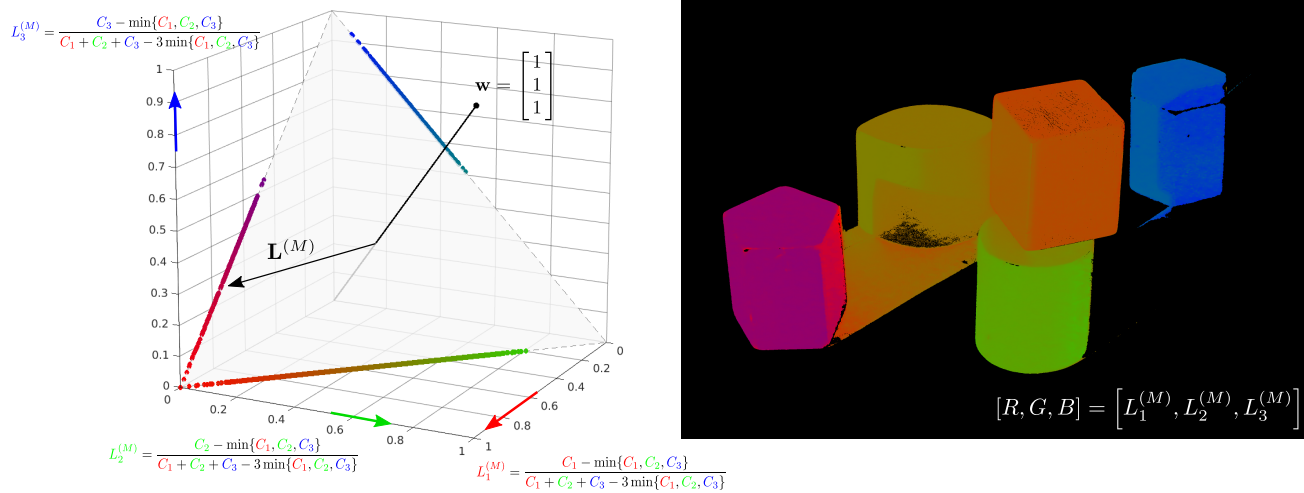


Figure 4. Spectral invariant proposed by Montoliu et al., computed on the image of Figure 3. **(Left panel):** location of transformed pixel values in the RGB unit cube, invariant $\mathbf{L}^{(M)}$ is constrained in a plane orthogonal to the white vector $\mathbf{w} = [1, 1, 1]^T$. **(Right panel):** transformed RGB image. Note how light reflected from the yellow oval toy produces a yellow hue on the white floor and also alters the hue on the rightmost side of the purple pentagon toy.

Hence, $\mathbf{L}^{(M)}$ is in a subspace (here, a plane) orthogonal to the ‘white’ direction. If we rotate this ‘white’ axis in order to align it onto the last Cartesian axis $[0, 0, 1]^T$, the triangle lies in the plane \mathbb{R}^2 and each side can be mapped linearly onto a 120° arc of a circle: the popular RGB color wheel is actually built in the exact same way (Hue–Saturation–Value model).

Obviously, the invariant cannot be computed for gray pixels having the same digital count in all bands, since we would have:

$$L_i^{(M)} = \frac{c_i}{\sum_{j=1}^n c_j} = \frac{C_i - \min_k \{C_k\}}{\sum_{j=1}^n (C_j - \min_k \{C_k\})} = \frac{0}{0} \quad (9)$$

In practice, we remove all ‘nearly gray’ pixels, such that

$$0 \leq \left(\sum_{j=1}^n c_j \right) < \varepsilon \quad (10)$$

It is worth noting that this will produce gaps in the image.

3.2. Multispectral Hue as a Directional Variable

The definition of the multispectral invariant $\mathbf{L}^{(M)}$ given by Montoliu et al. [16] essentially spells out the two basic steps needed to define a multispectral hue. Even though we can interpret these two steps in terms of resaturation and brightness normalization, they are essentially a way of reducing the dimensionality of the problem and minimizing the degrees of freedom for hue analysis. The only drawback of the definition is the topology of the subset in which the invariant lies. Therefore, we propose an alternate definition, which will yield a much smoother topology.

3.2.1. Mathematical Definition

In the resaturation step, we choose to remove the pixel-wise *mean* of all bands, instead of the minimum. We obtain the vector \mathbf{c}' with entries:

$$c'_i = C_i - \frac{1}{n} \sum_{j=1}^n C_j \tag{11}$$

Then, instead of normalizing with the sum of the resulting entries, we simply normalize this vector with its Euclidean norm, yielding the vector invariant \mathbf{U} :

$$\mathbf{U} = \frac{\mathbf{c}'}{\|\mathbf{c}'\|} \tag{12}$$

The entries of this vector read explicitly:

$$U_i = \frac{c'_i}{\sqrt{\sum_{j=1}^n c_j'^2}} = \frac{C_i - \frac{1}{n} \sum_{j=1}^n C_j}{\sqrt{\left(\sum_{j=1}^n C_j^2\right) - \frac{1}{n} \left(\sum_{j=1}^n C_j\right)^2}} = \frac{C_i - \text{Mean}_j\{C_j\}}{\sqrt{n-1} \text{StDev}_j\{C_j\}} \tag{13}$$

3.2.2. Illustration with the RGB Image

Just as we did for the invariant $L^{(M)}$ of Montoliu et al. [16], in Figure 5 we show the resulting transformed image and the location of transformed pixel values in the invariant space. It is worth noting that the transformation does not preserve the positivity of entries U_i . However, since \mathbf{U} has unit norm, it lies in the $[-1, 1] \times [-1, 1] \times \dots \times [-1, 1]$ cube and $\frac{U_i+1}{2}$ will remain inside the positive-valued unit cube; we use these values to reconstruct the R, G, and B components.

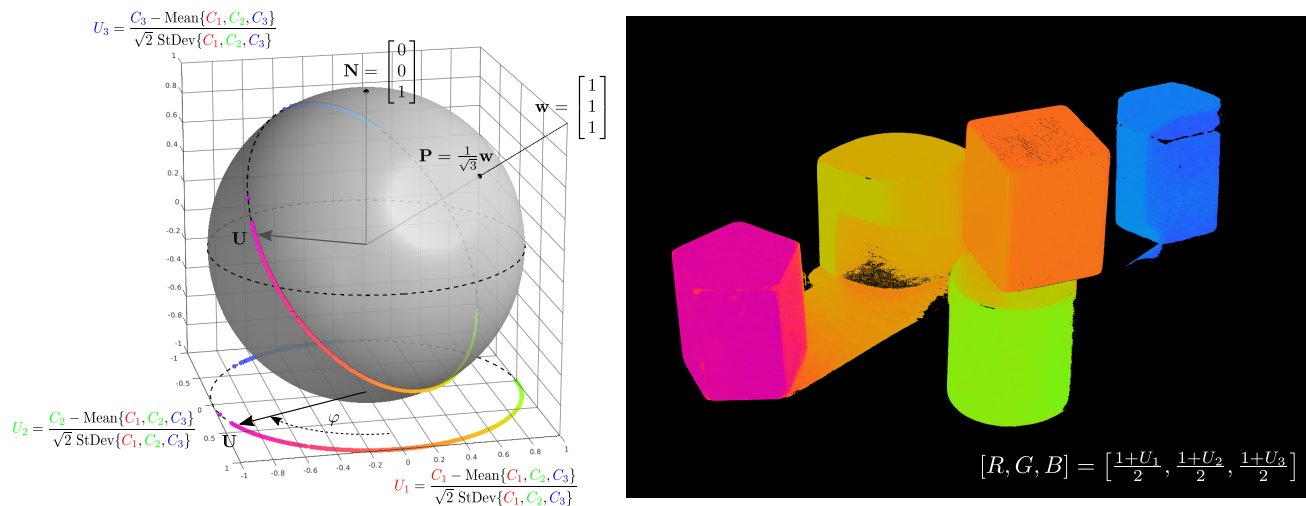


Figure 5. Illustration of multispectral hue \mathbf{U} computed on the RGB image of Figure 3. (Left panel): location of transformed pixel values on a great circle in a plane orthogonal to white vector $\mathbf{w} = [1, 1, 1]^T$. (Right panel): transformed RGB image.

This time, it is the resaturation condition that must be interpreted as a scalar product. Centering the values implies:

$$\sum_j U_j = 0 \iff \mathbf{U} \cdot \mathbf{w} = 0 \tag{14}$$

Again, the invariant lies in a subspace (a plane in \mathbb{R}^3) orthogonal to the ‘white’ vector \mathbf{w} . Moreover, it has unit norm by virtue of the normalization step: as a result, it directly lies on a great circle (‘tilted’ equator). Rotating the white axis onto the last Cartesian axis will

again allow us to discard this last coordinate: we are left with a circle that can be described with only two Cartesian coordinates and only one angular coordinate.

This property will extend in any dimension: the multispectral hue \mathbf{U} will always belong to a subset homeomorphic to \mathbb{S}^{n-2} the unit $(n - 2)$ -sphere embedded in \mathbb{R}^{n-1} , provided that we perform a final rotation that aligns the ‘white’ direction \mathbf{w} onto the ‘North Pole’ $\mathbf{n} = [0, 0, \dots, 0, 1]^T$ of \mathbb{R}^n . The general expression of the corresponding rotation matrix is given in Appendix A.

3.2.3. Summary of the Section

In this section, we defined a multispectral hue, which is an extension of the classical RGB hue to a higher number of bands: it allows us to resaturate and normalize the brightness of each pixel, whatever the number of bands. [7]. This multispectral hue denoted \mathbf{U} is a direction (unit vector) in a $(n - 1)$ dimensional Euclidean space, and it has $(n - 2)$ angular degrees of freedom:

- With $n = 3$ bands, \mathbf{U} is on \mathbb{S}^1 the unit circle in the Euclidean plane \mathbb{R}^2 : it is closely related to the ‘color wheel’;
- With $n = 4$ bands, \mathbf{U} is on \mathbb{S}^2 the ‘usual’ unit sphere in the Euclidean space \mathbb{R}^3 . It has two degrees of freedom, we might call them ‘color latitude’ and ‘color longitude’ as will be seen later;
- With $n > 4$ bands, the hypersphere \mathbb{S}^{n-2} cannot be pictured out. However, it is still a smooth Riemannian manifold, and all familiar properties on \mathbb{S}^1 and \mathbb{S}^2 (geodesic distance, Euler angles, rotation group, etc.) can be extended to any dimension.

We can then use directional statistics to analyze hue distributions in any dimension.

3.3. Mixture Models for the Analysis of Multispectral Hue Distribution

The distribution of multispectral hues in a given image can be rather complicated. For this reason, it may be convenient to model the probability density function of this directional variable as a mixture of simpler distributions. On the color wheel \mathbb{S}^1 for the RGB image displayed in Figure 3, we can use the Von Mises distribution as a basic component; its density reads:

$$f(\varphi | \mu, \kappa) = \frac{e^{\kappa \cos(\varphi - \mu)}}{2\pi I_0(\kappa)} \tag{15}$$

where φ is the hue angle, μ the mean angle, κ a concentration parameter and I_0 the modified Bessel function of order 0 for normalization. This density can also be written with vectors; noting

$$\mathbf{U} = \begin{bmatrix} \cos \varphi \\ \sin \varphi \end{bmatrix} \quad \mathbf{v}_m = \begin{bmatrix} \cos \mu \\ \sin \mu \end{bmatrix} \tag{16}$$

then

$$f(\mathbf{U} | \mathbf{v}_m, \kappa) = \frac{e^{\kappa \mathbf{v}_m \cdot \mathbf{U}}}{2\pi I_0(\kappa)} \tag{17}$$

where \mathbf{v}_m is now the mean (vector) direction.

3.3.1. Mixture Density

Having chosen the elementary component, the mixture density is simply a weighted sum of m components:

$$f(\varphi) = \sum_{j=1}^m \pi_j f_j(\varphi | \mu_j, \kappa_j) \tag{18}$$

The resulting density still has to sum to unity on the circle, so the weights must sum to 1:

$$\int_{\mathbb{S}} f(\mathbf{U}) d\mathbb{S} = \int_0^{2\pi} f(\varphi) d\varphi = 1 \implies \sum_{j=1}^m \pi_j = 1 \tag{19}$$

Figure 6 (left) shows the modeled density of hue in our RGB image: we use a five-component model, one for each wooden shape.

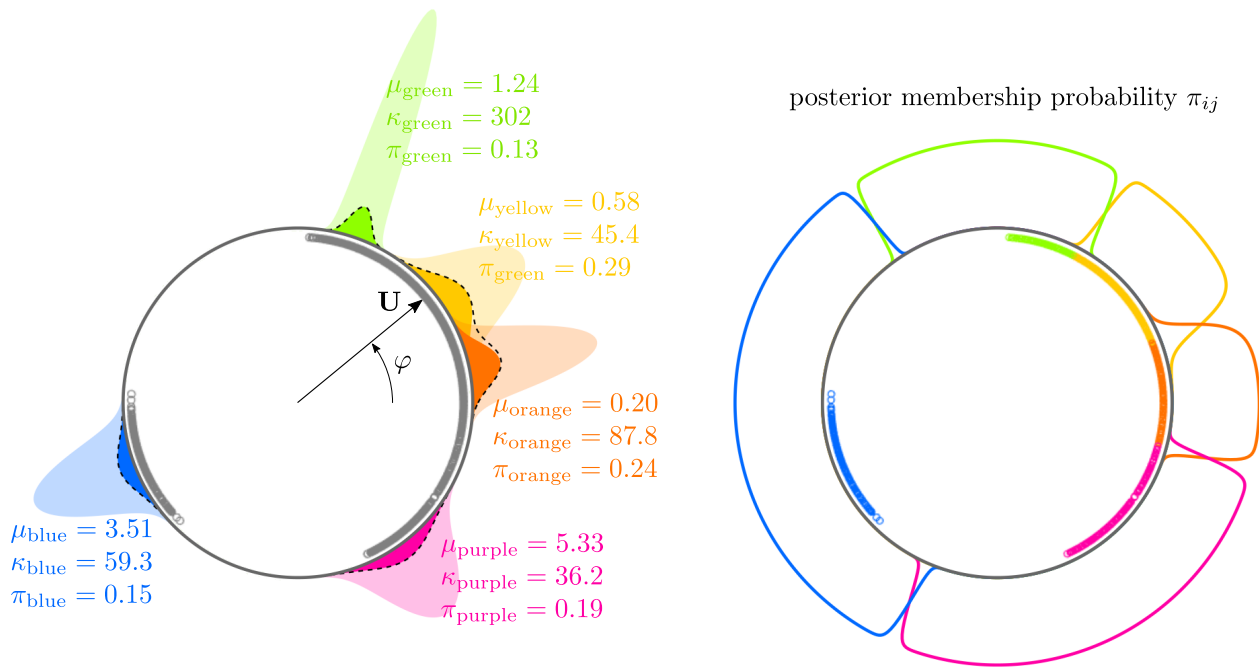


Figure 6. (Left panel) Hue distribution on the ‘wooden toys’ image (the density is the black dashed line), and its modeling with a mixture of 5 Von Mises distributions. We can check that $\pi_{\text{orange}} + \pi_{\text{yellow}} + \pi_{\text{green}} + \pi_{\text{blue}} + \pi_{\text{purple}} = 1$. (Right panel) shows the posterior membership probability π_{ij} (given by Bayes formula) that pixel i comes from component j .

3.3.2. Parameter Estimation Using Expectation–Maximization Algorithm

It is necessary to present briefly how the parameters $\{\mu_j, \kappa_j, \pi_j\}_{j=1\dots m}$ of the mixture density are estimated, because we will use a variant of this procedure for depth estimation. The procedure used is the iterative expectation–maximization (EM) algorithm [18]:

- E-step (Expectation): at each iteration (r), we start with current parameter estimates $\{\mu_j^{(r)}; \kappa_j^{(r)}; \pi_j^{(r)}\}_{j=1\dots m}$ and we define updated membership probabilities using Bayes formula:

$$\pi_{ij}^{(r+1)} = \frac{\pi_j^{(r)} f_j(\varphi_i | \mu_j^{(r)}, \kappa_j^{(r)})}{\sum_{j'=1}^m \pi_{j'}^{(r)} f_{j'}(\varphi_i | \mu_{j'}^{(r)}, \kappa_{j'}^{(r)})} \quad (20)$$

where $\pi_{ij}^{(r+1)}$ is the estimated probability that pixel i with hue angle φ_i comes from component j , with $\sum_{j=1}^m \pi_{ij}^{(r+1)} = 1$ for each pixel;

- M-step (Maximization): given these membership probabilities, the parameters of each component are re-estimated using a slightly modified maximum likelihood method based on the maximization of:

$$\ell_j(\mu_j, \kappa_j) = \sum_{i=1}^{N_{\text{pix}}} \pi_{ij}^{(r+1)} \log f_j(\varphi_i | \mu_j, \kappa_j) \quad (21)$$

We see that the weights in this log-likelihood are the membership probabilities, instead of just 1. $\ell_j(\mu_j, \kappa_j)$ can be maximized for each component j independently, leading to updated parameters $(\mu_j^{(r+1)}, \kappa_j^{(r+1)})$ and

$$\pi_j^{(r+1)} = \sum_{i=1}^{N_{\text{pix}}} \pi_{ij}^{(r+1)} \quad (22)$$

4. Using Multispectral Hue as a Predictor for Depth

In this section, we test our newly-defined multispectral hue as a predictor for depth on a 35-km reach of the Garonne River between Toulouse and Verdun-sur-Garonne (see Figure 7 left). This reach has a rather natural morphology with few embankments past Toulouse, and many wetlands (it is dubbed the “Garonne débordante”, which means “overflowing Garonne” in French). The average channel width is about 150 m and the maximum depth about 5–6 m at low flow. The channel is mainly bedrock, with typical sculpted molassic forms [19] and several major knickpoints with very shallow flow (depth < 1 m).

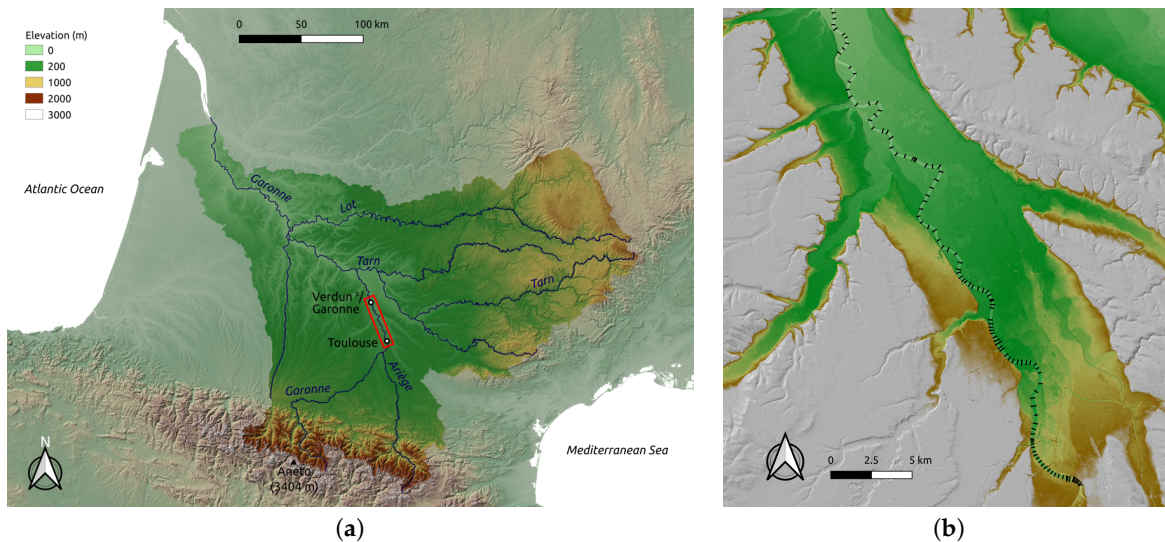


Figure 7. Location of the reach of the Garonne river investigated in this study. (a) Location of the reach in the whole Garonne catchment, between Toulouse and Verdun-sur-Garonne; (b) in situ cross-section data used for calibration.

This site has been widely used in the French remote sensing community to test discharge estimation algorithms using water surface elevation measurements such as the ones that will be produced by the joint NASA and CNES ‘Surface Water and Ocean Topography’ (SWOT) mission [20–22]. As our study contributes to these developments, we needed to produce a detailed bathymetric map on this specific reach, and we present the results in this section. As it will be shown further on, the dataset used has some flaws that prevent us from drawing very general conclusions regarding the advantages of the method. A full qualification of the method is needed and will be the object of a future publication using other publicly available datasets [23].

4.1. Dataset

4.1.1. High-Resolution Airborne Imagery

Our study is based on the BD ORTHO[®] database provided by the IGN (French National Geographic Institute). This nationwide dataset consists of RGB and Infrared Color (IRC) ortho-rectified and radiometrically corrected airborne images [24,25] with a resolution of 0.50 m. For our area of interest, we use images acquired during the period

17–30 July, 2016. This dataset provides us with four coregistered spectral bands, namely near-infrared (NIR), red, green, and blue.

4.1.2. In Situ Depth Measurements

As mentioned in Section 2, the relationship between hue and depth can be determined either physically (using radiative modeling) or empirically, using calibration against in situ data. In this study, we use the latter approach: the relationship is calibrated against in situ bathymetric data collected at 120 cross-sections along the reach (about 1 section every 400 m). These data were collected during surveys in the period 1987–2002: hence, this end-of-20th-century bathymetry may differ substantially from present bathymetry locally, even for a bedrock channel (this point will be discussed at the end of the paper).

4.1.3. LiDAR Data in the Floodplain

In order to better constrain the estimation of river depth, we use LiDAR elevation data available in the floodplain. These data come from the BDORTHO[®] database also provided by IGN. Even though raster tiles are without gaps, elevation data in the river channel must be ignored: the value at a pixel located in the channel is simply the result of an interpolation between nearby bank pixels. In order to filter out interpolated pixels, the BDORTHO[®] elevation rasters come with useful ancillary DST (dist) rasters, which indicate the distance of each pixel to the nearest valid LiDAR point (see Figure 8 right).

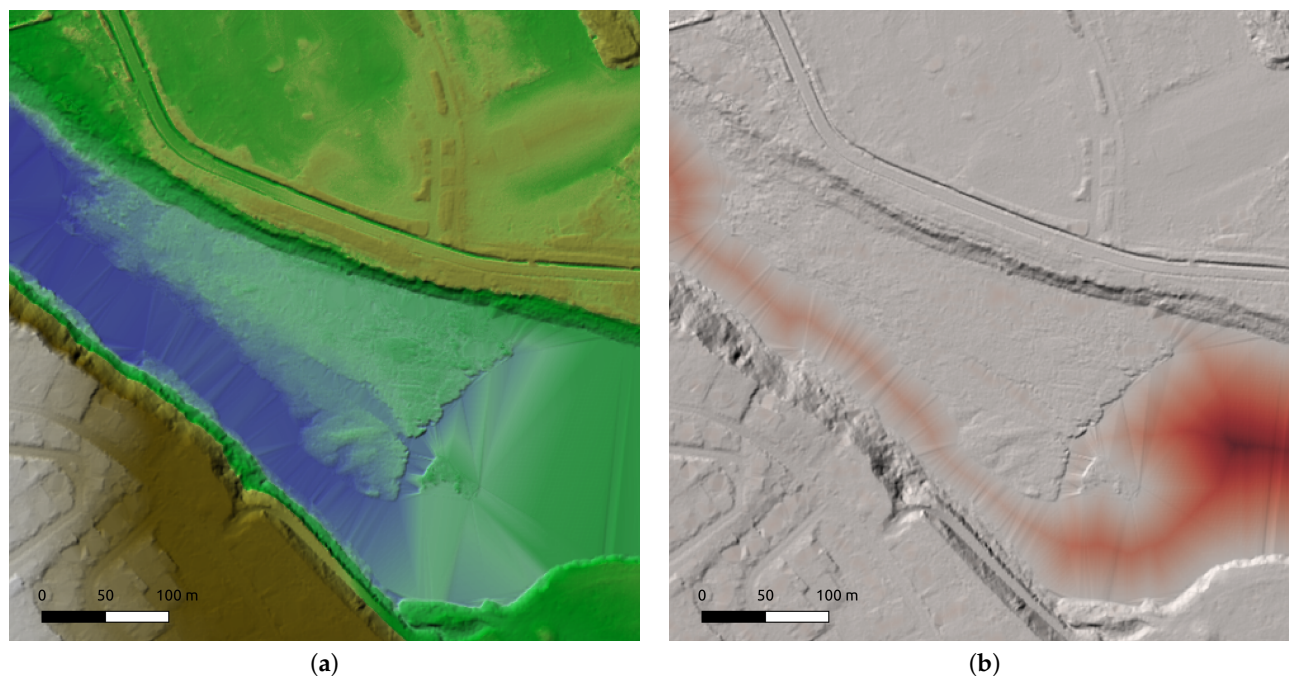


Figure 8. Sample of the IGN BDORTHO[®] database covering the floodplain. (a) Elevation raster at 1-meter planar resolution; (b) DST raster indicating the distance to the nearest valid LiDAR point for each pixel.

4.2. Hue Sphere \mathbb{S}^2 for $n = 4$ Bands

According to the approach presented in Section 3, with $n = 4$ the invariant \mathbf{U} does not lie on the circle (1-sphere \mathbb{S}^1) but on the $(n - 2)$ -sphere, i.e., the classical sphere \mathbb{S}^2 . In Appendix B, we give the computational detail of the mapping of a 4-band pixel ($\mathbf{C} \in \mathbb{R}^4$) onto the unit sphere \mathbf{U} on \mathbb{S}^2 . Here is the final mapping of the four ‘monochromatic’ points, in Cartesian coordinates of \mathbb{R}^3 :

$$\begin{aligned} C_{\text{NIR}} &= [1, 0, 0, 0]^T \longrightarrow U_{\text{NIR}} = \frac{1}{3\sqrt{3}} [5, -1, -1]^T \\ C_{\text{Red}} &= [0, 1, 0, 0]^T \longrightarrow U_{\text{Red}} = \frac{1}{3\sqrt{3}} [-1, 5, -1]^T \\ C_{\text{Green}} &= [0, 0, 1, 0]^T \longrightarrow U_{\text{Green}} = \frac{1}{3\sqrt{3}} [-1, -1, 5]^T \\ C_{\text{Blue}} &= [0, 0, 0, 1]^T \longrightarrow U_{\text{Blue}} = \frac{1}{3\sqrt{3}} [-3, -3, -3]^T \end{aligned}$$

In order to illustrate how the hue (hyper)sphere works, we show in Figure 9 where two patches will map, along with the four ‘monochromatic’ points mentioned above. The first patch (A) is composed of water pixels: as absorption is much stronger in IR than in any of the visible bands, it is no surprise that these pixels will map close to the ‘anti-NIR’ hue, corresponding to the multispectral hue of a pixel with digital counts $[0, 1, 1, 1]^T$. Conversely, a patch of vegetation (B) will map close to the pure NIR hue, as absorption is not the same in all visible bands but still much stronger than in the NIR.

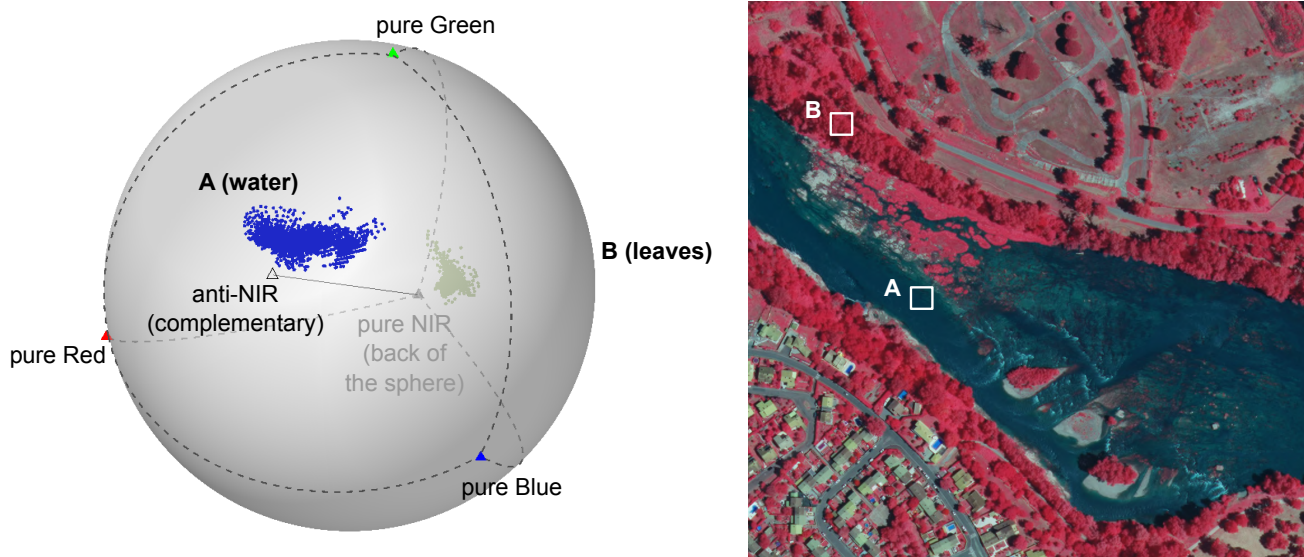


Figure 9. Representation of multispectral hue on the sphere S^2 for two terrain patches: a patch of water pixels (A) and a patch of vegetation pixels (B). Due to the low absorption in the visible bands and high absorption in the IR for water, and the opposite behavior for chlorophyll, the two patches map at the antipode of each other (close to the anti-NIR hue for water, and close to pure NIR hue for vegetation). The image displayed on the right is a NIR-R-G composite.

4.3. Masks

In order to reduce the extent of the area investigated and to narrow the spread of hue distributions, several boolean masks are applied on the raw images in order to retain only river pixels, as illustrated in Figure 10:

- Riparian vegetation as well as floating algae are masked using the classical Normalized Difference Vegetation Index (NDVI) [26], which indicates high chlorophyll content. In practice, the rather low threshold selected ($\text{NDVI} > -0.3$) excludes more than just vegetation pixels: all surfaces with any non-negligible reflectance in the IR will be masked;
- Very dark areas are thoroughly removed: they mainly consist of water pixels in the shadow of riparian vegetation; as these pixels are difficult to characterize spectrally

(light has traveled through both leaves and water), the criterion used is simply the mean of the four bands;

- A filter is designed to remove whitewater/white wakes in riffle areas: as these pixels appear in light gray in the visible bands (almost hueless in RGB) but with substantial absorption in the IR (typical digital counts are [0.2 0.6 0.6 0.6]), they are removed on the basis of their low saturation and high brightness in the RGB space along with a threshold in the NIR ($C_{\text{NIR}} < 0.3$). Note that they have nothing to do with specular reflections: the surface of water appears white there for any viewing angle, because of the scattering from the bubbles/air pockets, not from surface reflection;
- A mask for man-made features crossing the river, such as bridges or power lines as well as their casted shadows, is built manually (some of these pixels are already masked by the previous filters, but not all).

Once all these masks are combined, a final topological erosion of the resulting mask is performed.

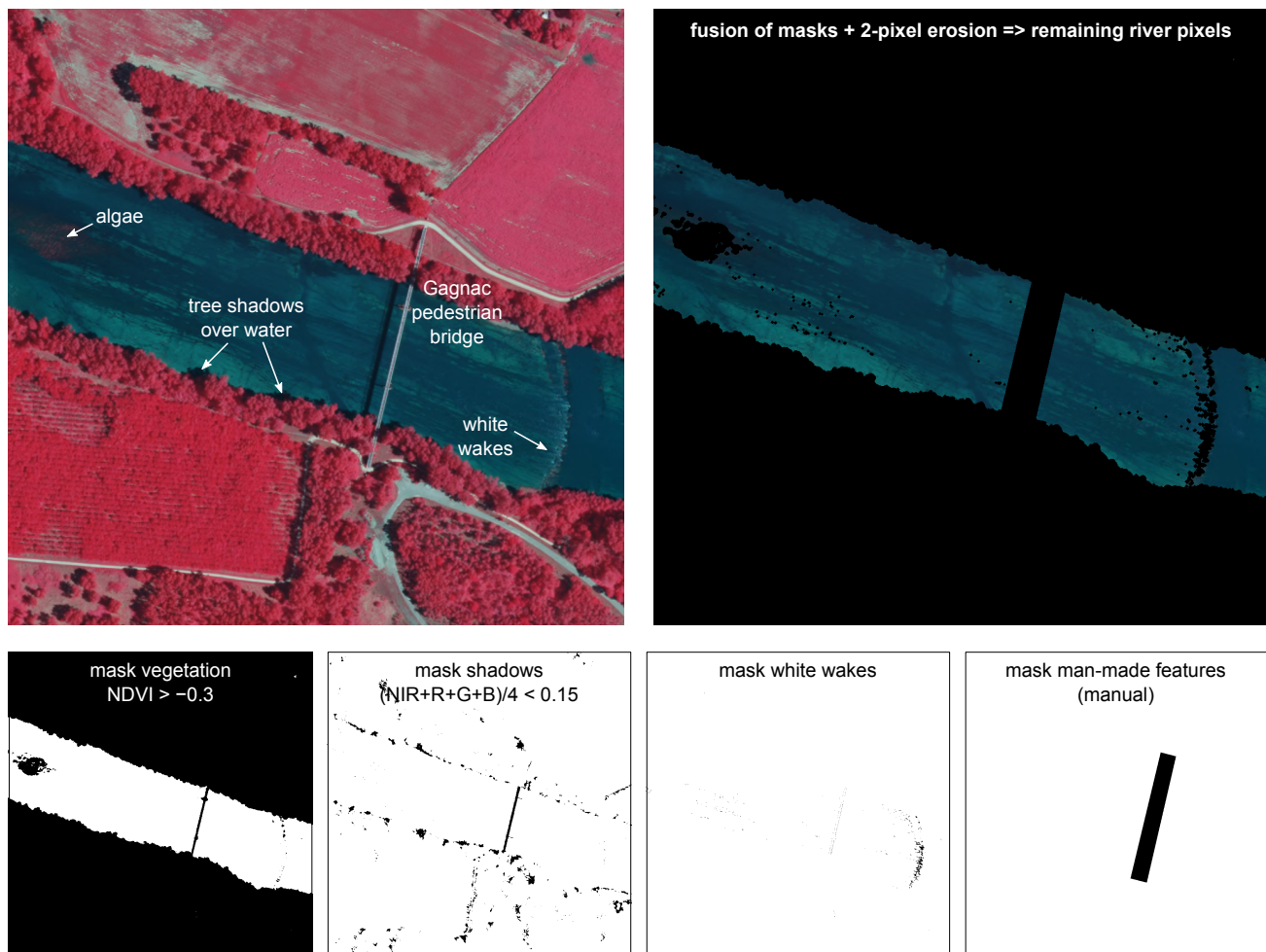


Figure 10. Illustration of the four masks applied to multispectral images in order to retain only river pixels. The image displayed is a NIR-R-G composite.

4.4. Hue–Depth Visual Correlation on \mathbb{S}^2

We now focus on river pixels with a valid multispectral hue. We can first look at the projection onto the sphere of all pixels located at depth measurement points. What we see first in Figure 11 is that they are located close to the anti-NIR hue, as already mentioned above. However, if we plot them with measured depth as color, we see that

‘deep’ pixels tend to concentrate close to a single pole of the sphere. This result is perfectly consistent with Philpot’s Equation (2): as depth increases, the weight of bed reflectance R_b in the mixture reflectance vanishes to zero and the digital counts of river pixels, which are basically reflectances, should tend to the limit vector

$$\mathbf{C}_{c,\infty} \sim \begin{bmatrix} R_{c,\infty}(\lambda_{\text{NIR}}) \\ R_{c,\infty}(\lambda_{\text{Red}}) \\ R_{c,\infty}(\lambda_{\text{Green}}) \\ R_{c,\infty}(\lambda_{\text{Blue}}) \end{bmatrix}$$

The hue associated with $\mathbf{C}_{c,\infty}$ is thus some point $\mathbf{U}_{c,\infty}$ on \mathbb{S}^2 , that we can roughly locate visually. Here, ‘infinitely’ deep means about 5 to 6 m deep.

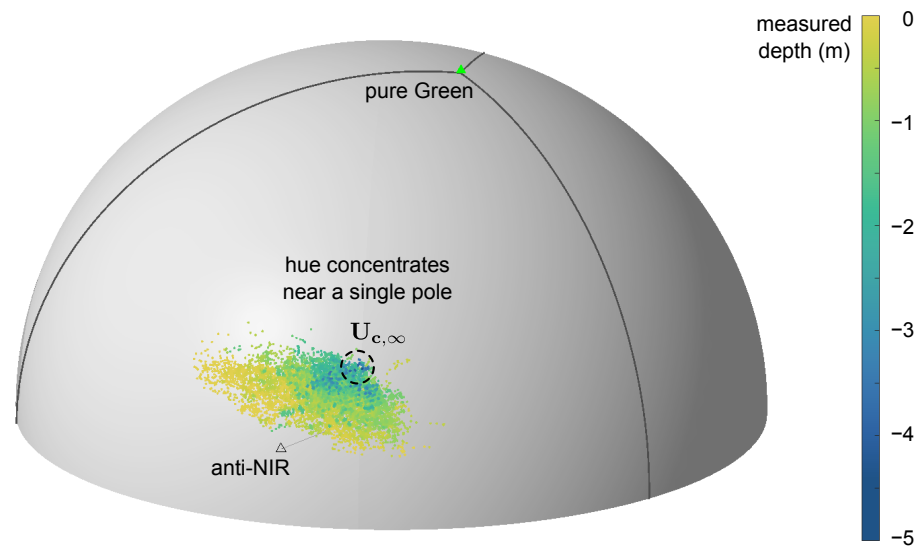


Figure 11. Representation of multispectral hue on \mathbb{S}^2 for the pixels located at depth measurement points. As depth increases, hue tends to concentrate near a single pole corresponding to the hue of a very deep water column, denoted $\mathbf{U}_{c,\infty}$.

4.5. Mixture Models on \mathbb{S}^2 and Higher Dimension, and Depth Predictor

Our last step is to find a quantitative predictor for depth. The two principles are the following:

- We assume that the probability density function of multispectral hue in river pixels can be modeled by a two-component mixture density: the first component will represent the statistical distribution of substrate hue, while the second component will represent the statistical distribution of ‘deep water’ hue;
- We will estimate the parameters of the two components so that the membership probability to ‘deep’ component correlates with depth: the probability $\pi_{i,\text{deep}}$ that pixel i with hue \mathbf{U}_i comes from the ‘deep’ component will be the predictor for depth h_i at pixel i .

The elementary component used on the unit circle \mathbb{S}^1 in Section 3 is the Von Mises distribution (‘circular normal distribution’). On the unit sphere \mathbb{S}^2 and higher dimension, we need an extension of this distribution, similar to a multivariate normal distribution wrapped on the (hyper)sphere. Such an extension is the Fisher–Bingham–Kent (FBK) distribution [27]; for the sake of clarity, basic details about this family of distributions are provided in Appendix C. For $\mathbf{U} \in \mathbb{S}^{n-2}$, the density reads:

$$f(\mathbf{U} | \{\mathbf{v}_j\}, \kappa, \boldsymbol{\beta}) = \frac{1}{\mathcal{C}(\kappa, \boldsymbol{\beta})} \exp\left\{ \kappa \mathbf{v}_1 \cdot \mathbf{U} + \sum_{j=2}^{n-1} \beta_j (\mathbf{v}_j \cdot \mathbf{U})^2 \right\} \quad (23)$$

The $(n - 1)$ vectors $\{\mathbf{v}_j\}$ for $1 \leq j \leq (n - 1)$ form an orthonormal basis in which the first vector \mathbf{v}_1 is the mean direction of the distribution, as f is maximum when $\mathbf{v}_1 \cdot \mathbf{U} = 1$. The normalization constant $\mathcal{C}(\kappa, \beta)$ does not have a closed form, but it can be evaluated numerically [28,29]. The number of degrees of freedom increases almost quadratically with the number of bands ($\sim \frac{1}{2}n^2$).

Figure 12 illustrates the decomposition of a given density on \mathbb{S}^2 into two FBK components. In our case, the two components will be the distributions f_{bed} of bed substrate hue and f_{deep} of deep water hue, respectively; for the sake of concision, we denote Θ_{bed} and Θ_{deep} the parameter set describing each component, so that the mixture density reads:

$$f(\mathbf{U}) = \pi_{\text{deep}} f_{\text{deep}}(\mathbf{U} | \Theta_{\text{deep}}) + (1 - \pi_{\text{deep}}) f_{\text{bed}}(\mathbf{U} | \Theta_{\text{bed}})$$

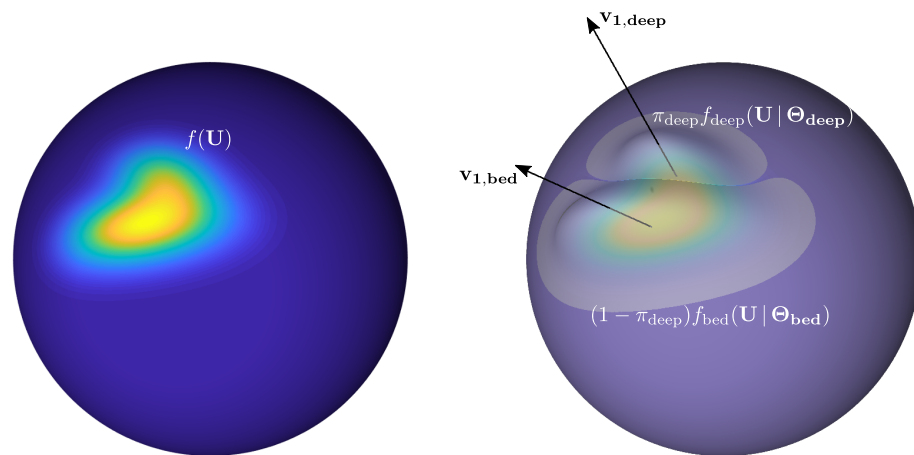


Figure 12. The probability density function $f(\mathbf{U})$ of a hue distribution on the sphere (left) can be modeled with a mixture of two or more Fisher–Bingham–Kent distributions (right). For the distribution of hue over river pixels, we used two components denoted f_{bed} and f_{deep} with respective parameters sets Θ_{bed} and Θ_{deep} : $\mathbf{v}_{1,\text{bed}}$ and $\mathbf{v}_{1,\text{deep}}$ to denote the mean direction of each component.

4.6. Modified EM Algorithm for Depth Estimation

The goal of the calibration process is, in the end, to be able to correlate depth h_i at an ‘unmeasured’ pixel i having multispectral hue \mathbf{U}_i , with the posterior membership probability to the deep component:

$$\pi_{i,\text{deep}} = \frac{\pi_{\text{deep}} f_{\text{deep}}(\mathbf{U}_i | \Theta_{\text{deep}})}{\pi_{\text{deep}} f_{\text{deep}}(\mathbf{U}_i | \Theta_{\text{deep}}) + (1 - \pi_{\text{deep}}) f_{\text{bed}}(\mathbf{U}_i | \Theta_{\text{bed}})}$$

Recall that the ‘posterior’ term means ‘after we get the additional information of pixel hue’. In contrast, π_{deep} is the ‘prior’, general probability that a randomly chosen pixel comes from the deep component (it is a constant, corresponding to the weight of the deep component in the mixture distribution): once we obtain the multispectral hue \mathbf{U}_i for a particular pixel i , we can update the prior membership probability using Bayes formula to obtain the individual posterior membership probability $\pi_{i,\text{deep}}$.

In order for this posterior probability to be correlated with depth, we use a slightly modified version of the expectation–maximization algorithm presented in Section 3, where we first estimated membership probabilities for each pixel (expectation E-step) and then updated the parameters of each component independently using the updated membership probabilities (maximization M-step). Here, we will add an intermediate regression (R-step) between these two steps: instead of feeding the maximization algorithm directly with the membership probabilities $\pi_{i,\text{deep}}^{(r+1)}$ for each pixel, we will regress the $\pi_{i,\text{deep}}^{(r+1)}$ against measured

depth h_i and feed the maximization algorithm with *regressed* membership probabilities $\hat{\pi}_{i,\text{deep}}^{(r+1)}(h_i)$. Here is the workflow performed at each iteration:

- E-step (Expectation): given current parameter estimates $\{\pi_{\text{deep}}^{(r)}; \Theta_{\text{deep}}^{(r)}; \Theta_{\text{bed}}^{(r)}\}$ of the mixture at iteration (r), compute the updated probability that pixel i with multispectral hue U_i comes from a deep component:

$$\pi_{i,\text{deep}}^{(r+1)} = \frac{\pi_{\text{deep}}^{(r)} f_{\text{deep}}(\mathbf{U}_i | \Theta_{\text{deep}}^{(r)})}{\pi_{\text{deep}}^{(r)} f_{\text{deep}}(\mathbf{U}_i | \Theta_{\text{deep}}^{(r)}) + (1 - \pi_{\text{deep}}^{(r)}) f_{\text{bed}}(\mathbf{U}_i | \Theta_{\text{bed}}^{(r)})}$$

- R-step (Regression): regress deep membership probability as a power law function of measured depth so that

$$\hat{\pi}_{i,\text{deep}}^{(r+1)} = a^{(r+1)} (h_i)^{b^{(r+1)}}$$

- M-step (Maximization): independently maximize the log-likelihood for each component

$$\begin{cases} \ell_{\text{deep}}(\Theta_{\text{deep}}) = \sum_{i=1}^{N_{\text{depth}}} \hat{\pi}_{i,\text{deep}}^{(r+1)} \log f_{\text{deep}}(\mathbf{U}_i | \Theta_{\text{deep}}) & \implies \text{updated } \Theta_{\text{deep}}^{(r+1)} \\ \ell_{\text{bed}}(\Theta_{\text{bed}}) = \sum_{i=1}^{N_{\text{depth}}} (1 - \hat{\pi}_{i,\text{deep}}^{(r+1)}) \log f_{\text{bed}}(\mathbf{U}_i | \Theta_{\text{bed}}) & \implies \text{updated } \Theta_{\text{bed}}^{(r+1)} \end{cases}$$

Finally, we update the prior membership probability π_{deep} (weight of the deep component in the mixture), which is the mathematical expectation of individual (posterior) membership probabilities:

$$\pi_{\text{deep}}^{(r+1)} = \frac{1}{N_{\text{depth}}} \sum_{i=1}^{N_{\text{depth}}} \hat{\pi}_{i,\text{deep}}^{(r+1)}$$

where N_{depth} is the total number of in situ depth measurement points. Then we iterate back at E-step and so on until convergence is reached.

5. Results

5.1. Error Statistics

The modified expectation–maximization algorithm has several outputs after convergence: the final parameters $\{\pi_{\text{deep}}; \Theta_{\text{deep}}; \Theta_{\text{bed}}\}$ of the mixture, but also the regression parameters (a, b) such that for each pixel i

$$\hat{\pi}_{i,\text{deep}} = a h_i^b$$

This output curve is shown in Figure 13, left panel. Taking $h_{\text{max}} = a^{-1/b}$, this relation can be reversed in order to express depth as a function of (known) posterior membership probability:

$$\hat{h}_i = h_{\text{max}} \pi_{i,\text{deep}}^{1/b}$$

Since $\pi_{i,\text{deep}}$ ranges between 0 and 1, estimated depth cannot exceed the ‘saturation’ value h_{max} , which is found to be about 4.9 m. In other words, when hue is close to the maximum of the density of the ‘deep’ component, all we can say is that depth is *at least* h_{max} , but could be much larger. This is a reasonable behavior: the model will never extrapolate to unbounded values. Figure 13, right panel, shows the resulting scatter plot for estimated vs. measured depth: the RMSE is about 0.59 m and the determination coefficient is $r^2 = 0.55$.

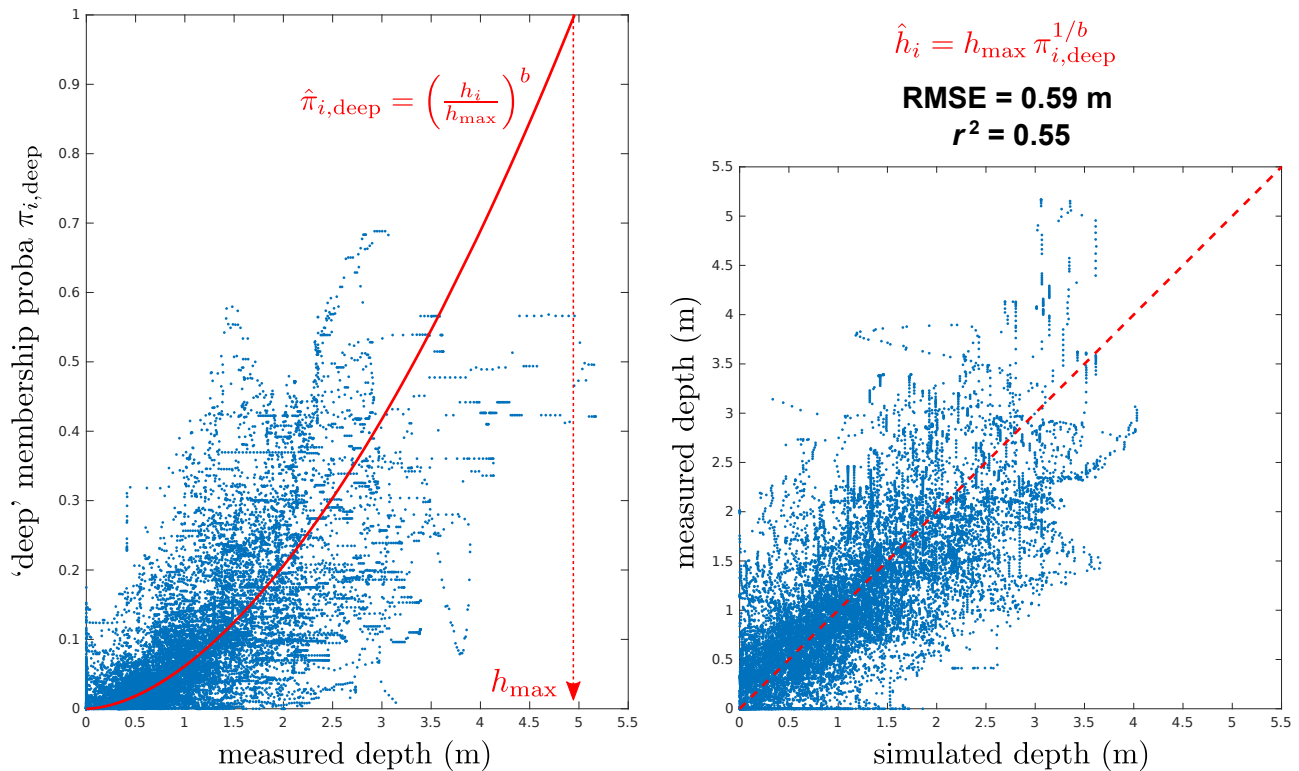


Figure 13. (Left panel): 'depth' vs. 'deep membership probability' regression as outputted by the modified expectation–maximization. This relation is of course meant to be used in the reverse direction in predictive mode (right panel): given the posterior membership probability $\pi_{i,deep}$ computed at pixel i with multispectral hue U_i , we can estimate depth h_i at this pixel.

In Figure 14, we plot the correlation between measured depth and the log of all spectral ratios that can be defined with four bands. Obviously, no single pair of bands exhibits a correlation stronger than the one identified on the hue hypersphere. Most noticeably, none of the spectral ratios is able to identify large depths unambiguously.

A simple benchmark is provided by a Multiple Linear (ML) regression using all independent log-ratios (i.e., ratios between pairs of adjacent bands). It is a specific type of MODPA analysis [15] and the resulting optimal combination of predictors reads:

$$h_{ML} = 2.60 \log\left(\frac{\text{Red}}{\text{NIR}}\right) + 11.48 \log\left(\frac{\text{Green}}{\text{Red}}\right) - 7.31 \log\left(\frac{\text{Blue}}{\text{Green}}\right) - 7.39 \quad (24)$$

The result is shown in Figure 15: the model yields an RMSE of 0.58 m, roughly similar to our procedure. However, the correlation is substantially lower: due to the least-square nature of the model, the variance of estimated depth is lower than the variance of measured depth and the model is not able to yield depths larger than 2.5 m. In contrast, the mixture model on the hypersphere yields depths as large as 4 m.

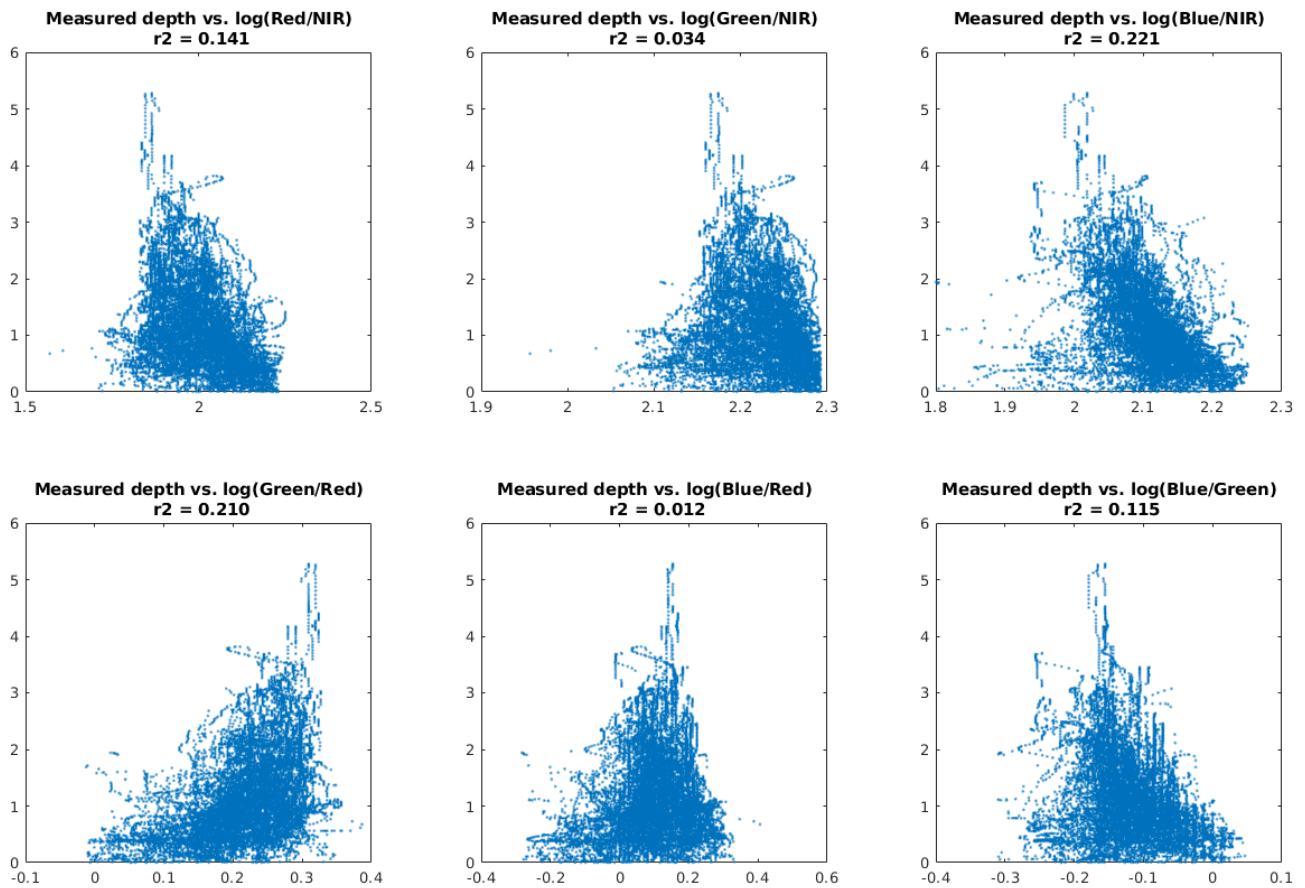


Figure 14. Correlation between measured depth and the log of the six different spectral ratios that can be defined with four-band imagery.

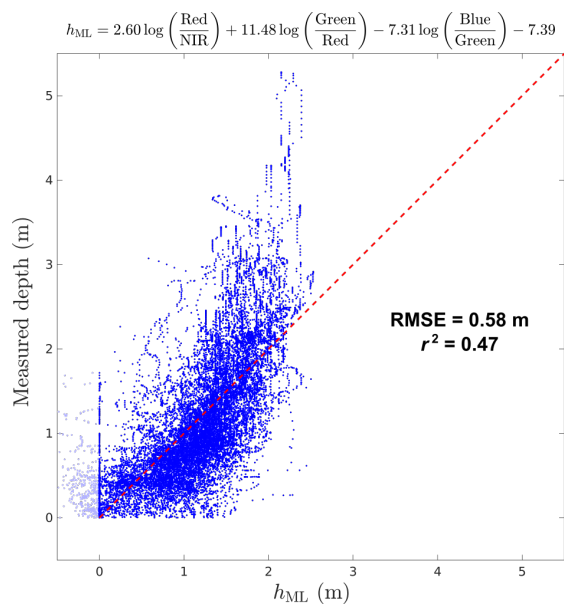


Figure 15. Result of a Multiple Linear (ML) regression against all three independent log-ratios. Negative predicted values have been set to zero before the computation of RMSE and r^2 .

5.2. Focus on Some Cross-Sections

In Figure 16, we show measured and simulated depth at two representative cross-sections. This illustrates how the method is able to identify localized bed features and high-frequency variations in bathymetry, which are typical of the bedrock channel of the Garonne Toulousaine [19].

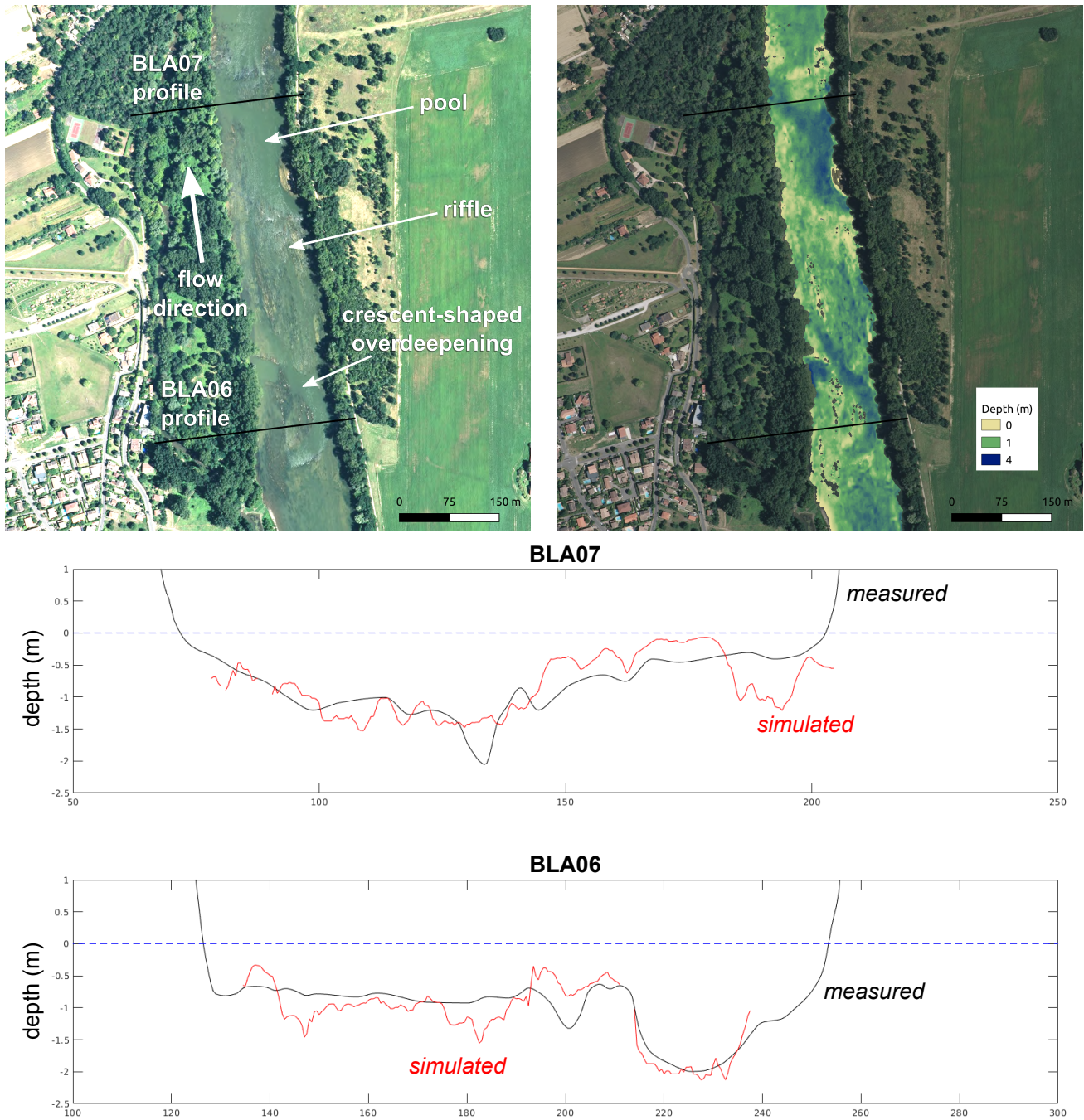


Figure 16. Comparison between measured and simulated depth along two surveyed cross-sections. The method is able to capture high-frequency variations in water depth: a typical example is the crescent-shaped feature visible at the bottom of the image, which is intersected by profile BLA06.

6. Discussion

6.1. Sources of Error

6.1.1. Radiometric Inhomogeneity

The BDORTHO[®] images used in this study form a nationwide dataset covering all 101 French departments. In practice, these images are radiometrically equalized in order to provide a seamless mosaic at the scale of each department [24,25]. There remain some variations in radiometric aspects inside the mosaic, and especially at the border between two departments. As the investigated reach spans two departments, namely Haute-Garonne (31) and Tarn-et-Garonne (82), the dataset is clearly not radiometrically homogeneous, and this increases the variability in both substrate and deep water hue. This additional variability reduces the precision of the hue—depth relation, but since the method is designed to deal with a dispersion in the distribution of both components through parameter sets Θ_{bed} and Θ_{deep} of the Fisher–Bingham–Kent distributions, the radiometric inhomogeneity can be statistically treated together with the natural variations. The key parameter acting as a quantification of non-uniformity is the concentration parameter κ_{bed} of the bed hue distribution. This is an important advantage of the method over log-ratio methods, which rely on the assumption that reflectance ratios $\frac{R_b(\mathbf{x}, \lambda_i)}{R_b(\mathbf{x}, \lambda_j)}$ are rather uniform spatially (see Section 2).

6.1.2. Age and Planar Precision of In Situ Data

As mentioned in Section 4.1.2, the bathymetric data used in this study were collected during surveys in the period 1987–2002. This end-of-twentieth-century bathymetry may differ substantially from present bathymetry locally, even for a bedrock channel. Moreover, a precise positioning of profile with GPS was not always available: errors up to several meters on the transverse position of cross-section points were obvious for a few of them, and likely for many. Profiles were manually repositioned where it was possible, but the remaining uncertainty is an important source of error.

6.1.3. Water Surface Elevation

The airborne images were taken at low flow during the month of July, 2016. However, we do not have a corresponding water surface profile of the reach for the same period; as in situ data consists of absolute bed elevation z_b (in meters a.s.l) and not depth, we needed an estimated map of water surface elevation z_{ws} at low flow, which we derived from a mean of two estimates:

1. The first estimate was simply given by the lowest valid elevation at each cross-section in the LiDAR DEM of the floodplain (which also covers the channel through interpolation between the banks). Indeed, the LiDAR survey was conducted during the 2012 low flow period, in hydraulic conditions that we assumed roughly similar to those of the 2016 images;
2. The second estimate is the elevation that yields the same width as the one observed in the 2016 images (as defined by the NDWI mask), for each surveyed profile. We then interpolate linearly between the surveyed cross-sections.

We then corrected the values so that our estimated low flow surface elevation profile has strictly monotonically decreasing values. Even if the estimated water surface profile seems reasonable, the uncertainty in water surface elevation, and hence in field estimate of water depth $h = z_{ws} - z_b$, may be several tens of centimeters locally. As large depths drive the parameter estimation, it does not hinder the possibility to identify a robust hue–depth relation, though.

6.2. Perspectives

This work is a proof-of-concept that large-scale identification of river bathymetry is possible using publicly available nationwide datasets. The bathymetric map of the Garonne River produced in this study will be used as an input for reach-scale hydraulic modeling

and the analysis of reach-scale hydraulic geometry relationships following [2], with a spatial resolution that could never have been achieved on the sole basis of existing in situ data. However, the shortcomings of the Garonne dataset, listed in Section 6.1, make it necessary to continue the work using higher-quality data.

Author Contributions: Conceptualization, N.L.M. and M.M.; software, N.L.M.; data curation and validation, N.L.M. and M.M.; supervision and project administration, N.L.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received partial external funding through the Ph.D. grant of Mounir Mahdade (SU C18/1749-S18LRAP039/CNES 510004634).

Data Availability Statement: The tiles of the BDORTHO[®], dataset used in this study are publicly available from IGN Open-Access portal (accessed 14 December 2020): <https://geoservices.ign.fr/documentation/diffusion/telechargement-donnees-libres.html>.

Acknowledgments: We thank the Direction Régionale de l'Environnement, de l'Aménagement et du Logement (DREAL) Occitanie and the Syndicat Mixte d'Études et d'Aménagement de la Garonne (SMEAG), as well as Hélène Roux and André Paquier, for providing the cross-section data. We thank Katell for providing the experimental material of Section 3.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. General Expression of the Rotation Matrix Needed for Computing the Multispectral Hue in Any Dimension

Let \mathbf{w} and \mathbf{n} be two unit vectors of vectors of \mathbb{R}^n (i.e., two points of \mathbb{S}^{n-1}). The n -dimensional rotation matrix sending \mathbf{w} on \mathbf{n} can be obtained as follows [30]. Noting

$$\begin{aligned}\alpha &= \arccos(\mathbf{w} \cdot \mathbf{n}) = \arccos(\mathbf{w}^T \mathbf{n}) \\ \mathbf{q} &= \frac{\mathbf{w} - \mathbf{n} \cos \alpha}{\|\mathbf{w} - \mathbf{n} \cos \alpha\|} \\ \mathbf{A} &= \mathbf{n} \mathbf{q} - \mathbf{q} \mathbf{n}\end{aligned}$$

where $\mathbf{n} \mathbf{q} = \mathbf{n} \mathbf{q}^T$ is the outer (dyadic) product, the desired rotation matrix is

$$\mathbf{M} = \mathbf{I} + (\sin \alpha) \mathbf{A} + ((\cos \alpha) - 1) (\mathbf{n} \mathbf{n} + \mathbf{q} \mathbf{q})$$

The rotation matrix allowing the transition from \mathbb{S}^{n-1} to \mathbb{S}^{n-2} in the last step of hue computations is the one sending the 'unit white vector' $\mathbf{w} = \frac{1}{\sqrt{n}} [1, 1, \dots, 1, 1]^T$ of \mathbb{R}^n to the 'North Pole' $\mathbf{n} = [0, 0, \dots, 0, 1]^T$ of \mathbb{R}^n .

In \mathbb{R}^4 , this matrix reads:

$$\mathbf{M} = \frac{1}{6} \begin{bmatrix} 5 & -1 & -1 & -3 \\ -1 & 5 & -1 & -3 \\ -1 & -1 & 5 & -3 \\ 3 & 3 & 3 & 3 \end{bmatrix} \text{ with } \mathbf{M}^{-1} = \mathbf{M}^T \Leftrightarrow \mathbf{M}^T \mathbf{M} = \mathbf{I}, \text{ and } \det(\mathbf{M}) = +1$$

Appendix B. Computational Example of the Mapping from \mathbb{R}^4 to \mathbb{S}^2 with Four Bands

In this Appendix, we detail the computations mapping a 'pure green' pixel $\mathbf{C}_{\text{Green}} = [0, 0, 1, 0]^T$ onto the unit hue sphere \mathbb{S}^2 .

- We first remove the mean of all bands to each band in order to obtain the vector \mathbf{c}' , and then we normalize it:

$$\mathbf{c}'_{\text{Green}} = \begin{bmatrix} -1/4 \\ -1/4 \\ +3/4 \\ -1/4 \end{bmatrix} \implies \frac{\mathbf{c}'_{\text{Green}}}{\|\mathbf{c}'_{\text{Green}}\|} = \frac{2}{\sqrt{3}} \begin{bmatrix} -1/4 \\ -1/4 \\ +3/4 \\ -1/4 \end{bmatrix} = \frac{1}{2\sqrt{3}} \begin{bmatrix} -1 \\ -1 \\ +3 \\ -1 \end{bmatrix}$$

- The latter vector is already on a two-sphere, but this sphere is embedded in a three-dimensional subspace of \mathbb{R}^4 whose axes do not coincide with a triplet of axes of the initial basis (this three-dimensional subspace is orthogonal to the unit ‘white’ vector $\mathbf{w} = \frac{1}{2}[1, 1, 1, 1]^T$). In order to drop one Cartesian coordinate (e.g., the last one), we rotate \mathbf{w} to the ‘North Pole’ $\mathbf{n} = [0, 0, 0, 1]^T$ of \mathbb{R}^4 , with the 4D rotation matrix \mathbf{M} such that:

$$\mathbf{M} \mathbf{w} = \mathbf{M} \begin{bmatrix} 1/2 \\ 1/2 \\ 1/2 \\ 1/2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \mathbf{n}$$

Performing this final rotation and discarding the last, the zero coordinate yields:

$$\mathbf{U}_{\text{Green}} = \mathbf{M} \frac{\mathbf{c}'_{\text{Green}}}{\|\mathbf{c}'_{\text{Green}}\|} = \frac{1}{12\sqrt{3}} \begin{bmatrix} -4 \\ -4 \\ 20 \\ 0 \end{bmatrix} \sim \frac{1}{3\sqrt{3}} \begin{bmatrix} -1 \\ -1 \\ 5 \end{bmatrix} \in \mathbb{R}^3$$

- Here we express \mathbf{U} in Cartesian coordinates of \mathbb{R}^{n-1} , but we can check that it is indeed in \mathbb{S}^{n-2} :

$$\|\mathbf{U}_{\text{Green}}\| = \frac{1}{3\sqrt{3}} \sqrt{(-1)^2 + (-1)^2 + 5^2} = \frac{1}{3\sqrt{3}} \sqrt{27} = 1$$

It is often easier to manipulate the invariant in Cartesian coordinates of \mathbb{R}^{n-1} , instead of $(n-2)$ Euler angles: one reason is that computations on a hypersphere involve many scalar products which are straightforwardly evaluated in Cartesian coordinates. However, we must always recall that the invariant has only $(n-2)$ degrees of freedom.

Appendix C. Overview of the Fisher–Bingham–Kent (FBK) Distribution

In this appendix, we provide a quick overview of the properties of the Fisher–Bingham–Kent distribution for a random vector \mathbf{x} on the $(p-1)$ -sphere \mathbb{S}^{p-1} embedded in \mathbb{R}^p . We remind that in the paper, $p = n - 1$ where n is the number of bands. The probability density function reads:

$$f(\mathbf{x}; \{\mathbf{v}_j\}, \kappa, \boldsymbol{\beta}) = \frac{1}{\mathcal{C}(\kappa, \boldsymbol{\beta})} \exp\left\{\kappa \mathbf{v}_1 \cdot \mathbf{x} + \sum_{j=2}^p \beta_j (\mathbf{v}_j \cdot \mathbf{x})^2\right\} \quad (\text{A1})$$

$$\begin{cases} \sum_{j=2}^p \beta_j = 0 \\ 0 < |\beta_j| \leq \frac{\kappa}{2} \\ \mathbf{v}_i \cdot \mathbf{v}_j = \mathbf{v}_i^T \mathbf{v}_j = \delta_{ij} \quad \text{vectors form an orthonormal basis} \end{cases}$$

$\kappa > 0$ is a concentration parameter controlling the dispersion around the mean direction \mathbf{v}_1 : indeed, the term $(\kappa \mathbf{v}_1 \cdot \mathbf{x})$ is maximum when the scalar product is 1, for $\mathbf{x} = \mathbf{v}_1$, and minimum when the scalar product is -1 , in $\mathbf{x} = -\mathbf{v}_1$ (antipode of \mathbf{v}_1). Constants β_j and vectors $\{\mathbf{v}_j\}_{j=2\dots p}$ determine the asymmetry of the distribution in the $(p-1)$ directions orthogonal to \mathbf{v}_1 .

The Fisher–Bingham–Kent distribution thus has $N = \frac{(p-1)(p+2)}{2}$ degrees of freedom, distributed in:

- Degrees of freedom corresponding to the minimum number of rotations (aside from degenerated cases) allowing the change-of-basis from the canonical basis of \mathbb{R}^p to the orthonormal basis defined by the $\{\mathbf{v}_j\}$; there are as many such rotations as independent rotation planes in \mathbb{R}^p , that is, the number of combinations of two orthogonal directions chosen from among p :

$$\binom{p}{2} = C_p^2 = \frac{p!}{(p-2)!2!} = \frac{1}{2}p(p-1)$$

- p degrees of freedom corresponding to the concentration parameter κ and the $(p-1)$ asymmetry parameters $\beta_j, 2 \leq j \leq p$,
- 1 degree of freedom removed due to the condition on the β_j

$$\sum_{j=2}^p \beta_j = 0$$

Table A1 below summarizes the number of parameters of the FBK distribution depending on the number of bands (n) or dimension of the corresponding Euclidean space $p = n - 1$ ($\mathbf{U} \in \mathbb{S}^{n-2} \subset \mathbb{R}^{n-1}$).

Table A1. Number N of degrees of freedom for the Fisher–Bingham–Kent distribution as a function of the dimension p of the initial space.

n Number of Bands	$p = n - 1$	Euclidean Space	Hypersphere	$N = \frac{(p-1)(p+2)}{2}$	Remark
3	2	\mathbb{R}^2	\mathbb{S}^1 (circle)	2	Von Mises distribution
4	3	\mathbb{R}^3	\mathbb{S}^2 (sphere)	5	classical Fisher–Bingham “FB5”
5	4	\mathbb{R}^4	\mathbb{S}^3	9	
10	9	\mathbb{R}^9	\mathbb{S}^8	44	
50	49	\mathbb{R}^{49}	\mathbb{S}^{48}	1224	
100	99	\mathbb{R}^{99}	\mathbb{S}^{98}	4949	

References

1. Wyrick, J.; Pasternack, G. Geospatial organization of fluvial landforms in a gravel–cobble river: Beyond the riffle–pool couplet. *Geomorphology* **2014**, *213*, 48–65. [\[CrossRef\]](#)
2. Mahdade, M.; Le Moine, N.; Moussa, R.; Navratil, O.; Ribstein, P. Automatic identification of alternating morphological units in river channels using wavelet analysis and ridge extraction. *Hydrol. Earth Syst. Sci.* **2020**, *24*, 3513–3537. [\[CrossRef\]](#)
3. Kinzel, P.J.; Wright, C.W.; Nelson, J.M.; Burman, A.R. Evaluation of an Experimental LiDAR for Surveying a Shallow, Braided, Sand-Bedded River. *J. Hydraul. Eng.* **2007**, *133*, 838–842.
4. Hilldale, R.C.; Raff, D. Assessing the ability of airborne LiDAR to map river bathymetry. *Earth Surf. Process. Landforms* **2008**, *33*, 773–783.
5. Nakao, L.; Krueger, C.; Bleninger, T. Benchmarking for using an acoustic Doppler current profiler for bathymetric survey. *Environ. Monit. Assess.* **2021**, *193*, 356. [\[CrossRef\]](#) [\[PubMed\]](#)
6. Lyzenga, D.R. Passive remote sensing techniques for mapping water depth and bottom features. *Appl. Opt.* **1978**, *17*, 379–383. [\[CrossRef\]](#) [\[PubMed\]](#)
7. Marcus, W.A.; Fonstad, M.A. Optical remote mapping of rivers at sub-meter resolutions and watershed extents. *Earth Surf. Process. Landforms* **2008**, *33*, 4–24.
8. Legleiter, C.J.; Roberts, D.A.; Lawrence, R.L. Spectrally based remote sensing of river bathymetry. *Earth Surf. Process. Landforms* **2009**, *34*, 1039–1059.
9. Campbell, J.B. *Introduction to Remote Sensing*, 2nd ed.; The Guilford Press: New York, NY, USA, 1996.
10. Philpot, W.D. Bathymetric mapping with passive multispectral imagery. *Appl. Opt.* **1989**, *28*, 1569–1578. [\[CrossRef\]](#)
11. Shah, A.; Deshmukh, B.; Sinha, L. A review of approaches for water depth estimation with multispectral data. *World Water Policy* **2020**, *6*, 152–167.
12. König, M.; Oppelt, N. A linear model to derive melt pond depth from hyperspectral data. *Cryosph. Discuss* **2019**, *2019*, 1–17.
13. König, M.; Birnbaum, G.; Oppelt, N. Mapping the Bathymetry of Melt Ponds on Arctic Sea Ice Using Hyperspectral Imagery. *Remote Sens.* **2020**, *12*, 2623. [\[CrossRef\]](#)
14. Doxani, G.; Papadopoulou, M.; Lafazani, P.; Pikridas, C.; Tsakiri-Strati, M. Shallow-water bathymetry over variable bottom types using multispectral Worldview-2 image. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2012**, *39*, 159–164. [\[CrossRef\]](#)

15. Niroumand-Jadidi, M.; Vitti, A.; Lyzenga, D.R. Multiple Optimal Depth Predictors Analysis (MODPA) for river bathymetry: Findings from spectroradiometry, simulations, and satellite imagery. *Remote Sens. Environ.* **2018**, *218*, 132–147. [CrossRef]
16. Montoliu, R.; Pla, F.; Klaren, A.C. Illumination Intensity, Object Geometry and Highlights Invariance in Multispectral Imaging. In Proceedings of the Second Iberian Conference on Pattern Recognition and Image Analysis—Volume Part I; IbPRIA'05; Springer: Berlin/Heidelberg, Germany, 2005; pp. 36–43. [CrossRef]
17. Kay, S.; Hedley, J.D.; Lavender, S. Sun Glint Correction of High and Low Spatial Resolution Images of Aquatic Scenes: A Review of Methods for Visible and Near-Infrared Wavelengths. *Remote Sens.* **2009**, *1*, 697–730. [CrossRef]
18. Dempster, A.P.; Laird, N.M.; Rubin, D.B. Maximum Likelihood from Incomplete Data via the EM Algorithm. *J. R. Stat. Soc. Ser. B Methodol.* **1977**, *39*, 1–38.
19. Jantzi, H.; Carozza, J.M.; Probst, J.L. Les formes d'érosion en lit mineur rocheux: Typologie, distribution spatiale et implications sur la dynamique du lit. Exemple à partir des seuils rocheux molassiques de la moyenne Garonne toulousaine (Sud-Ouest, France). *Géomorphologie* **2020**, *26*, 79–96.
20. Garambois, P.A.; Biancamaria, S.; Monnier, J.; Roux, H.; Dartus, D. Variational data assimilation of AirSWOT and SWOT data into the 2D shallow water model Dassflow, method and test case on the Garonne river (France). In Proceedings of the 20 Years of Progress in Radar Altimetry, Venice, Italy, 24–29 September 2012.
21. Oubanas, H.; Gejadze, I.; Malaterre, P.O.; Mercier, F. River discharge estimation from synthetic SWOT-type observations using variational data assimilation and the full Saint-Venant hydraulic model. *J. Hydrol.* **2018**, *559*, 638–647. [CrossRef]
22. Goutal, N.; Goeury, C.; Ata, R.; Ricci, S.; Mocayd, N.E.; Rochoux, M.; Oubanas, H.; Gejadze, I.; Malaterre, P.O. Uncertainty Quantification for River Flow Simulation Applied to a Real Test Case: The Garonne Valley. In *Advances in Hydroinformatics*; Gourbesville, P., Cunge, J., Caignaert, G., Eds.; Springer: Singapore, 2018; pp. 169–187.
23. Legleiter, C.; Overstreet, B. Hyperspectral image data and field measurements used for bathymetric mapping of the Snake River in Grand Teton National Park, WY. U.S. Geological Survey Data Release, 2018 (accessed 2021-11-02). [CrossRef]
24. Institut Géographique National. BD ORTHO[®] Version 2.0/ORTHO HR[®] Version 1.0: Descriptif de Contenu, May 2013, Updated July 2018. Available online: https://geoservices.ign.fr/ressources_documentaires/Espace_documentaire/ORTHO_IMAGES/BDORTHO_ORTHOHR/DC_BDORTHO_2-0_ORTHOHR_1-0.pdf (accessed on 14 December 2020).
25. Chandelier, L.; Martinoty, G. Radiometric aerial triangulation for the equalization of digital aerial images and orthoimages. *Photogramm. Eng. Remote Sens.* **2009**, *75*, 193–200. [CrossRef]
26. Kriegler, F.; Malila, W.; Nalepka, R.; Richardson, W. Preprocessing transformations and their effects on multispectral recognition. *Remote Sens. Environ.* **1969**, *6*, 97.
27. Kent, J.T. The Fisher-Bingham Distribution on the Sphere. *J. R. Stat. Soc. Ser. B Methodol.* **1982**, *44*, 71–80. [CrossRef]
28. Kume, A.; Wood, A.T.A. Saddlepoint Approximations for the Bingham and Fisher-Bingham Normalising Constants. *Biometrika* **2005**, *92*, 465–476. [CrossRef]
29. Kume, A.; Preston, S.P.; Wood, A.T.A. Saddlepoint approximations for the normalizing constant of Fisher-Bingham distributions on products of spheres and Stiefel manifolds. *Biometrika* **2013**, *100*, 971–984.
30. Amaral, G.J.A.; Dryden, I.L.; Wood, A.T.A. Pivotal Bootstrap Methods for k-Sample Problems in Directional Statistics and Shape Analysis. *J. Am. Stat. Assoc.* **2007**, *102*, 695–707. [CrossRef]

3.4 Simulation de référence de crue sur la Garonne

Au cours des dernières décennies, l'augmentation des crues fluviales due au changement climatique s'est accompagnée d'une accélération du développement des outils de modélisation. Le défi est alors de trouver un compromis entre le temps, la précision et la robustesse.

Dans cette étude, nous présentons un modèle de cartographie des inondations basé sur la résolution des équations de Saint-Venant en 2D avec un raffinement adaptatif du maillage (Basilisk). Ce type de maillage et l'utilisation de la parallélisation dans Basilisk permettent de produire une simulation à haute résolution (8 m) sur un long tronçon de 40 km de la Garonne, en prenant en compte les affluents. Pour cela, nous développons des techniques dans Basilisk pour gérer les entrées multiples de débits et le calcul parallèle. De plus, pour augmenter la précision des résultats, nous proposons une méthodologie de calage du terme de friction du lit majeur en utilisant des repères de crue. Nous simulons deux événements d'inondation pour notre cas test : l'événement non-débordant de juin 2002 et l'événement débordant de juin 2000. Nous calibrons le terme de friction du lit mineur en utilisant la courbe de tarage aval et la dynamique de l'évènement non-débordant, tandis que le coefficient en lit majeur est calé avec les repères de Plus Hautes Eaux (PHE) de la crue de juin 2000 ainsi que l'hydrogramme aval à Verdun-sur-Garonne.

Cet article est en cours de préparation pour une soumission dans la revue *Geoscientific Model Development*. Les temps de calculs étant assez longs à la résolution maximale (plusieurs jours même en utilisant le maximum de nœuds / cœurs sur le cluster de calcul), le test des valeurs de coefficient de rugosité en lit majeur n'a pas été aussi exhaustif que possible et est à poursuivre (pour des raisons de comparabilité des résultats, nous n'avons pas voulu dégrader la résolution en cours de test).

Le modèle Basilisk a été utilisé dans des études fluviales en injectant des pluies et a montré de bons résultats en termes de précision et de temps de calcul. L'un des objectifs de ce travail a été de développer des outils pour améliorer la cartographie avec Basilisk dans un cas réel d'un grand tronçon avec des injections des débits au niveau de plusieurs affluents. Nous avons également tenté de proposer un cadre rigoureux de comparaison entre données ponctuelles de laisses de crues, et données à la maille de calcul (pouvant comporter une variabilité topographique sous-maille importante).

2D high resolution numerical simulation of historical flood events on the Garonne River, based on adaptive mesh refinement and a specific handling of tributary inlets for parallelization

Mounir Mahdade, Nicolas Le Moine, and Pierre Ribstein

Abstract : In recent decades, the increase in the impact of floods due to both climate and land use changes has been accompanied by an acceleration in the development of modeling tools. The challenge is then to find a compromise between time, accuracy and robustness.

In this study, we present a flood mapping model based on the solving the 2D Saint-Venant Equations with adaptive mesh refinement using Basilisk software. This type of mesh and the use of parallelization in Basilisk reduce the computational time. The objective of this work is to test this model on a long reach of 40km of the Garonne River (France) with several tributaries. For that, we develop some techniques in Basilisk to handle multiple inlets and parallel computing. Furthermore, to increase the results precision, we propose a methodology for calibrating channel and floodplain friction coefficients using flood marks. We simulate two flood events for our test case: the June 2002 non-overflowing event and the June 2000 overflowing event. We calibrate friction coefficient using in situ downstream rating curve for the channel and flood marks provided by a national collaborative platform for the floodplain. The resulting RMSE in flood marks elevation is about 52 cm ; the error on the peak discharge is about 6% and the error in timing at the peak is about 1.5 hours.

1 - Introduction

Floods are among the most devastating recurrent natural disasters. During the 20th century, floods caused approximately 6.8 million deaths between 1980 and 2009 (*Doocy et al.*, 2013; *Kousky*, 2010). The estimate losses will increase from 6 billion dollars per year in 2005 to more than 60 billion dollars per year in 2050 (*Hallegatte et al.*, 2013).

To reduce the human, material and economic impact of floods, scientists, public authorities, and insurance companies are interested in studying this phenomenon to quantify the risks associated with it using numerical hydraulic models. This type of models are developed to understand the processes involved in the flow of rivers and to predict their variables, namely volume, water surface elevation, flood extent and flow velocity. Flood modeling applications require careful attention to the level of accuracy required and the computational efficiency requirements. Some may require fast runtime and real-time flood mapping (e.g. flash floods, *Kirstetter et al.*, 2021). Others may require more accuracy on modeled velocity or maximum flood extent and water depth (for a review, see *Teng et al.*, 2017). These requirements depend on the type of modeling, the model inputs (bathymetry, friction coefficients, and hydrological dataset), and their uncertainties.

There are three most important types of models. Empirical models (e.g., *O'Connor et Costa*, 2004) use statistical tools on observed data such as in situ measurements, surveys, aerial photographs, and satellite images. This type of models is quick and easy to implement, based solely on observations, but challenging to use in scenario modeling for example.

Conceptual models are non-physical models which are based on simplified hydraulic assumptions (e.g., quasi steady state). An example is the Rapid Flood Spreading Method (RFSM, *Gouldby et al.*, 2008; *Wallingford*, 2006) which takes flood volumes as input. Its process is to spread the water over Impact Zones (IZs) in the floodplain. The IZs are topographic depressions in the floodplain where the water accumulates in case of flooding. The output of the model is a flood depth grid (*Lhomme et al.*, 2008). Following the same concept and on a continental scale is the HAND model (Height Above the Nearest Drainage, *Nobre et al.*, 2011). These models require less computational time and effort. They are fast and robust but are not suitable for applications requiring, for example, velocity outputs and accuracy of flow dynamics.

Detailed flood dynamics are simulated using hydrodynamic models. These models solve the Saint-Venant equations (*Saint Venant*, 1871) numerically in 1D, 2D, or Navier-Stokes in 3D and can be linked to hydrological and river models to provide flood maps. The 1D models provide fast real-time simulations for a clear one-dimensional flows (e.g. confined channel, pipe, etc.) but are less suitable for compound bed flows,

where velocities are very different in the channel and the floodplain. 2D and 3D models provide a sufficient representation of flood maps in complex environments, but they are computationally expensive. Comparative studies (*Néelz et al.*, 2010, 2013) have confirmed that 2D models are able to adequately predict variables, including flood extent, velocity, and water surface elevation, on which flood risk management decisions are based (*Teng et al.*, 2017).

However, these flood risk management decisions are effected by several sources of uncertainties (calibration and validation data, hydraulic parameters, topographic data, etc.). At the watershed scale, one of the problems of 2D hydrodynamic models is the lack of accurate data (e.g., continuous bathymetry of the reach). One of the techniques to enrich the watershed dataset is remote sensing. Indeed, both passive and active optical sensors offer the ability to map flooding. On one hand, using near-real-time images of the flood extent, damage can be assessed and interventions can be made in case of risk (*Shen et al.*, 2019a; *Sanyal et Lu*, 2004). On the other hand, multi-spectral images can be used to extract river bathymetry (*Legleiter et al.*, 2009; *Le Moine et Mahdade*, 2021) and thus facilitate flood simulations. Moreover, altimetry data can be used to validate the results of numerical model simulations (*Shen et al.*, 2019b; *Hardesty et al.*, 2018; *Afshari et al.*, 2018).

Furthermore, the computational times related to 2D hydrodynamic models can be extremely important in multi-scenario modeling and in flood impact studies. To reduce this computing time, we can act on the spatial discretization type. In fact, to improve the numerical accuracy and stability, the refinement of the computational mesh is necessary. However, the computation time increases significantly when the number of computational meshes increases. For this, the use of a flexible mesh has been proposed in several studies of river flooding (*Kim et al.*, 2014; *Hoch et al.*, 2018). These models use meshes with triangular cells, quadrilateral cells, and any combination of triangles and quadrilaterals (*Kim et al.*, 2014). This type of mesh has been incorporated into several hydraulic models (MIKE, DELFT3D FM, TUFLOW FV, etc.). The idea is to have finer resolution around complex areas and coarser resolution over large areas with little spatial variance to reduce computational effort (*Teng et al.*, 2017).

With this same goal of reducing the computational time of the simulations, we use the Basilisk model. This model solves the full 2D Saint-Venant equations using adaptive grid refinement (*Popinet*, 2003). This model has been tested in fluvial studies on reaches with rain injections and shows good results in terms of accuracy and computational time. In *Kirstetter et al.* (2021), the model is tested on a flash flood event in an urban setting in a small watershed in the South of France. This model was applied first to test cases where the subcritical, critical, and supercritical flows are well simulated. Second, the model was tested on an experimental model (Toce case study). *Kirstetter et al.* (2021) found that Basilisk correctly models the response using a single upstream inlet with a flood hydrograph, rain input, and with the presence of buildings. Finally, the model is tested on a real case: the flood of October 2015 in Cannes on the French Riviera. Results showed a good agreement between in situ measurements and simulations (*Kirstetter et al.*, 2021, 2020).

The aim of this work is to develop some tools to improve the mapping with Basilisk in a real case with a large river with tributaries discharge injection. For this, we use:

- the adaptive mesh refinement
- continuous high resolution topography
- geometrical and numerical control of multiple inlets with parallel computing

We present in this paper the evaluation of this model on 40km reach of the Garonne river with 5 tributaries. We simulate two historical flood events: the non overflowing event of 2002 and the overflowing flood event of 2000. The section 2 describe the Basilisk model, its features, and the new developments in terms of the control of the inlets and the implementation of source terms. The sections 3 presents the case study of the Garonne river, the calibration of friction terms and the results.

2 - Flood mapping model

The flood model presented in this study solves the 2D Saint-Venant equations using the Basilisk solver. In this section, we present this solver, its specific features, B-flood tool (*Kirstetter et al., 2021*), and the new developments to allow parallel computing in rivers with tributaries.

2.1 - Basilisk solver

Basilisk is an open source software for the solution of the partial differential equations on adaptive meshes. It solves the 2D Saint-Venant Equations (*Saint Venant, 1871*) which is an approximation of the Navier-Stokes equations. These equations represent the water flow under the assumption that the characteristic vertical length of the flow is very small with respect to the characteristic horizontal length.

$$\begin{aligned} \frac{\partial h}{\partial t} + \frac{\partial(hu_x)}{\partial x} + \frac{\partial(hu_y)}{\partial y} &= 0 \\ \frac{\partial(hu_x)}{\partial t} + \frac{\partial}{\partial x} \left(hu_x + \frac{gh^2}{2} \right) + \frac{\partial(hu_x u_y)}{\partial y} &= -gh \frac{\partial z_b}{\partial x} - \frac{1}{\rho} \tau_x \\ \frac{\partial(hu_y)}{\partial t} + \frac{\partial}{\partial y} \left(hu_y + \frac{gh^2}{2} \right) + \frac{\partial(hu_x u_y)}{\partial x} &= -gh \frac{\partial z_b}{\partial y} - \frac{1}{\rho} \tau_y \end{aligned}$$

With $h(x, y, t)$ is the water depth in m, $u_x(x, y, t)$ and $u_y(x, y, t)$ are the two components of the horizontal velocity m/s in a given time t , $z_b(x, y)$ is the bed elevation of the river in m, $\tau_x(x, y, t)$ and $\tau_y(x, y, t)$ are the two components of the bed shear stress in Pa, and ρ is the density of water in kg/m³.

This system can be written in conservative form:

$$\begin{aligned} \frac{\partial}{\partial t} \begin{pmatrix} h \\ hu_x \\ hu_y \end{pmatrix} + \nabla \cdot \begin{pmatrix} hu_x & hu_y \\ hu_x^2 + \frac{1}{2}gh^2 & hu_x u_y \\ hu_x u_y & hu_y^2 + \frac{1}{2}gh^2 \end{pmatrix} &= -gh \begin{pmatrix} 0 \\ \frac{\partial z_b}{\partial x} \\ \frac{\partial z_b}{\partial y} \end{pmatrix} - \frac{1}{\rho} \begin{pmatrix} 0 \\ \tau_x \\ \tau_y \end{pmatrix} \\ \frac{\partial}{\partial t} \mathbf{W} + \nabla \cdot F(\mathbf{W}) &= S_0(\mathbf{W}) + S_f(\mathbf{W}) \end{aligned}$$

Where $\mathbf{W} = \begin{pmatrix} h \\ hu_x \\ hu_y \end{pmatrix}$ is the vector containing the conservative variables, F is the flux, $S_0 = -gh \begin{pmatrix} 0 \\ \frac{\partial z_b}{\partial x} \\ \frac{\partial z_b}{\partial y} \end{pmatrix}$ is

the topographic source term, and $S_f = -\frac{1}{\rho} \begin{pmatrix} 0 \\ \tau_x \\ \tau_y \end{pmatrix}$ the friction source term.

2.1.1 - Numerical scheme

The Saint-Venant solver of Basilisk uses the hydrostatic reconstruction method of *Audusse et al. (2004)*. It preserves ‘‘lake at rest’’ states and ensures the positivity of water depth. The numerical flux is computed using the HLLC (Harten-Lax-van Leer contact) approximate Riemann solver. In the two spatial dimensions, the extension of this scheme is obtained using the MUSCL-type reconstruction combined with a predictor corrector time stepping scheme (*Popinet, 2011*). For more details, the reader can consult the Basilisk website¹ for the source code of the Saint-Venant solver.

The treatment of additional source terms is done with the time splitting technique. In B-flood, which is a subset of Basilisk functions designed to deal with fluvial applications (*Kirstetter et al., 2021*), these terms could be rain, infiltration, and friction terms. In our study, we only use a friction source term as no rain is added to the domain (see section 2.2.2).

¹basilisk.fr

2.1.2 - Adaptive mesh refinement

One of the advantages of Basilisk is the adaptive mesh refinement. The method developed by *Popinet* (2015) is explained in details in *Van Hooft et al.* (2018). It relies on a tree structure (*bitree* in 1D, *quadtree* in 2D and *octree* in 3D): the main idea of the method is that each finite volume cell (which is a square in 2D) may be parent to up to four children (in 2D) if refinement is needed (e.g., presence of water, inlet position, or any point of interest). In contrary, in dry areas or areas of uniform flow variable, cells may be aggregated. The level of a cell is defined by starting from the root cell, which is in the top of the tree and adding one every time a group of four descendant children is added (*Popinet*, 2003).

The function *adapt_wavelet()* refines or enlarge the mesh according to the error estimated by the algorithm between two levels and compared to the error criterion which is set by the user. We can also choose to explicitly maintain the maximum resolution (highest level) at some positions (e.g., tributary inlet sections).

2.1.3 - Inlet with the mask feature

Because of the tree structure used in the definition of grid levels, the overall domain of Basilisk simulations has to be square. In order to handle domains of any shape, parts of the square domain can be deactivated using the *mask()* feature. The cell directly outside the boundary of the domain is called a “ghost” cell where the boundary condition is imposed.

For the sake of clarity, the computation of the flow at an inlet section in the B-flood approach² is first described in 1D in Figure 3.2. As described before, the *mask()* function is a boundary delimiter that creates “ghost” cells. We assume that we have a homogeneous Neumann condition on bed elevation z_b and normal velocity component u_n (here u_x), that is to say, the velocity and bed elevation values are assigned the same value in the “ghost” cell as in the inner cell. The idea is to find the water surface elevation η_{bc} that will achieve the desired discharge through the boundary, with:

$$\eta = z_b + h$$

where h is the water depth and η the water surface elevation.

The *bflux()* function is used to estimate a discharge from a given water surface elevation. It is basically a call to the Riemann solver (*Kurganov et Levy*, 2002) using the defined states on either side of the boundary. However, what is required is the inverse algorithm. It allows to find the value η_{bc} that yield the desired flux, using a false position method. This optimization methodology on the boundary is described in the Figure 3.3.

In 1D, the total flux through the boundary is simply a flux between 2 cells. In the 2D case, each inlet is composed of several boundary cell faces (see Figure 3.4 below), but the procedure is essentially the same: for each trial value of η_{bc} , the total flux through the inlet is computed as a sum of elementary fluxes given by the Riemann solver. Then the same false position method is used to find the value η_{bc} that is uniformly prescribed on all ghost cells associated with the inlet.

While Basilisk allows for parallel computing with MPI (Message Passing Interface³), this specific mask feature is currently not compatible with it. Actually, inflows in the kind of Q_1 in Figure 3.4 are supported, that is, as long as they lie on one of the edges of the square domain. Internal inflows in the kind of Q_2 , however, are not supported together with MPI.

In the rest of this paper we propose developments and improvements that solves this technical issue and increase the computational efficiency, and we test these new enhancements on a long reach of the Garonne River with multiple inlets.

²<http://basilisk.fr/src/discharge.h>

³<https://www.open-mpi.org/software/ompi/v4.0/srpm.php>

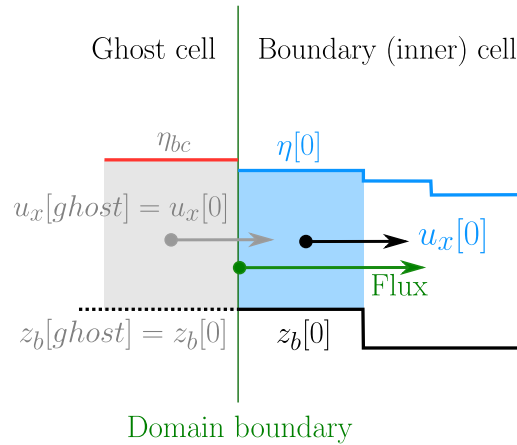


Figure 3.2: A 1D case where the domain boundary is defined using the *mask()* function. The value of velocity and bed elevation in the ‘ghost’ cell on the outer side of the boundary (in gray) are assumed to be the same as in the domain (inner) cell [0] (in blue) with default homogeneous Neumann conditions on z_b and u_x . The only different state variable is depth or, equivalently, water surface elevation $\eta_{bc} \neq \eta[0]$: this value is optimized in order to get the desired flux, as computed by the Riemann solver (see the Figure 3.3).

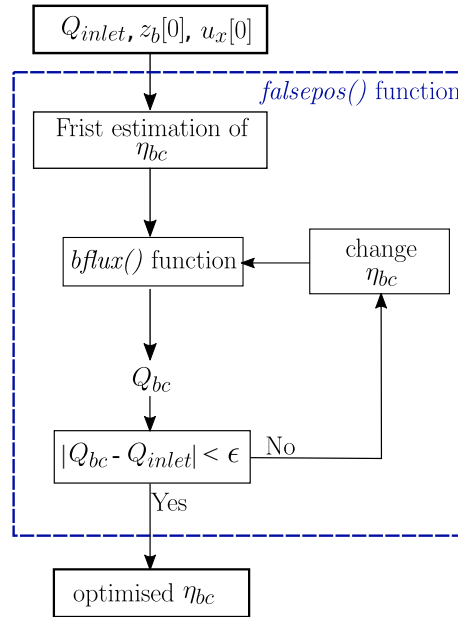


Figure 3.3: Flowchart of the optimization algorithm with *falsepos()* function. The idea is to minimize the error between the target flow and the computed flow with the *bflux()* function. This function optimizes η_{bc} and requires as inputs the target flow Q_{inlet} , the bottom elevation at the upstream position $z_b[0]$ and the velocity at the boundary $u_x[0]$.

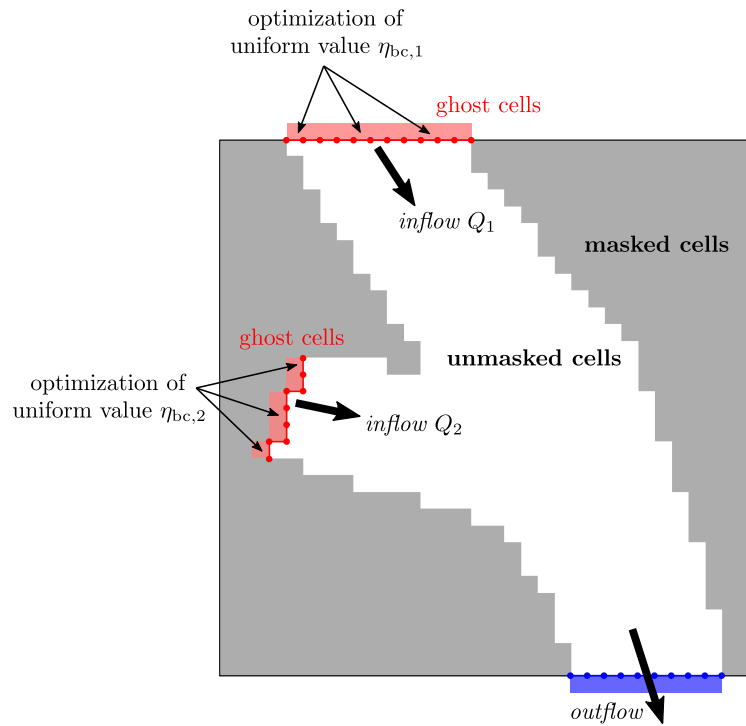


Figure 3.4: Effect of the mask function on a Basilisk square domain in 2D. Gray cells are deactivated; inlet sections can be defined and ghost cells contain the values implementing the desired boundary conditions.

2.2 - New developments in Basilisk

2.2.1 - Defining an inlet without the mask feature

The effect of not using masks on the domain is simple: all cells basically belong to the computational domain, and none of them can be truly “deactivated”. Therefore, cutting away some part of the domain in order to replace it with an inlet section is a bit tricky and needs an ad hoc solution.

The nice property of masks is that they implement a true boundary condition, represented by the values of the different fields (h, u_x, u_y , etc.) in the ghost cells. If we are to implement an inflow routine without ghost cells, we must find a way to force the values of these fields in some parts of the computational domain in order to meet the desired conditions. Contrarily to ghost cells, the area in which we will modify the fields cannot just be a 1-cell-wide buffer, because the cells in this area will exchange fluxes in all directions and not only in the direction of inner, “real” domain cells. The solution we use is illustrated in Figure 3.5:

- first, all cells that we would have liked to mask (deactivate) are set to a high elevation. Thus, there is a kind of “wall” around the computational domain where flooding might occur;
- second, at each inlet section entering the floodplain, an alcove or pool is “carved” into the wall, defining an area in which we can (relatively) safely tweak the values of fields h, u_x , and u_y in order to yield the desired inflow discharge.

For the sake of simplicity, the new procedure is again presented in 1D, and then in 2D.

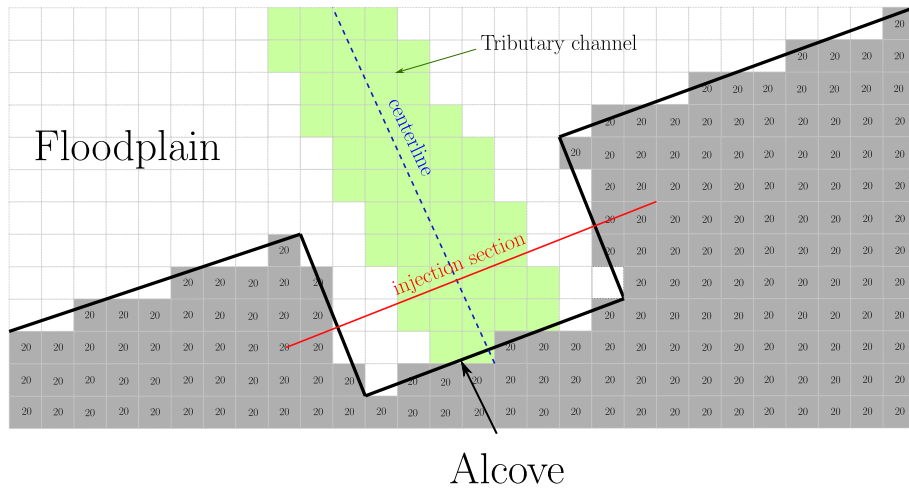


Figure 3.5: An example of the alcove/pool built at the inlet section (red line). The tributary channel is illustrated in green and the center-line is in blue. Cells with white color are the inside domain cells. Grey cells are in the outside of the domain, in this example we assign a high value (20m) of altitude to have a wall at the boundary between the inside domain cells and the outside domain cells.

a - Numerical implementation in 1D case

The numerical implementation of the injections is almost close to the tool presented in Section 2.1.3. However, in our case, we do not have a true boundary condition for the bottom elevation z_b . Moreover, we have normal cells instead of ghost cells at the boundaries. For this purpose, we define a control section (injection section) just in front of the wall (Figure 3.6). The flow injection at this area follows the same routine as in using the two functions $bflux()$ and $falsepos()$. However, a hydrostatic reconstruction of left/right states as in the Saint-Venant solver must be added at the boundary, because bed elevation is not the same on either

side of the boundary and we cannot assume a Neumann condition anymore. For more details, the reader can see the source code in the Appendix (C.4.4).

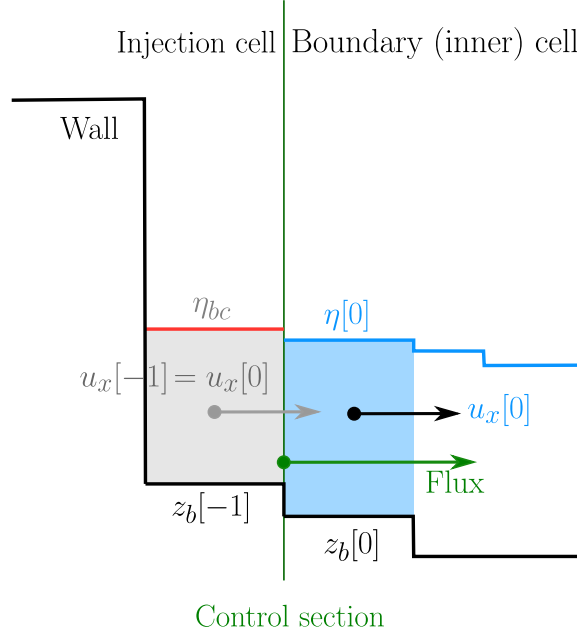


Figure 3.6: A 1D case where the domain boundary is delimited without using the *mask()* function. The gray cell is the pseudo-ghost (outer) cell in which we optimize the value of water surface elevation η_{bc} that will yield the desired flux across the boundary, given the state variables in the boundary (inner) cell. Note that in contrast with Figure 3.2, bed elevation is not the same on either side of the boundary ($z_b[-1] \neq z_b[0]$); hydrostatic reconstruction of left/right states is thus necessary in this case.

b - Numerical implementation in 2D case

In the 2D case, defining inflow sections without the use of the mask feature requires an even greater complexification of the `bflux` approach. Figure 3.7 gives an overview of the rationale used in this case. First, each inlet segment (segment [AB] on the Figure, from left bank to right bank) is rasterized on grid faces (polyline in red on the Figure). With \vec{n}_{AB} the unit normal vector to the segment oriented in the downstream direction, and P a position in the domain, an elementary face is found when the sign of the scalar product $\vec{n}_{AB} \cdot \vec{AP}$ changes from negative to positive between the two cells on either side of the face: any such face defines a pair “pseudo ghost cell” & “domain cell”. Each process / thread shares the full rasterization of each inlet section. Then, we define a rectangular inflow area in which we will allow the routine to set a desired, uniform value of water surface elevation $\eta = \eta_{bc}$ at each time step (the extension of this area is limited to a distance D_{max} upstream of the inlet segment). Just as in the 1D case, the value of η_{bc} is selected so as to yield the target discharge at the current date. For this, for each trial value of η_{bc} we set a loop on the elementary faces that make up the inlet section. The sum of elementary fluxes (black arrows on Figure 3.7) computed with the Riemann solver with this trial state is returned by the `bflux()` function using a reduction clause when running in parallel (each process / thread computes the elementary fluxes across faces located in its computational domain only); then we use the same trial and error procedure. In order for the rasterization to remain unchanged during the simulation, the refinement of the mesh is forced to the maximum in the vicinity of the inlet segment; however, adaptation can occur both downstream of the segment (in the ‘real’ domain) and upstream (in the inlet pond). This is the reason why we must allow an extension D_{max} of the inflow area in the order of a few times the coarser grid size.

In Figure 3.7, the velocity in the inlet pond is shown to be set to zero for each trial value of η_{bc} . Another option is to let a non-zero velocity field develop in the pond, and to use this velocity in the trial and error procedure allowing the identification of η_{bc} . The two options lead to very similar results: in any case, the inflow routine assumes a low Froude regime where fluxes are essentially controlled by pressure gradients (slope of the free surface), and where inertia is negligible.

c – Error correction mechanism

The iterative method presented above generally achieves the desired discharge with an accuracy of about 10%. The residual error is due to the fact that, in the iterative loop where we call the `bflux()` function to compute the discharge for each value of η tested, it does not take into account the effect of the friction term. To correct this residual error, a PID (Proportional Integral Derivative) controller is used at each inlet. The main objective of the controller (Figure 3.8) is to decrease the error to zero. The error is the difference between the actual discharge computed a few meter downstream of the inlet section, and the target discharge. The PID controller modifies the setpoint value according to past errors, using a derivative, a proportional and an integrative component. To put it simply, if the response discharge is too low compared to the target, the setpoint is changed to a value slightly larger than the target; if the response is too large, the setpoint is decreased.

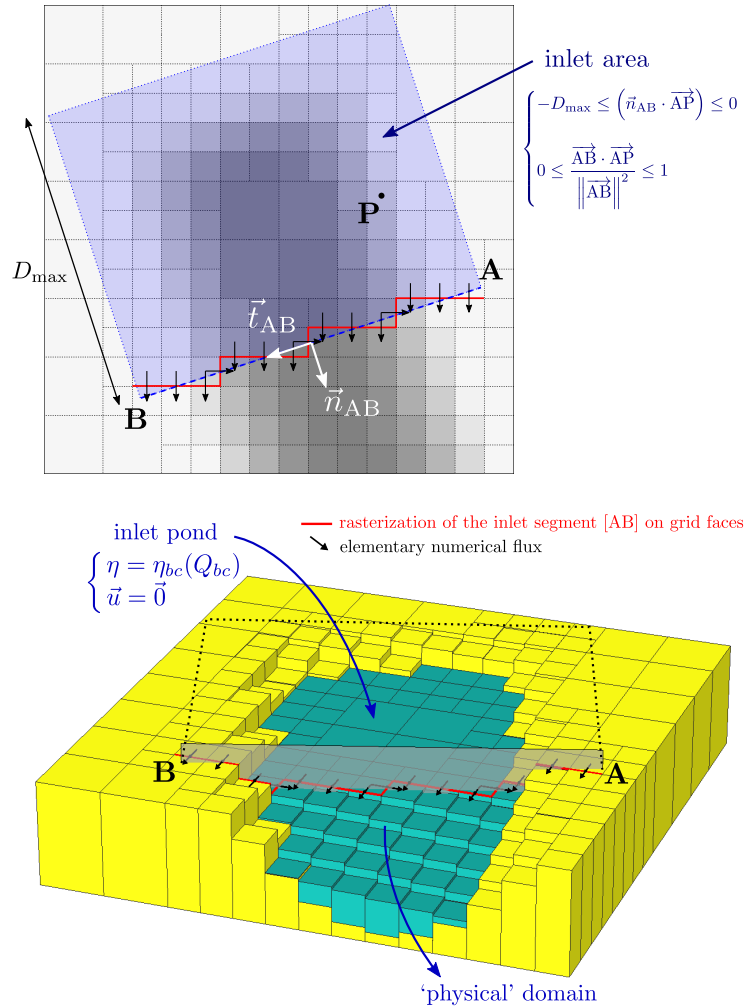


Figure 3.7: Illustration of the inflow routine in the 2D case, across an inlet segment [AB]. The top scheme is a planar view with bed elevation in grayscale, while the bottom scheme is a perspective view. First, the inlet segment is rasterized on boundary faces (red polyline). For each trial value of η_{bc} the water surface elevation upstream of the inlet section (inlet pond), we compute the sum of elementary fluxes given by the Riemann solver for the current state variables (black arrows).

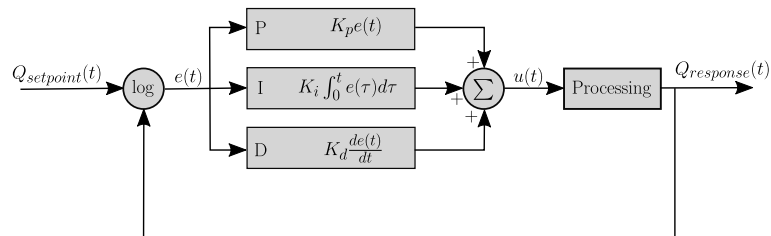


Figure 3.8: Diagram showing the iterative process to compute the targeted discharge by minimizing the error between the setpoint discharge $Q_{setpoint}$ and the response discharge $Q_{response}$ using the formula :

$$e(t) = \log \left(\frac{Q_{response}(t)}{Q_{setpoint}(t)} \right).$$

2.2.2 - Source terms implementation

To guarantee a free flow in the outlet that corresponds to a uniform regime, there are some specific downstream conditions to impose:

1. A Neumann type downstream condition on the bed elevation z_b , the water depth h , and the normal component of the velocity u_n ;
2. A zero Dirichlet condition for the tangential component of the velocity u_t .

It is important to note that the rating curve is not used to specify this downstream condition, which will allow it to be used as validation data.

These downstream conditions of the domain ensure the same value of the variables in the cell just after the boundary. However, the nature of the numerical scheme used by Basilisk (hydrostatic reconstruction, see section 2.2.3) implies that the Neumann condition on the bottom elevation must be homogeneous: the value of the derivative is constant but, in addition, equal to zero. This means that the bottom must be locally flat at the outlet. For this, we use a decomposition of the topography into a regional trend (plane) and a local component z'_b of constant mean:

$$z_b(x, y) = z'_b(x, y) - I_x x - I_y y$$

Where $z'_b(x, y)$ is the detrended bed elevation (i.e. flat on average), and I_x and I_y are the slopes in the x and y directions respectively. In this case, the friction term in the equation becomes:

$$S_0 = -gh \begin{pmatrix} 0 \\ \frac{\partial z'_b}{\partial x} \\ \frac{\partial z'_b}{\partial y} \end{pmatrix} + gh \begin{pmatrix} 0 \\ I_x \\ I_y \end{pmatrix}$$

We see therefore that there is no problem in providing the “detrended” topography z'_b as an input to Basilisk, as long as the regional tilt is added in a second source term that must be provided anyway to take friction into account. This configuration simplifies the initialization of water depths in the studied domain, we can then initialize the simulation with a constant water surface elevation η'_{ini} (as if it were a lake), with zero velocities and thus:

$$h(x, y, t = 0) = \max \{0, \eta'_{ini} - z'_b(x, y)\}$$

In a steady state, the topographic source term balances the friction term and this results in a free surface profile of lake at rest in the “tilted” reference frame.

Thus, we can write:

$$\frac{\partial}{\partial t} \mathbf{W} + \nabla \cdot F(\mathbf{W}) = S'_0(\mathbf{W}) + S_f^*(\mathbf{W})$$

Where

$$S'_0(\mathbf{W}) = -gh \begin{pmatrix} 0 \\ \frac{\partial z'_b}{\partial x} \\ \frac{\partial z'_b}{\partial y} \end{pmatrix}$$

is the topographic source term (taken into account by default in Basilisk) and

$$S_f^*(\mathbf{W}) = gh \begin{pmatrix} 0 \\ I_x \\ I_y \end{pmatrix} - \frac{1}{\rho} \begin{pmatrix} 0 \\ \tau_x \\ \tau_y \end{pmatrix}$$

is now the slope + friction source term, which must be added “manually” in the model.

We use a Manning-Strickler model for the friction term:

$$\boldsymbol{\tau} = \rho g n^2 h^{-\frac{1}{3}} |\mathbf{u}| \mathbf{u}$$

where n is the Manning coefficient and \mathbf{u} the velocity vector. For more details of the implementation of the friction term, the reader can check the source code in the Appendix (C.4.5).

3 - Case study: Garonne River

In this section, we are going to test the new developments on a long reach with tributaries and a complex topography. We chose the Garonne River since this study contributes to the developments of the joint NASA and CNES mission “Surface Water and Ocean Topography” (SWOT). These pre-mission studies aim at testing discharge estimation algorithms using altimetry measurements (*Biancamaria et al., 2016; Durand et al., 2016*). We first present the data used, and then explain the steps of the simulation and the assumptions made.

Our objective is to test the flood mapping model on a real case. For that, we simulate two flood events : the June 2002 and June 2000 events. These simulations require the calibration of two parameters, namely the Manning friction coefficients in the channel, n_{ch} , and in the floodplain, n_{fp} . The coefficient in the channel is first calibrated using the rating curve at the downstream section of the reach, and the propagation dynamics of the non-overflowing flood event of June 2002. Then, the coefficient in the floodplain is calibrated during the major overflowing event of June 2000, using flood marks and the propagation dynamics between the upstream and downstream sections of the 40-km-long reach.

3.1 - Study reach and data

3.1.1 - Study reach and associated subcatchment

The Garonne River is one of the largest rivers of France, flowing from the Pyrénées mountain range down to the Bay of Biscay. Figure 3.9 shows the map of the whole catchment. In this study, we focus on the reach between the cities of Toulouse and Verdun-sur-Garonne. The subcatchment between the junction of the Garonne River and the Ariège River at Portet-sur-Garonne (10 km upstream of Toulouse) is shown in Figure 3.10: the drained area increases from 9980 km² at Portet-sur-Garonne to 13700 km² at Verdun-sur-Garonne.

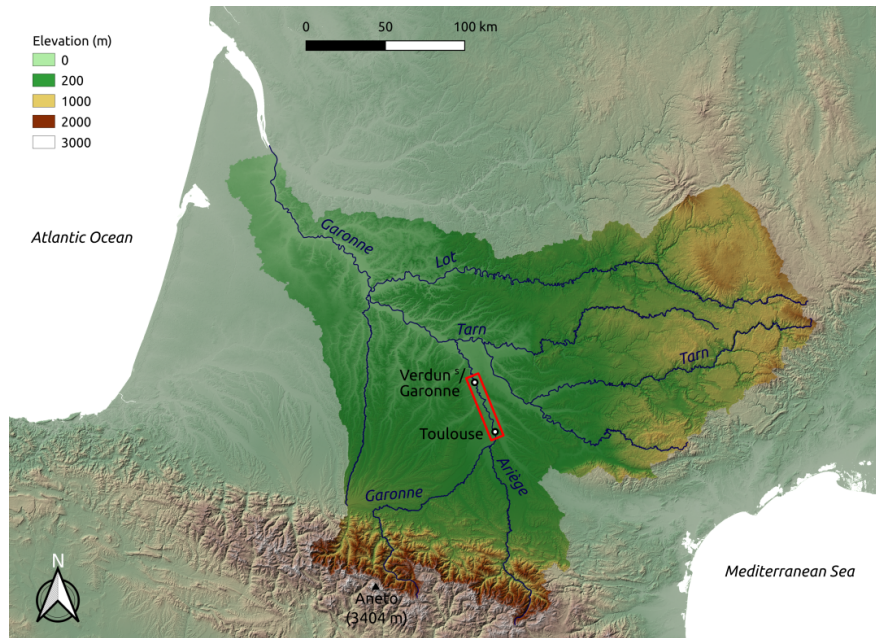


Figure 3.9: Catchment of the Garonne River (55,000 km²).

In order to reduce the overall computational cost, we chose to remove the tributary catchments from the hydraulic domain (hatched in red in Figure 3.10) : the corresponding stream flows are injected at inlet

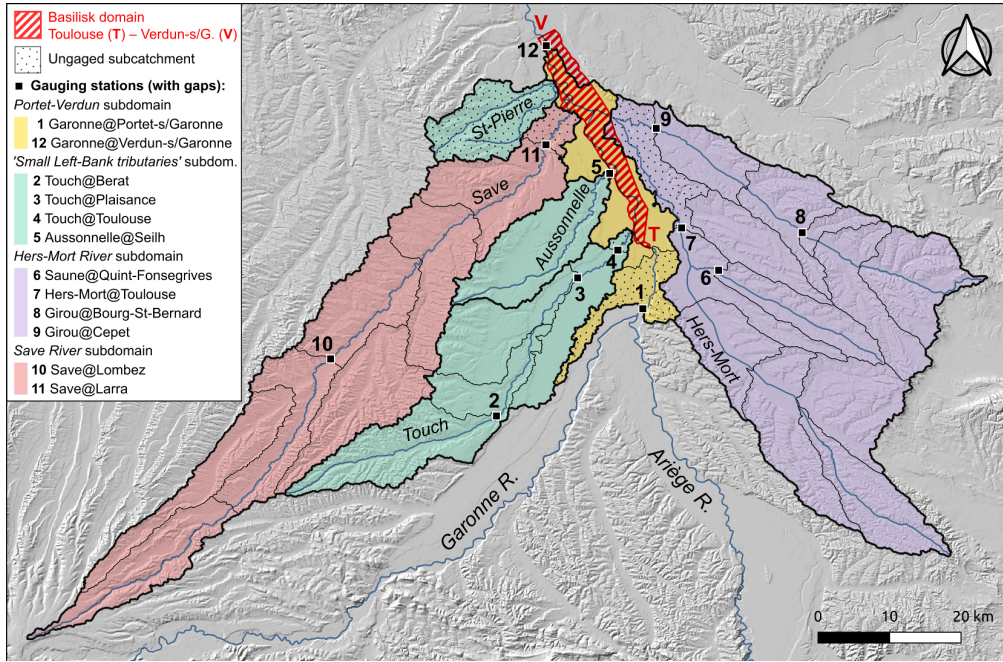


Figure 3.10: Map of the subcatchment of the Garonne River between Portet-sur-Garonne (Garonne–Ariège junction at label 1) and Verdun-sur-Garonne (label 12). Thin black lines show the hydrological units of the hydrologic model (semi-distributed GRSD model, see the section 3.1.2); the same parameterization is used in all units sharing the same color.

sections following the procedure described in section 2.2.3. As a result, no rainfall input is applied on the hydraulic domain.

The first and main inflow is of course the Garonne itself in Toulouse (Pont des Catalans). While the gauging site is a bit upstream at Portet-sur-Garonne, we chose to exclude the reach between Portet and Toulouse from the Basilisk hydraulic domain: there are many hydropower facilities in this reach, and it would be quite tedious to provide and implement stage-discharge relationships for all of them.

We then use five additional inlets for the main tributaries composing the subcatchment between Portet-sur-Garonne and Verdun-sur-Garonne: the Touch, Aussonnelle, Hers-Mort, Save, and Saint-Pierre rivers. The inlets are carefully positioned in narrowings or sections with steep banks, typically at road bridges (see Table 3.1) as close as possible from the edge of the floodplain of the Garonne River.

Inlet	Lat/Lon (WGS84)	Description
Garonne (main)	1.4277 / 43.6034	Pont des Catalans, Toulouse
Touch	1.3969 / 43.6209	Pont d’Ancely, Toulouse–Blagnac
Aussonnelle	1.3555 / 43.6919	D2 bridge, Seilh
Hers-Mort	1.3562 / 43.7670	Aqueduct of the <i>Canal latéral à la Garonne</i>
Save	1.2887 / 43.7707	D29 bridge, Grenade
Saint-Pierre	1.2578 / 43.8074	D26 bridge, Verdun-sur-Garonne

Table 3.1: Location of the six inlets to the Basilisk hydraulic domain.

The characteristics of the tributaries are summarized in Table 3.2, along with gauging information. The subcatchment of interest in this study is the one that contains the main reach (subcatchemnt filled in

Tributary	Upstream area at inlet (km ²)	Downmost gauging site(s)	Internal gauging sites
Touch R.	528	Touch @ Toulouse (515 km ²)	Touch @ Plaisance (415 km ²) Touch @ Bérat (179 km ²)
Aussonnelle R.	192	Aussonnelle @ Seilh (192 km ²)	—
Hers-Mort R.	1524	Girou @ Cepet (526 km ²) Hers-Mort @ Toulouse (768 km ²)	Girou @ Bourg-St-B. (310 km ²) Saune @ Quint-Fonsegr. (106 km ²)
Save R.	1146	Save @ Larra (1110 km ²)	Save @ Lombez (424 km ²)
Saint-Pierre R.	135	—	—
TOTAL	3525	3111 km ² (88%)	

Table 3.2: Available streamflow data for tributaries. Time series are neither free of gaps nor fully overlapping.

yellow in Figure 3.10, with the Basilisk domain hatched in red). The other areas are used in the hydrological model for the elaboration of the data injected at the inlets listed in Table 3.1. This part is detailed in next subsection (3.1.2). The handling and the elaboration of the channel and floodplain topography in the Basilisk domain are presented in the subsection 3.1.3.

3.1.2 - Preparation of forcing inflow discharge data

It is necessary to have continuous hydrographs at each inlet since the simulation of the two events is done in transient regime. Some studies directly use available streamflow measurements as inflows (e.g., *Latapie et al.*, 2012). In this study, we chose to calibrate the GRSD hydrological model (see e.g., *de Lavenne et al.*, 2019) against these measurements in order to produce homogenized inflow series. This hydrological modeling step is desirable for several reasons:

- the tributary catchments are not completely gauged, as illustrated with dotted areas in Figure 3.10. It is therefore helpful to model the effect of the ungauged reaches, not so much to correct the volume of inflows but to correct their timing (at first order) from the downmost gaging site(s) in each tributary catchment,
- actually, all streamflow series are not free of gaps and do not fully overlap; if we want to be able to simulate any flood event in the domain (historical or under future climate), we need to complete the streamflow series and to be able to produce new ones.

We use a hourly-time-step version of the GRSD model to produce these homogenized inflow series using meteorological forcing data. The original routing is slightly modified, we use Hayami routing kernels with a design similar to the MORDOR-TS model (*Rouhier et al.*, 2017).

Precipitation and temperature data were extracted from the SAFRAN dataset (*Vidal et al.*, 2010) at the hourly time step on the period 1999–2019. Temperature data was used to compute potential evapotranspiration using *Oudin et al.* (2005) formula. A second precipitation dataset was available on the 1997–2015 period, namely the COMEPHORE dataset (*Tabary et al.*, 2012). As this second precipitation dataset turned out to be slightly more satisfying than the SAFRAN precipitation data on the overlapping period, inflow series are currently produced with COMEPHORE precipitation data and SAFRAN temperature data on the period 1999–2015. This period contains two major flood events of the Garonne, the June 2000 flood and the January 2014 flood.

For the sake of brevity, the calibration procedure of the GRSD model is presented in appendix A of the paper. In Figure 3.11 we display the observed and simulated hydrographs at Portet-sur-Garonne and Verdun-sur-Garonne, as well as the cumulative volumes for the two largest flood events of the period. There is a good agreement on total volume for the June 2000 event (5.7% error), and a substantially higher error

for the January 2014 event (14%).

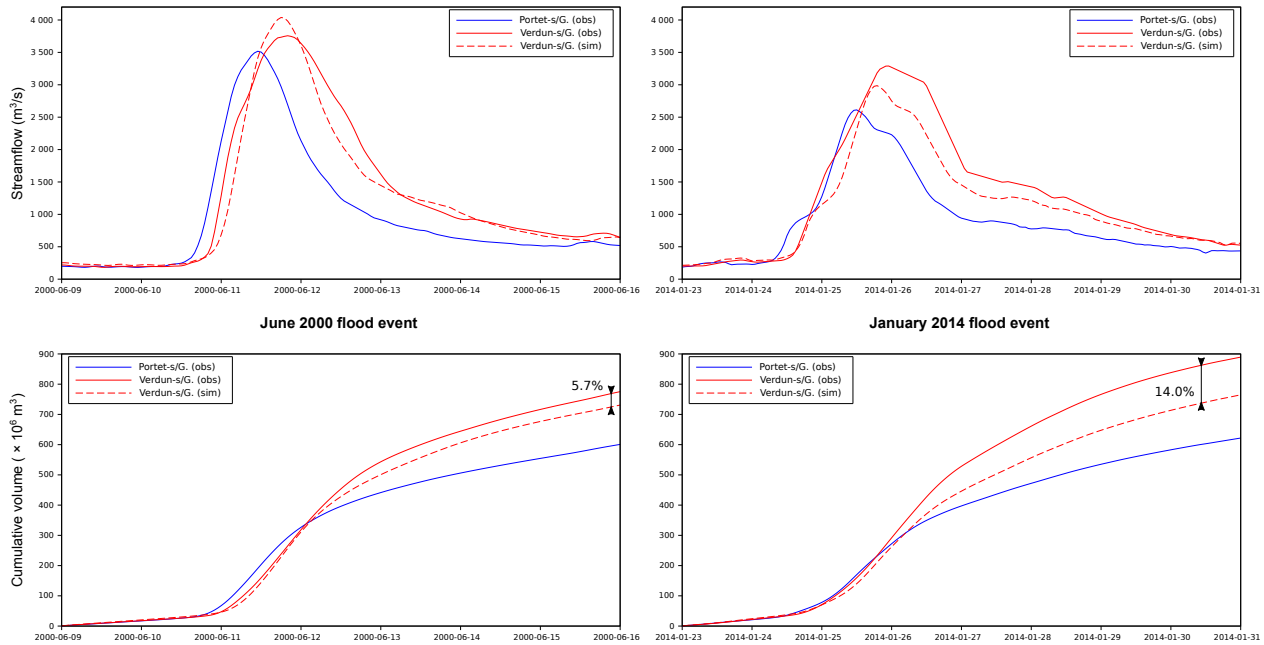


Figure 3.11: Observed hydrographs at Portet-sur-Garonne (solid blue) and Verdun-sur-Garonne (solid red) for the two largest flood events of the period, along with the hydrograph at Verdun simulated by the GRSD model (dashed red). As the propagation on the reach will be performed more precisely by Basilisk, it is interesting to focus on the cumulative volume (graphs at the bottom). The overall event volume is correctly simulated for the June 2000 event (error of 5.7%), while the discrepancy is larger for the January 2014 flood (14.0%).

We extract the calibrated hydrographs of the June 2000 flood event (Figure 3.12) from 2000-06-08 00:00 UTC to 2000-06-17 00:00 and June 2002 flood event (Figure 3.13) from 2002-06-03 00:00 UTC to 2002-06-17 00:00 UTC. These hydrographs are the forcing of our hydraulic model (Basilisk) in order to simulate the flood events.

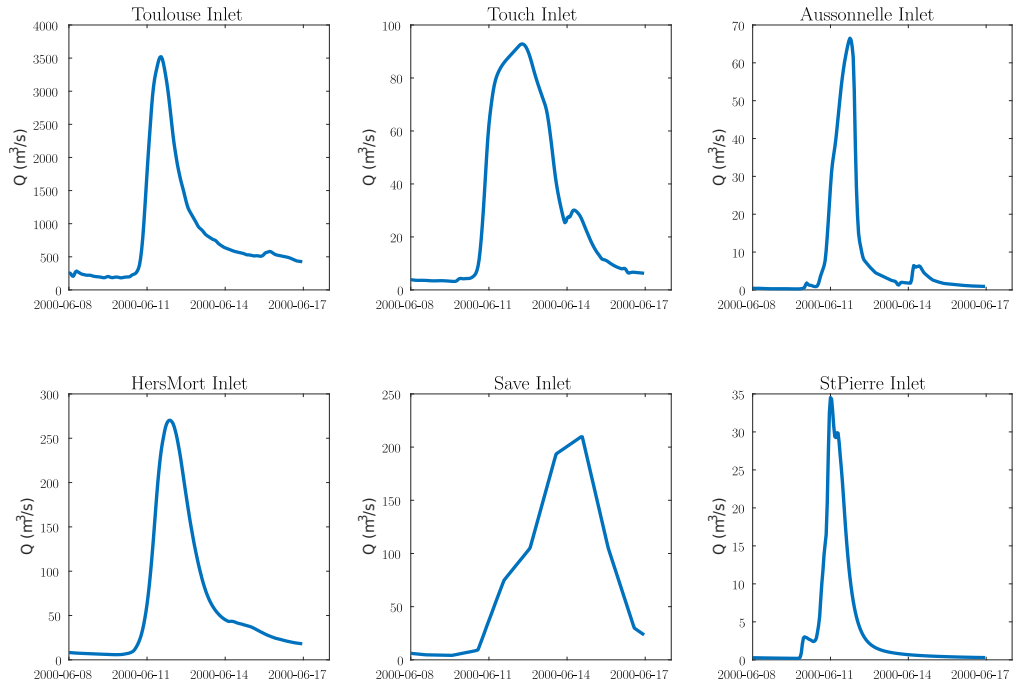


Figure 3.12: Injected hydrographs of the June 2000 event for the six inlets. These hydrographs are simulated and calibrated using the GRSD hydrological model.

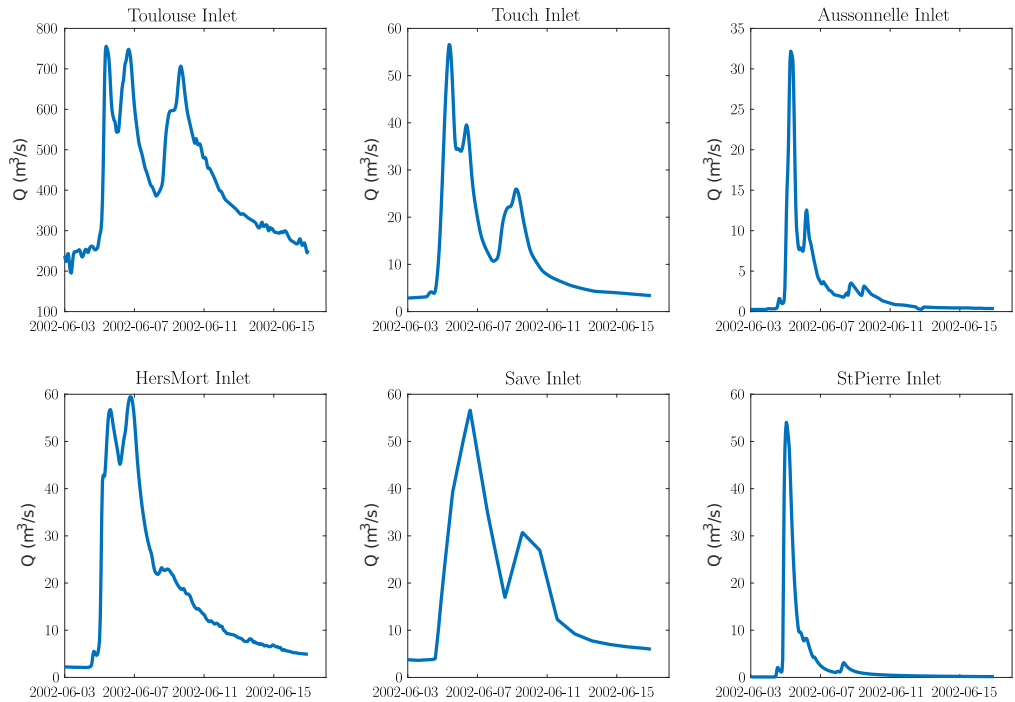


Figure 3.13: Injected hydrographs of the June 2002 event for the six inlets. These hydrographs are simulated and calibrated using the GRSD hydrological model.

3.1.3 - Channel and floodplain topographic data in the Basilisk domain

For this study, we need a continuous topography of both channel and floodplain. The estimation of channel bathymetry in this reach of the Garonne River has been the subject of a specific publication (*Le Moine et Mahdade, 2021*). We developed a method based on the multispectral analysis of airborne visible (RGB) and near-infrared (IRC) images, the BD ORTHO database provided by the IGN. This high-resolution (0.5 m) dataset (*Chandelier et al., 2009; IGN, 2018*) allowed us to define a relationship between depth and a spectral invariant independent of lighting conditions, named multi-spectral hue. This hue–depth relationship was calibrated against *in situ* depth measurements at 120 surveyed cross-sections in the channel, yielding a RMSE of about 55 cm on spatially-distributed depth.

This model is applied along the reach and blended with the DEM of floodplain area to create the topography in Figure 3.14-a. This 1m high resolution DEM is the topography of the reach used in this study to simulate the flood events.

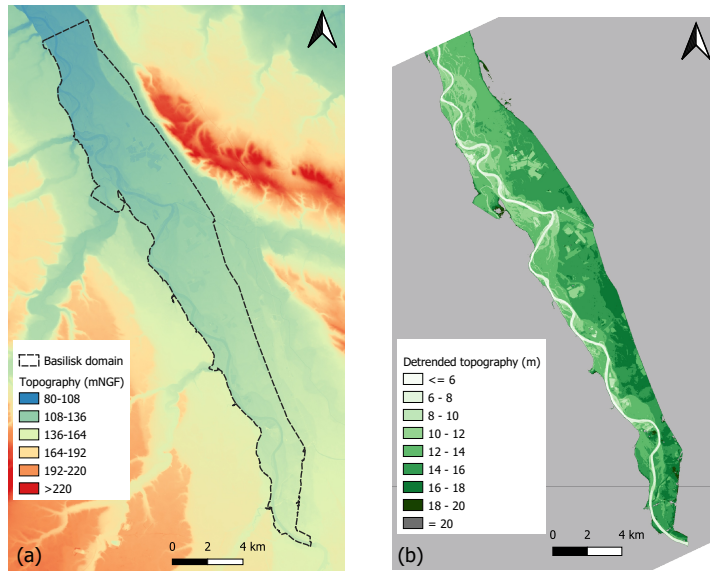


Figure 3.14: (a) Topography $z_b(x, y)$ of the studied reach with the Basilisk hydraulic domain in mNGF; (b) detrended topography $z'_b(x, y)$ after removal of the regional linear (plane) component.

We recall that the application of the new developments in Basilisk presented in section 2 (parallelized inflow routine) requires some preconditioning of topographic data:

- first, we decompose the floodplain topography into a regional trend (plane) and a local fluctuation z'_b of constant mean. The numerical parameters of this decomposition are

$$z'_b(x_{L93}, y_{L93}) = z_b(x_{L93}, y_{L93}) + I_x \times (x_{L93} - x_{c0}) + I_y \times (y_{L93} - y_{c0}) - 100$$

where $(x_{c0}, y_{c0}) = (565\,000, 6\,294\,000)$ is a point of round coordinates (in the French legal projection system Lambert 93) located in the middle of the domain, $I_x = -4.88 \times 10^{-4}$ m/m the tilt in the X-direction (i.e., the average floodplain topography decreases towards the West), and $I_y = +9.86 \times 10^{-4}$ m/m the tilt in the Y-direction (i.e., the average floodplain topography decreases towards the North);

- In the detrended topography z'_b (Figure 3.14-b), the maximum level of the historical June 2000 flood event is about 14 to 15 meters. Hence, we floor the detrended elevation data to a value of 20 meters.

Figure 3.15 then illustrates the rationale for positioning an inlet with a small pool carved into terrain, as described in the section 2.2. This procedure is applied to all inlets of the study. In Figure 3.14-b, we also show the position of the downstream (outflow) section of the reach at Verdun-sur-Garonne, which is perpendicular to the regional floodplain direction (a final rotation is performed in Basilisk in order to align this outflow section on the top boundary of the computational domain). At this section, a free uniform outflow boundary condition is prescribed in Basilisk (Neumann condition on z'_b , h and u_n the normal velocity component, and Dirichlet homogeneous condition $u_t = 0$ on transverse component).

Finally, the detrended topography with conditioned inlet pools is provided to Basilisk at the maximum resolution of 8m.

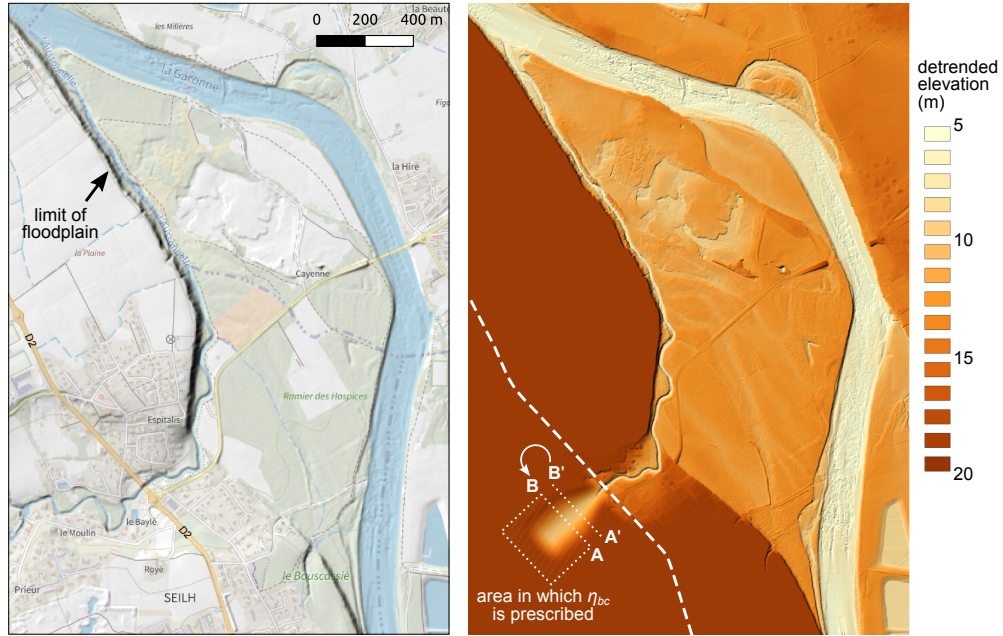


Figure 3.15: Detail on the Aussonnelle River inlet at Seilh, under the bridge of D2 road. We clearly see the transition between the floodplain and the higher neighbouring terrain: the detrended elevation of this higher terrain is capped at 20 m, and a small inflow pool is carved in it. The hatched area upstream of section A–B are the Basilisk cells in which we specify the value of η_{bc} that will yield the desired discharge. Just a few meters downstream, at section A’–B’, we perform a virtual discharge ‘measurement’ that will feed the PID regulator.

3.2 - Calibration of the channel friction term

In this section, we present methodology and results of calibration of channel Manning coefficient n_{ch} . This calibration is performed on two aspects of the hydraulic behavior:

- First, we perform a set of steady-state simulations with non-overflowing discharges; we then compare the values (η, Q) simulated at the hydrometric station of Verdun-sur-Garonne, with the portion of the rating curve below the bankfull level. We recall that a free uniform outflow boundary condition is used at the outlet of the reach (not a stage-discharge relationship) and that, consequently, the rating curve at Verdun-sur-Garonne is available for use as validation data;
- Second, we perform a transient simulation on the June 2002 flood event, a non-overflowing event with

a peak discharge of about $1000 \text{ m}^3/\text{s}$. We then compare the simulated vs. observed propagation time between the inflow hydrograph at Toulouse and the outflow hydrograph at Verdun-sur-Garonne.

3.2.1 - First estimate of channel roughness using downstream rating curve in steady state

For the first estimate of the friction coefficient in the channel, we use the available rating curve at the Verdun-sur-Garonne station (Figure 3.16). This rating curve is a high water rating curve between 1999 and 2015 extracted from the ‘‘Hydroportail’’ platform (www.hydro.eaufrance.fr). We use only the discharges below the bankfull level which is estimated at $\sim 1000 \text{ m}^3/\text{s}$.

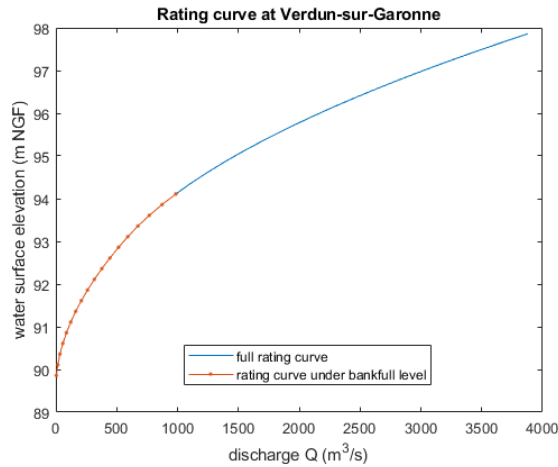


Figure 3.16: Rating curve at Verdun-sur-Garonne station. Points in orange color represent the rating curve below the bankfull level.

Since the calibration is done in steady state, it is necessary to have only a range of discharges to be injected into each tributary and not hydrographs. We inject discharges that are lower than the bankfull discharge (estimated $\sim 1000 \text{ m}^3/\text{s}$).

A simple calculation is used to find the percentage of discharge to be injected at each inlet in relation to the desired downstream discharge at Verdun-sur-Garonne. A linear regression is performed of the daily discharges between the Portet-sur-Garonne station and the Verdun-sur-Garonne station between 1992-08-01 and 2008-01-27. We consider that there is no difference in discharges between the Toulouse inlet section (Pont des Catalans) and the Portet-sur-Garonne station. Only flows below $1000 \text{ m}^3/\text{s}$ were selected and results are depicted in the Figure 3.17.

In the following, we consider that :

$$Q_{jPortet} \approx 0.925 \times Q_{jVerdun}$$

Then, the flow injected at the main inlet is 92.5% of the flow at Verdun-sur-Garonne. That is to say that the volume of water arriving downstream comes mainly from upstream (92.5%). The rest (7.5%) comes from the tributaries. It is clear that it is small compared to the amount of water coming from the main channel but these contributions should not be neglected. The idea is to allocate this volume (7.5%) of water to the tributaries in proportion to their watershed areas. The Table 3.3 shows for each tributary the percentage of flow to be injected relative to the desired flow.

Moreover, injected discharges are determined according to the rating curve in the Figure 3.16 (downstream discharges). We choose to inject 9 discharges (30, 60, 90, 180, 290, 420, 580, 760, and $1000 \text{ m}^3/\text{s}$) and we test several Manning coefficients ranging between $0.029 \text{ s} \cdot \text{m}^{-1/3}$ and $0.045 \text{ s} \cdot \text{m}^{-1/3}$ using the setup presented

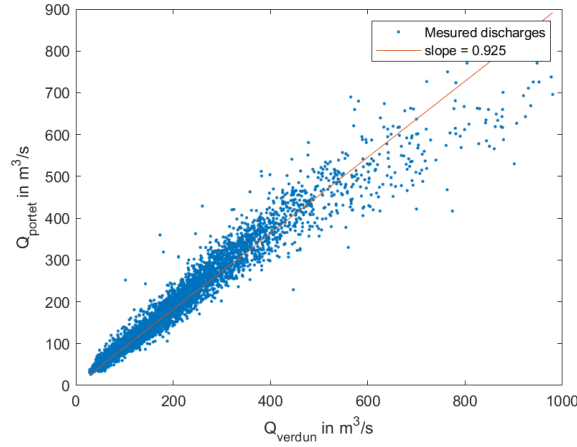


Figure 3.17: Linear regression results between discharges at Verdun-sur-Garonne and Portet-sur-Garonne stations. We find that $Qj_{Portet} \approx 0.925 \times Qj_{Verdun}$

Tributary	Upstream area at inlet (km ²)	Watershed percentage from the total area	Injected flow percentage
Touch R.	528	15%	1.1%
Aussonnelle R.	192	5%	0.4%
Hers-Mort R.	1524	43%	3.2%
Save R.	1146	33%	2.5%
Saint-Pierre R.	135	4%	0.3%

Table 3.3: Distribution of the flows injected into the tributaries according to their areas.

in Table 3.4. We use an adaptive mesh refinement with a tolerance value of 1 cm on the water depth and 0.01 m/s on the velocity. We apply refinement also on the inlets because they are sensitive and we should have accurate computations in these areas without slowing down our simulation. The simulated duration for each discharge value is 1 day (86400 s) in order reach a steady state regime and capture the discharge and the water surface elevation at the outlet of the reach.

Min resolution (m)	Max resolution (m)	Duration (hours)	Tolerance on water depth (cm)	Tolerance on velocity (m/s)
512	8	24	1	0.01

Table 3.4: Simulations setup for the calibration of the channel Manning coefficient. These parameters are related to the minimal and maximal resolution and the criteria of the adaptive mesh refinement.

At the inlet level, the adjustment of the three K for the PID controller is done after several tests. Therefore, we choose the most adequate values for our simulations ($K_p = 0.1$, $K_i = 30$, and $K_d = 5$). The discharge just below each inlet section is checked using the *segment_flux()* function (see Figure 3.15). An example of the 180 m³/s targeted discharge at Verdun-sur-Garonne is illustrated in the Figure 3.18. At each inlet, the achieved discharge was compared to the injected discharge and we find that the discharge fluctuates a little before reaching the target discharge in half an hour and remains stable afterwards. This amount of time is a “warm-up” period during which the simulated values should be ignored. Moreover, in the downstream discharge and water surface elevations Figures (Figure 3.18), a stable steady state regime is reached in ~ 13 hours.

Figure 3.19 shows results of the calibration of Manning coefficient in the channel. It represents the water surface elevations (η) as a function of injected discharge (Q). We calculate the RMSE between the simulated and observed rating curves (Table 3.5) using the formula:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (\eta_{sim}^i - \eta_{obs}^i)^2}$$

Where η_{sim}^i is the simulated water surface elevation for a given discharge and a Manning coefficient n_{ch} , η_{obs}^i is its related observed water surface elevation, and N is the number of discharges (in our case $N = 9$).

n_{ch} ($s \cdot m^{-1/3}$)	0.029	0.030	0.033	0.040	0.042	0.045
RMSE (m)	0.421	0.363	0.352	0.075	0.083	0.180

Table 3.5: Simulations RMSE results for the Manning coefficient calibration

Results show that the rating curve with a Manning coefficient of $n_{ch} = 0.040 s \cdot m^{-1/3}$ overlaps best with the observed rating curve and has the lowest RMSE = 0.075 m compared to all Manning coefficients. Therefore, we will suppose in what follows that the Manning coefficient of the channel equals to $0.040 s \cdot m^{-1/3}$.

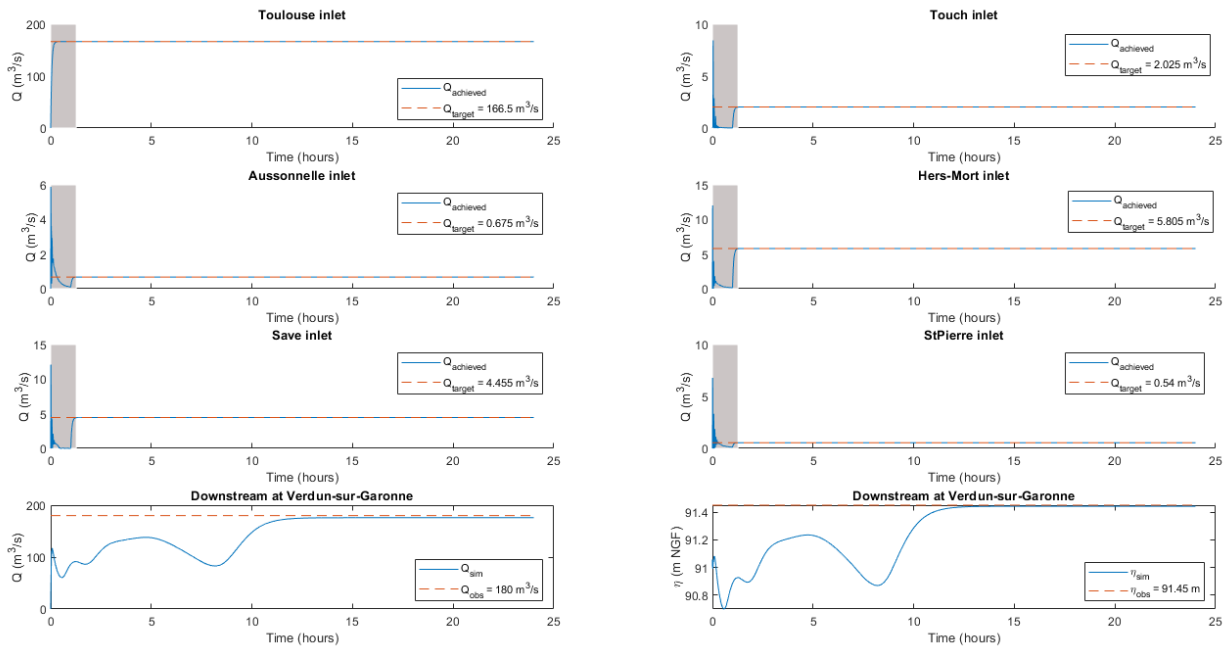


Figure 3.18: An example of the the $180 m^3/s$ targeted discharge at Verdun-sur-Garonne. The first six Figures illustrate the achieved discharges at the six inlets in blue color compared to the targeted discharges in dashed red lines (calculated from the percentages presented in the Table 3.3). In the two Figures below, we represent the discharge and the water surface elevation (m NGF) at Verdun-sur-Garonne. The discharge and water surface elevation reached a stable level and a steady state at 13 hours of simulation. The grey area in the Figures shows a “warm-up” period during which the simulated values should be ignored. These oscillations are made by the PID controller to reach the desired flow in a non-brutal way.

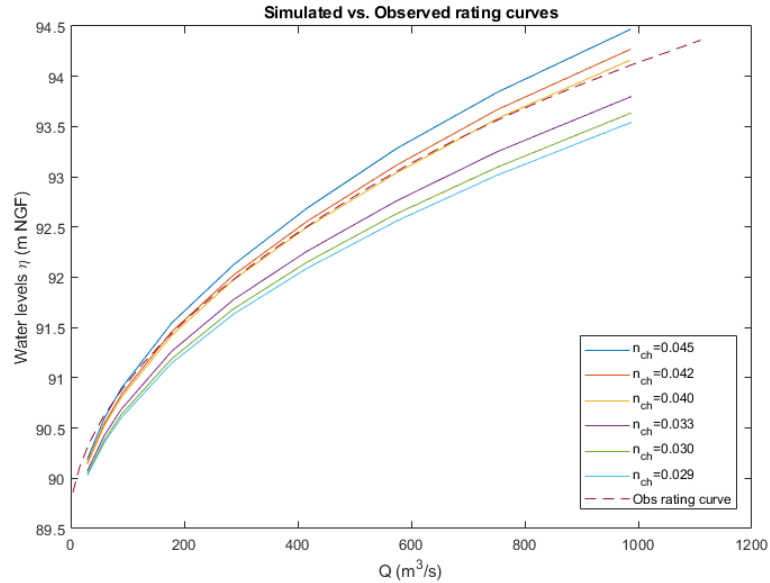


Figure 3.19: Results of calibration of the channel Manning coefficient. Each simulated rating curve corresponds to a certain Manning coefficient. The dashed line curve is the observed curve rating.

3.2.2 - Comparison / validation on the June 2002 non-overflowing flood event

The 2002 flood event on the Garonne River is a non-overflowing event recorded between June 03, 2002 and June 17, 2002 (Figure 3.20) and reaches three peaks consecutively in the dates 2002-06-04, 2002-06-05, and 2002-06-08. The purpose of this simulation is to see how well our model could simulate this non overflowing event with all these 3 peaks and also to evaluate the choice of the channel Manning coefficient.

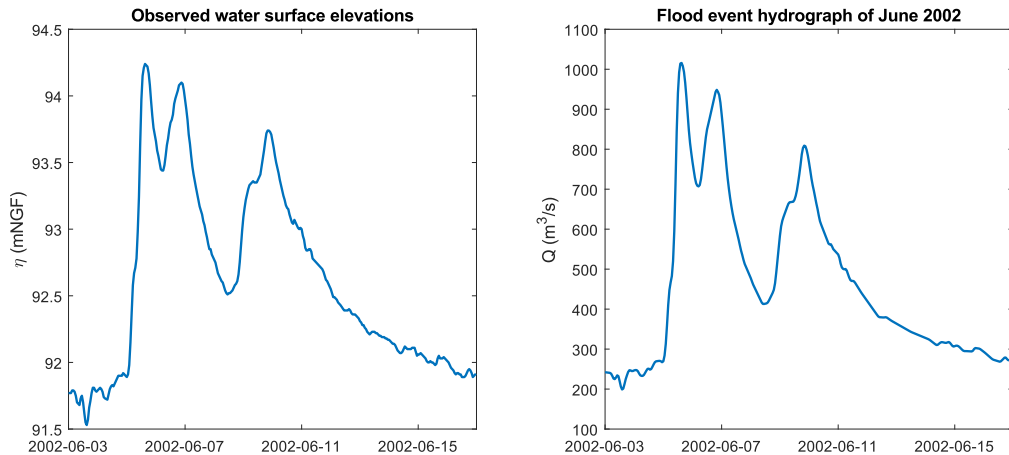


Figure 3.20: Flood event hydrograph of June 2002 from 2002-06-03 00:00 UTC to 2002-06-17 00:00 UTC (right) and the corresponding water surface elevations η (m NGF). They are measured in Verdun-sur-Garonne station and provided by “Hydroportail” platform (www.hydro.eaufrance.fr).

To do so, we use the same topography presented in section 3.1.3. However, here we are going to inject

flood hydrographs instead of discharges since the simulation is in the transient regime. As mentioned earlier these homogenized inflow hydrographs are produced by the GRSD model (see Appendix A for more details) and they are represented in Figure 3.13. The injection is done in the same way with a PID controller with parameters ($K_p = 0.1$, $K_i = 30$, and $K_d = 5$) for each inlet. The simulation is done using the channel Manning coefficient found in the calibration part: $n_{ch} = 0.040 \text{ s} \cdot \text{m}^{-1/3}$.

We use an adaptive refinement with a threshold value of 1 cm on the water depth and 1 cm on the water surface elevation. We apply refinement on the inlets as in the precedent simulations. The simulation covers the 14-day period between 2002-06-03 00:00 UTC and 2002-06-17 00:00 UTC (Table 3.6). The simulation was performed with parallel computing on 1 node with 48 processors in the MESOcentre cluster (Climserv-IPSL).

Min resolution (m)	Max resolution (m)	Duration (days)	Tolerance on water depth (cm)	Tolerance on water surface elevation (cm)
512	8	14	1	1

Table 3.6: Simulations setup for the calibration of the Manning coefficient in the channel. These parameters are related to the minimal and maximal resolution and criteria of the adaptive mesh refinement. The duration of the simulation is 14 days between 2002-06-03 00:00 UTC and 2002-06-17 00:00 UTC.

The inflow routines gives satisfactory results. For example, at the Toulouse inlet in the Figure 3.21, we can see that the error between the target discharge and the achieved discharge using the PID controller is very low (we neglect the “warm-up” part, the grey area of the Figure 3.21).

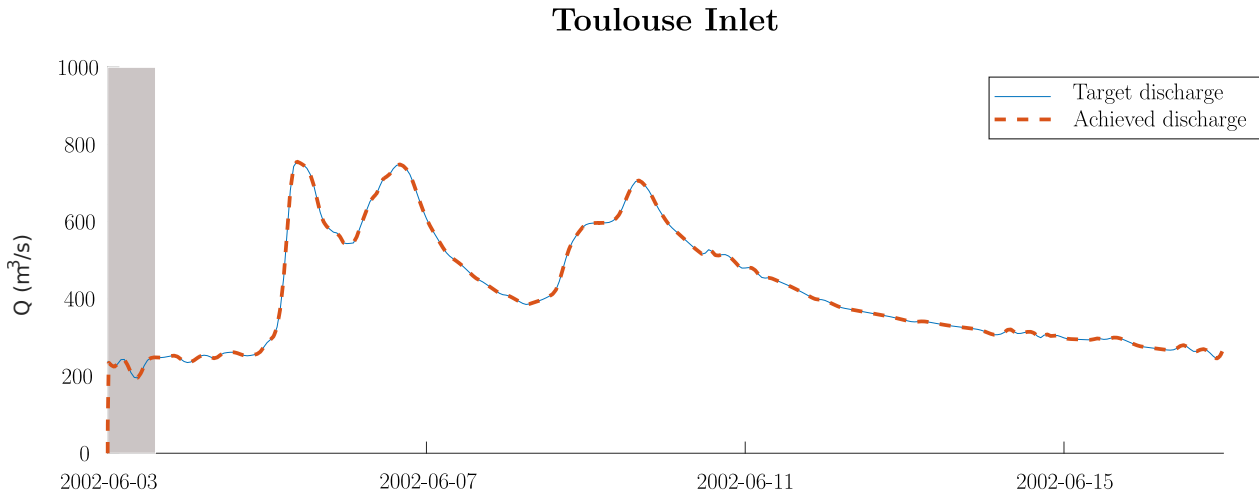
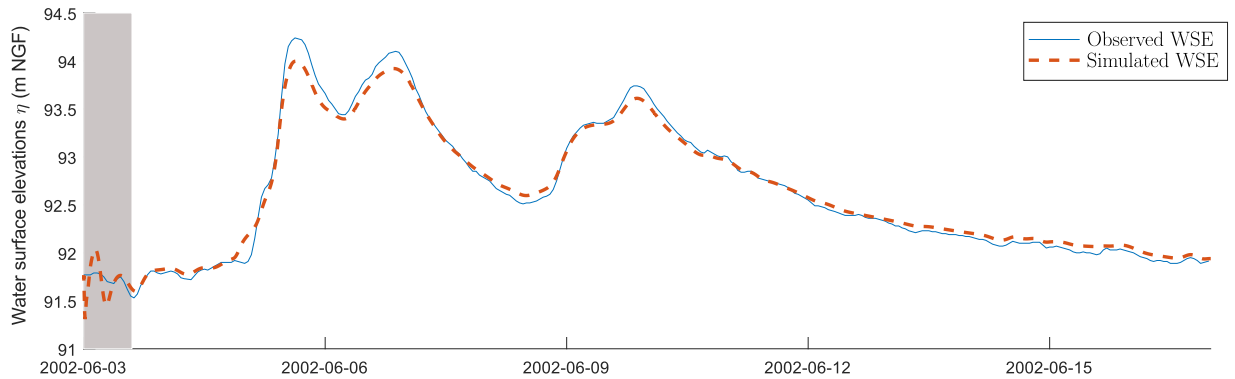


Figure 3.21: Achieved vs. target hydrograph at the Tououse inlet. The grey area in the figures shows a “warm-up” period during which the simulated values should be ignored.

We then extract the simulated discharge and water surface elevation (η) at the downstream of the reach at Verdun-sur-Garonne. The results in Figure 3.22 show that the error between the simulated discharge and the observed discharge is very small ($\text{RMSE}(Q) = 30.7 \text{ m}^3/\text{s}$ and $\text{RMSE}(\eta) = 0.081 \text{ m}$). We also notice that the peaks are well modeled with an average error of $\sim 0.20\text{m}$ for the water surface elevation and $\sim 70\text{m}^3/\text{s}$ for the discharge.

The flood propagation time is calculated at the 3 different peaks between the main inlet (Toulouse inlet) and the downstream outlet at Verdun-sur-Garonne (Figure 3.23). We found an error in timing at the first peak of 13min, for the second peak of 23min, and for the third peak of 52min. Thus, we find an average error in timing of about 30min which is acceptable.

Simulated vs. Observed downstream water surface elevations of June 2002



Simulated vs. Observed downstream hydrograph of June 2002

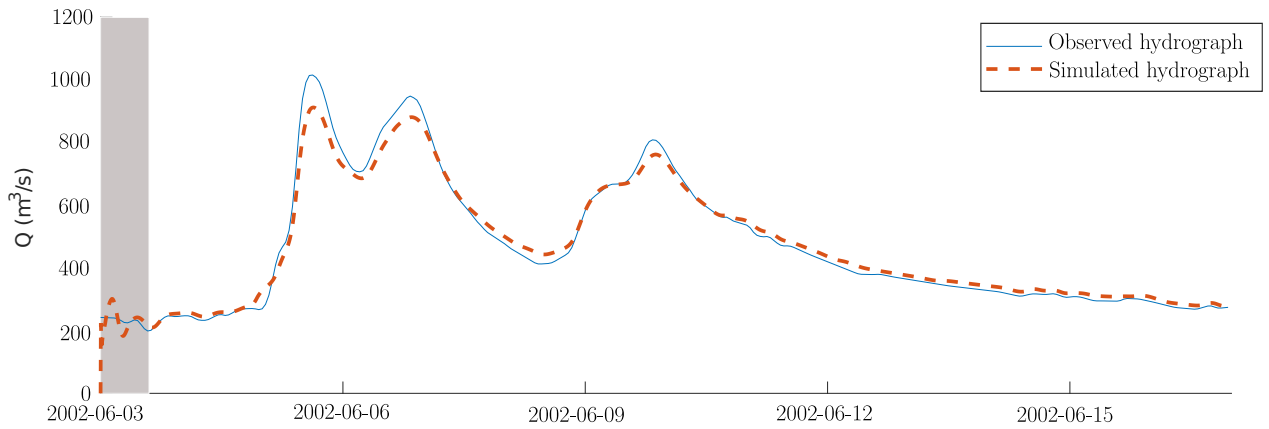


Figure 3.22: Simulated vs. observed water surface elevation (top) and hydrograph (down) at Verdun-sur-Garonne. The grey area in the Figures shows a “warm-up” period during which the simulated values should be ignored.

Propagation time of flood peaks

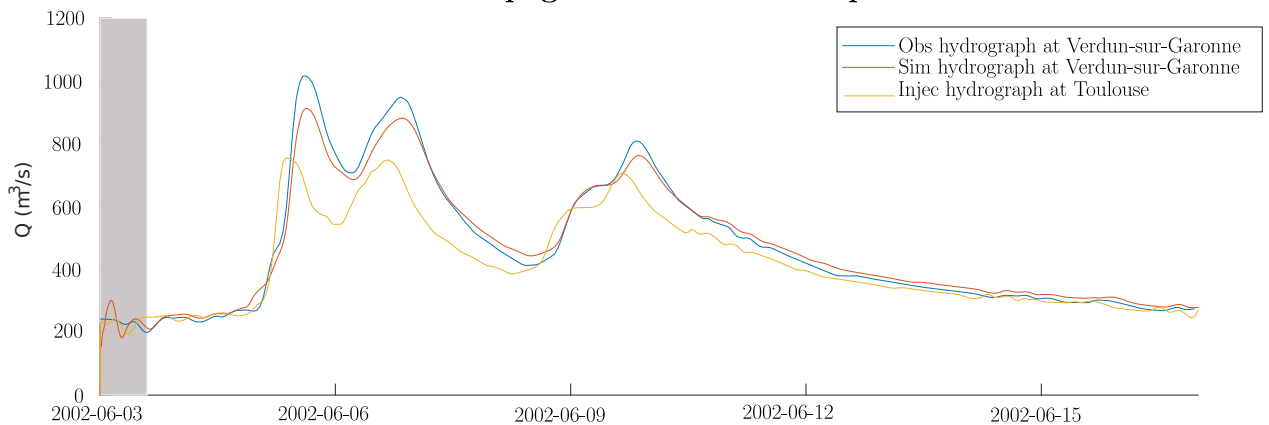


Figure 3.23: Propagation time between injected hydrograph at Toulouse inlet and the simulated hydrograph at Verdun-sur-Garonne outlet.

The Figure 3.24 shows a mapping of the water depths on the simulated domain at the time of the peaks. We find water depths between 4m and 6m which is in agreement with the known water depth in this reach. The results in terms of calculation time are reasonable (several days) but without optimizing the resolution and adaptivity criteria.

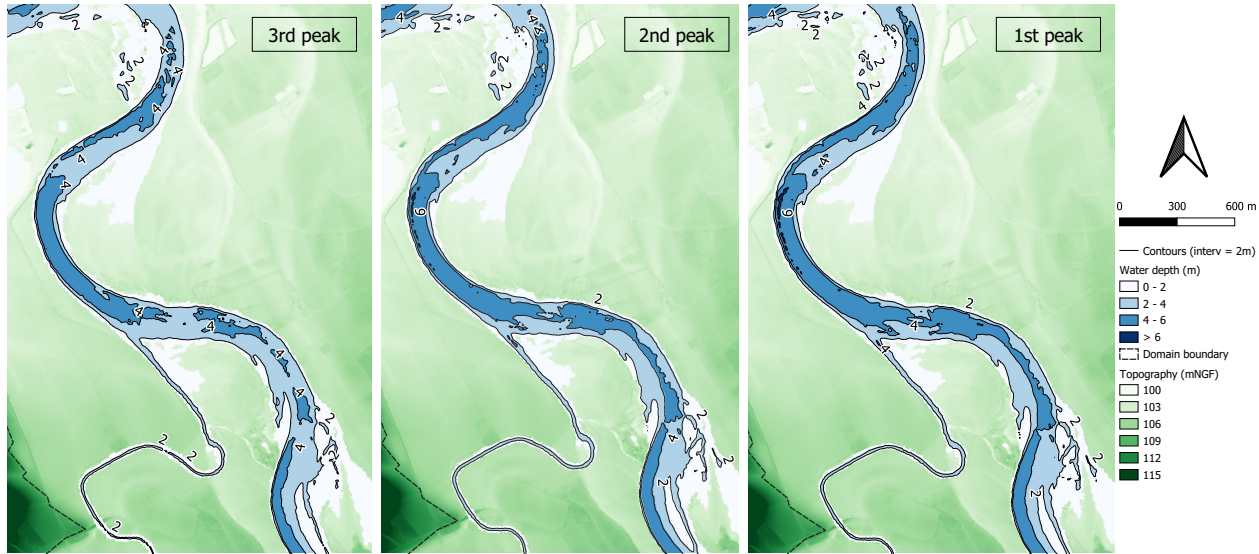


Figure 3.24: Maps of water depth at the 3 peak consecutively at 2002-06-05 09:00 UTC, 2002-06-06 16:00 UTC, and 2002-06-09 16:00 UTC

3.3 - Calibration of floodplain friction term

In the previous section, we calibrated a reach-averaged Manning coefficient for the channel. We now consider this coefficient known ($n_{ch} = 0.040 \text{ s} \cdot \text{m}^{-1/3}$), and we seek a procedure to calibrate the Manning roughness coefficient in the floodplain, n_{fp} . For this, we will use data of the major flood event of June 2000, which return period has been estimated to about 25 years. This flood caused damages that reached 20 M euros according to the CCR (Caisse Centrale de Réassurance⁴). Figure 3.25 shows the observed flood hydrograph and water surface elevation at the downstream station of Verdun-sur-Garonne between 2000-06-08 00:00 UTC and 2000-06-17 00:00 UTC, with a peak discharge of about $3750 \text{ m}^3/\text{s}$ on June, 11th (the maximum upstream in Toulouse was observed on June, 10th).

We will chose the value of n_{fp} using two sources of information:

- first, we will use flood marks surveyed just after the event at several locations in the floodplain,
- second, we will look at the timing and error on observed water surface elevation at the downstream station of Verdun-sur-Garonne, just as we did for the June 2002 non-overflowing event.

3.3.1 - First estimate of floodplain roughness using flood marks

Flood marks are traces left by the inundation on buildings, mud and debris on roads, etc. The agents responsible for post-flood surveys retrieve data related to each mark site: location, a few photos, and an estimated water depth (offset from the ground) at this location. This information is important for the calibration of river models. In this section, we present a methodology to calibrate the floodplain friction coefficient using

⁴https://catastrophes-naturelles.ccr.fr/-/001679_inondations-du-sud-ouest-en-juin-2000

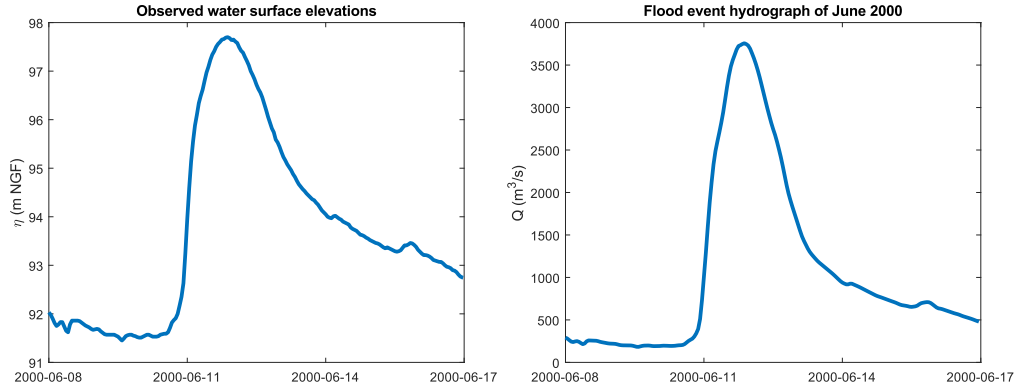


Figure 3.25: Flood event hydrograph of June 2000 from 2000-06-08 00:00 UTC to 2000-06-17 00:00 (right) and the corresponding water surface elevation η (m a.s.l. in the NGF system). They are measured in Verdun-sur-Garonne station and provided by “Hydroportail” platform (www.hydro.eaufrance.fr).

flood marks observed after the flood event of June 2000.

The dataset we used consist in 9 marks listed in Table 3.7 (more details are displayed in Supplementary materials, Appendix A of the thesis manuscript). The raw data provides the location and offset from the ground of each mark. Unfortunately, no absolute elevation is associated to the marks, so we translated this data into maximum absolute water surface elevation η_{obs} with two assumptions:

- first, we consider that the offset from the ground corresponds to the maximum submersion depth at the location of the mark; for example, a zero offset for a mud trace on a road will be interpreted as a maximum water surface elevation just equal to the local ground elevation;
- second, we estimated local ground elevation with the RGE ALTI LiDAR DEM of the IGN at 1-meter pixel size.

ID	Code	Lat (WGS84)	Lon (WGS84)	Z (mNGF)	Offset (m)	η_{obs} (m)
1	GTL_R_4798	1.4066908	43.6341388	128.58	0.2	128.78
2	GTL_R_4799	1.4014359	43.6387163	126.75	1.5	128.25
3	GTL_R_4800	1.3933053	43.6757865	121.69	0.5	122.19
4	GTL_R_4801	1.3707427	43.6825321	120.22	0.42	120.64
5	GTL_R_4802	1.3675042	43.7035736	117.94	0	117.94
6	WEB_R	1.3649320	43.7101942	117.8	0	117.8
7	GTL_R_4804	1.3316509	43.7756331	108.98	0	108.98
8	GTL_R_4805	1.3108768	43.7797117	105.98	1.4	107.38
9	GTL_R_4807	1.2949808	43.7752059	105.51	1.32	106.83

Table 3.7: Code, position, ground elevation, offset from the ground and corresponding water surface elevation $\eta_{obs} = Z + \text{Offset}$ for the 9 flood marks available for the June 2000 flood event; the offset from the ground is interpreted as a maximum submersion depth. This data comes from the national collaborative platform of flood sites and marks.

Sometimes, the GPS positions of the marks given in the database did not match a close visual inspection on Google Street View or on high-resolution orthophotos; a small shift in the coordinates was then necessary (usually a few tens of meters, but up to 250 m with a high remaining uncertainty for flood mark GTL_R_4800).

In the Basilisk literature, the use of this kind of marks to calibrate the friction coefficients is limited to a comparison between the observed and the simulated water depths (relative offset from local dry topography) at the location of the mark (*Kirstetter et al.*, 2021). However, Basilisk computes water depths and velocities by adapting the meshes according to the presence or absence of water. This adaptivity affects the resolution of topography (which is the reference level) and consequently decreases the accuracy of the comparison. In fact, even the LiDAR DEM at the 1-meter resolution misses very local topographic features such as building shapes, walls, ditches, etc. This effect is obviously even stronger when using a downsampled version of topography, such as used in Basilisk. Figure 3.26 illustrates an example of this issue, (A) shows the position of the flood mark (red dot), the contours of water depth (dashed black lines) and the contours of water surface elevation in meters a.s.l. (yellow lines, NGF system). At the location of the mark, the local topography is at 105.98 m in the 1×1m high resolution topography (C), and the offset of the mark from the ground is mentioned to be 1.4 m. This gives “point-scale” estimates of local high water attributes:

$$\begin{cases} z_{b,\text{point}} = 105.98 \text{ m} \\ h_{HW,\text{point}} = 1.40 \text{ m} \\ \eta_{HW,\text{point}} = 107.38 \text{ m} \end{cases}$$

Now if we look at the Basilisk simulation, the corresponding grid values read:

$$\begin{cases} z_{b,(8\times 8)} = 106.30 \text{ m} \\ h_{HW,(8\times 8)} = 1.70 \text{ m} \\ \eta_{HW,(8\times 8)} = 108.00 \text{ m} \end{cases}$$

We see that, due to resolution effects, the cell-averaged topography in the cell containing the mark is 0.32 meters above the point-scale estimate. If we were to compute the error by comparing maximum observed and simulated depths, we would find an error of $h_{HW,\text{point}} - h_{HW,(8\times 8)} = 1.70 - 1.40 = +0.30$ m. However, we see from Figure 3.26-A that the smoothest variable is the water surface elevation (in meters a.s.l. in the NGF system), which corresponds to an energy level. Therefore, a comparison of water surface elevations in mNGF is more robust. Here it gives an error $\eta_{HW,\text{point}} - \eta_{HW,(8\times 8)} = 108.00 - 107.38 = +0.62$ m, quite different from the error on depth. These specific values are an illustration and it must not be concluded that the error computed on η is always larger than the error on h . The conclusion is rather that the error on depth is substantially more biased by topographic averaging than the error on η , and that the latter variable should be preferred for optimization of roughness coefficients.

Algorithm 1 explains how η_{HW} is updated at each time step of the Basilisk simulation: the idea is to capture the maximum water surface elevation at the flood mark. However, in some cases Basilisk will just fail to flood pixels where a flood mark was observed. Recalling that flood mark offsets might be zero (e.g., mud trace on a sloping road), a zero simulated water depth should not be considered as a criterion. Sticking to the idea of comparing water surface elevations only, we then search for the nearest “wet” cell grid in a 16-m-wide (8m grid cell ×2) neighborhood of the flood mark if no flooding was simulated there (Figure 3.27).

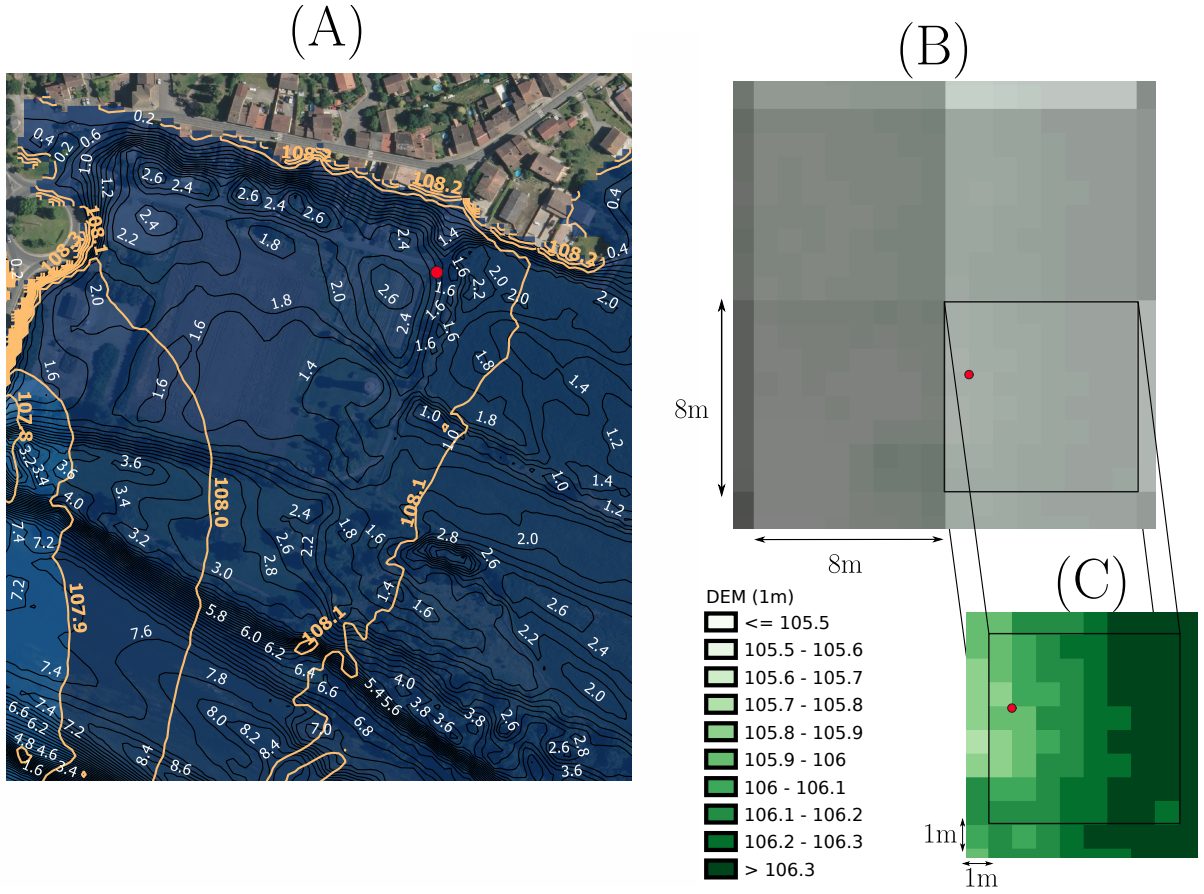


Figure 3.26: (A) Position of the mark GTL_R_4805 of the 2000 flood event on the Garonne River. The mark is represented by the red dot. The blue semi-transparent layer represents the extent and the elevation of the high waters η_{HW} between 107.7m and 108.2m. These water surface elevations are represented by yellow contours. The black dashed contours are the water depths; (B) Zoom on the red dot in Figure (A) around the position of the mark. It is formed by 4 8×8 m pixels. These are the Basilisk pixels; (C) A zoom on a single 8×8 m pixel constituted by 64 pixels of 1×1 m of the DEM with a resolution of 1m.

Algorithm 1 Algorithm that computes the high water surface elevation at the flood marks. η_{HW} is updated only if η increases and the cell is not dry at the current iteration.

```

 $\eta_{HW} \leftarrow z_b$ 
for  $t \in \text{time}$  do
  compute  $h$  and  $\eta$ 
  for  $i \in \text{cells}$  do
    if  $\eta(t, i) > \eta_{HW}(t, i)$  and  $h(t, i) > \text{dry}$  then
       $\eta_{HW}(t, i) = \eta(t, i)$ 
    end if
  end for
end for

```

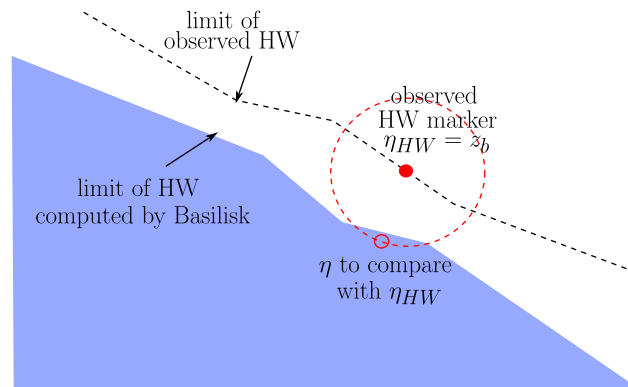


Figure 3.27: The procedure of extracting the high water (HW) elevation simulated by Basilisk η_{HW} at the mark. This water surface elevation can be extracted from the intersection of the red circle with the high water limit computed by Basilisk. This circle has as origin the mark position with a radius of 8m to 32m.

The calibration strategy of the floodplain Manning coefficient n_{fp} consists in running flood simulations in transient regime and evaluating water surface elevations at the flood marks positions. The idea is to vary the floodplain Manning coefficient for each simulation and then to look for the Manning value that minimizes the error between the observed water surface elevation (η_{obs}) and the simulated high water surface elevation (η_{HW}) of the flood marks.

The value of the channel Manning coefficient is $0.040 \text{ s} \cdot \text{m}^{-1/3}$ (the one found in the previous calibration part). Three Manning coefficients n_{fp} are tested (0.050 , 0.059 , and $0.067 \text{ s} \cdot \text{m}^{-1/3}$). Each simulation covers the 10-day period between 2000-06-08 00:00 UTC and 2000-06-17 00:00 UTC (see the flood hydrograph, Figure 3.25).

We use the same topography, the same discharge inlet principle presented in the previous section and the same setup. The injected hydrographs are presented in the Figure 3.12. We apply refinement on the inlets and also on the flood marks. We change in the configuration of the parallel computing. The simulations are executed with 1 node and 64 processors on the same MESOcentre cluster (Climserv IPSL). The Table 3.8 below shows the setup of the simulations.

Min resolution (m)	Max resolution (m)	Duration (days)	Tolerance on water depth (cm)	Tolerance on water surface elevation (cm)	n_{ch} ($\text{s} \cdot \text{m}^{-1/3}$)
512	8	10	1	1	0.040

Table 3.8: Simulations setup for the calibration of the floodplain Manning coefficient. These parameters are related to the minimal and maximal resolution and criteria of the adaptive mesh refinement.

We present results of the high water surface elevations of the flood marks in the Table 3.9. For each flood mark and each selected floodplain Manning coefficient, we present their observed water surface elevation, their simulated high water surface elevation and the computed error. We compute an RMSE between all the marks and we find that the RMSE is minimal when a value of the Manning coefficient equal to $0.050 \text{ s} \cdot \text{m}^{-1/3}$ is chosen.

ID	Code	η_{obs} (m)	η_1 (m)	η_2 (m)	η_3 (m)	ϵ_1 (m)	ϵ_2 (m)	ϵ_3 (m)
1	GTL_R.4798	128.78	129.28	129.27	129.25	+0.50	+0.49	+0.47
2	GTL_R.4799	128.25	128.96	128.88	128.71	+0.71	+0.63	+0.46
3	GTL_R.4800	122.19	122.63	122.6	122.55	+0.44	+0.41	+0.36
4	GTL_R.4801	120.64	120.98	120.95	120.81	+0.34	+0.31	+0.17
5	GTL_R.4802	117.94	118.49	118.46	118.46	+0.55	+0.52	+0.52
6	WEB_R	117.80	117.90	117.85	117.83	+0.11	0.05	0.03
7	GTL_R.4804	108.98	109.83	109.79	109.73	+0.85	+0.81	+0.75
8	GTL_R.4805	107.38	108.06	108.00	107.93	+0.68	+0.62	+0.55
9	GTL_R.4807	106.83	107.37	107.32	107.25	+0.54	+0.49	+0.42
RMSE (m)						0.56	0.52	0.46

Table 3.9: Flood marks results: η_1 , η_2 , and η_3 correspond respectively to the high water surface elevation at the flood mark position using different values of Manning floodplain coefficient $n_{fp,1} = 0.067 \text{ s} \cdot \text{m}^{-1/3}$, $n_{fp,2} = 0.059 \text{ s} \cdot \text{m}^{-1/3}$, and $n_{fp,3} = 0.050 \text{ s} \cdot \text{m}^{-1/3}$. ϵ_1 , ϵ_2 , and ϵ_3 are the corresponding errors between observed and simulated water surface elevation at the flood marks, with $\epsilon_1 = (\eta_1 - \eta_{obs})$, $\epsilon_2 = (\eta_2 - \eta_{obs})$, and $\epsilon_3 = (\eta_3 - \eta_{obs})$.

3.3.2 - Evaluation of Manning floodplain coefficient values using flood dynamics

To evaluate the choice of the Manning floodplain coefficient, we compare the flood dynamics of two simulations. The first one is performed using the $n_{fp,2} = 0.059 \text{ s} \cdot \text{m}^{-1/3}$ and the second one is performed using $n_{fp,3} = 0.050 \text{ s} \cdot \text{m}^{-1/3}$. At the reach downstream at Verdun-sur-Garonne, the simulated hydrograph and

water surface elevations are illustrated in the Figure 3.28. It shows that the simulated hydrograph overlaps the observed hydrograph and the volume of water is captured in the downstream of the reach with an error on the peak discharge about 6% for both simulations using $n_{fp,3}$ and $n_{fp,2}$. The water surface elevation did not reach the maximum elevation in the station with an error of $\sim 1\text{m}$ in both simulations. However, the flood recession is well represented in the water surface elevations and the hydrograph.

The flood propagation time is calculated at the peak between the main inlet at Toulouse and the downstream of the reach at Verdun-sur-Garonne. From Figure 3.29, we can find that the observed flood propagation time is $\Delta t_{obs} = 8h$ and the simulated flood propagation times are $\Delta t_2 = 6h30min$ and $\Delta t_3 = 6h$. The error in timing in the peak is about 1h30 for $n_{fp,2}$ and 2h for $n_{fp,3}$. We choose the $n_{fp,2}$ value since the propagation dynamics are well performed and the RMSE is still acceptable even if it is higher than the RMSE of $n_{fp,3}$.

The computing time is more than 10 days. This result is acceptable if we take into account the large domain and also its high resolution (8m). Moreover, as shown before, the results at the flood marks elevation are satisfactory and we find an average error of 52cm between the observed and simulated water surface elevation values in the flood marks.

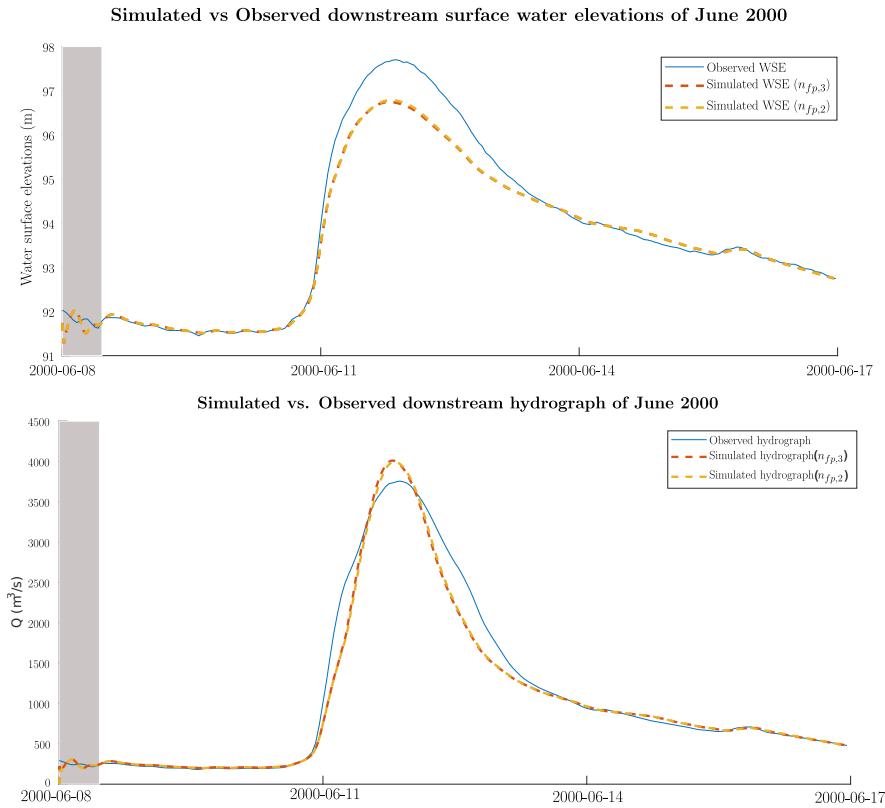


Figure 3.28: In the top, we present the observed vs. simulated water surface elevations in m NGF at Verdun-sur-Garonne for the June 2000 flood event and for the two Manning floodplain coefficients $n_{fp,2}$ and $n_{fp,3}$. we capture locally the water surface elevations in the channel with the function *segment_etamean()*. In the down, we present the simulated vs. observed hydrographs for $n_{fp,2}$ and $n_{fp,3}$ captured in the largest possible cross-section to cover the entire volume of water. The grey area in the Figures shows a “warm-up” period during which the simulated values should be ignored.

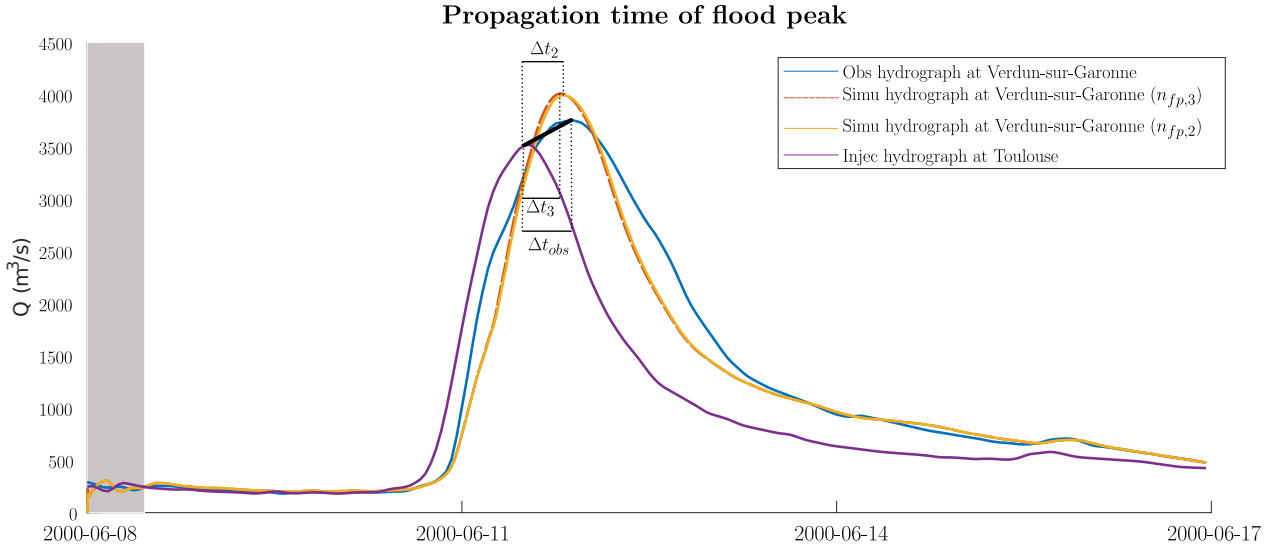


Figure 3.29: Propagation time between injected hydrograph at Toulouse inlet and the two simulated hydrographs at Verdun-sur-Garonne outlet for $n_{fp,2} = 0.059 \text{ s} \cdot \text{m}^{-1/3}$, and $n_{fp,3} = 0.050 \text{ s} \cdot \text{m}^{-1/3}$. Propagation times of the flood peak are represented as $\Delta t_{obs} = 8h$, $\Delta t_2 = 6h30min$, and $\Delta t_3 = 6h$.

4 - Conclusion

In this study, we presented new developments in the Basilisk model. This model solves the 2D Saint-Venant equations using adaptive grid refinement. We specifically improved and developed subroutines to handle tributary inflows with parallel computing in flood simulations on large domains. Moreover, we built a methodology for calibrating the friction coefficient in the floodplain using flood marks. The model was tested on a 40-km-long reach of the Garonne River with a high resolution bathymetry (8 m). The calibration of the Manning coefficients was done using the downstream rating curve for channel bed roughness (as it was not used as a boundary condition), while the calibration of the coefficient in the floodplain was done using flood marks. From this case, we demonstrate that the Basilisk model with the new developments can simulate high resolution floods in a reasonable time, even though a sensitivity analysis on tolerance criteria for grid adaptivity should be performed in order to further optimize the computational time.

Appendix A : Calibration of the GRSD hydrological model for tributaries

This part is for calibrating the GRSD hydrological model for tributaries.

A.1 - Subdomains and multiobjective optimization criteria

The calibration of the model on the domain is as follows. We first independently calibrate three parameter sets:

- a parameter set for the Hers-Mort subcatchment (largest right-bank tributary of the Garonne in blue on Figure 3.10),
- a parameter set for the Save subdomain (largest left-bank tributary in pink/red on Figure 3.10),
- a parameter set common to the three other smaller left-bank tributaries: Touch, Aussonnelle and Saint-Pierre (in green on Figure 3.10),

The goal is to have enough calibration sites in each subdomain, for a robust identification of routing parameters (celerity and diffusivity). In each subdomain, we perform a multiobjective calibration of the model using the CaRaMEL algorithm (*Monteil et al., 2020*). Two (2) signatures are computed for each available streamflow series, each consisting in a Kling-Gupta Efficiency criterion (*Gupta et al., 2009*):

- the KGE on hourly flows, simply coined ‘KGE’,
- the KGE on inter-annual daily regime (KGE daily regime, ‘KGEr’), which reflects the interaction between water and energy availability as well as catchment storage (*Monteil et al., 2020*); this is an interesting signature for recessions.

Then, the three (3) objective functions passed to the optimizer are:

- F_1 the mean value of hourly KGE at the downmost gauging site(s) in the subdomain, the ultimate criterion for the simulation,
- F_2 the mean value of hourly KGE at internal gauging sites, which will help identify routing parameters,
- F_3 the mean value of KGEr (KGE daily regime) at all sites, which will help identify production parameters.

Then we select as the final parameter set the one maximizing F_1 in the Pareto set.

Once the model has been calibrated on the tributaries, a last parameter set is optimized for the Portet–Verdun subdomain in yellow on Figure 3.10. This last step may seem unnecessary, as the propagation in this reach will basically be performed much more precisely by Basilisk. It has two benefits, though:

- it provides a first-order timing correction between Portet-sur-Garonne and the main Garonne inlet at Toulouse (Pont des Catalans),
- it provides a check of consistency of volume increase between the two gauging stations of Portet and Verdun-sur-Garonne during large floods events, for which the rating curves have large uncertainties.

A.2 - Results

We show in Table 3.10 the criteria computed at all available gauging sites (internal sites are in italics). The results are satisfactory except for the Girou, a subtributary of the Hers-Mort River. As the results are satisfactory for the internal site of Bourg-Saint-Bernard, it is difficult to know if this is a failure of the model, or measurements issues.

Gauging site	KGE hourly flows	KGEr daily regime
Touch @ Toulouse	0.730	0.723
<i>Touch @ Bérat</i>	<i>0.592</i>	<i>0.581</i>
<i>Touch @ Plaisance</i>	<i>0.729</i>	<i>0.690</i>
Aussonnelle @ Seilh	0.880	0.872
Hers-Mort @ Toulouse	0.704	0.673
<i>Saune @ Quint-Fonsegrives</i>	<i>0.701</i>	<i>0.705</i>
Girou @ Cepet	0.235	0.128
<i>Girou @ Bourg-St-Bern.</i>	<i>0.668</i>	<i>0.689</i>
Save @ Larra	0.565*	0.329
<i>Save @ Lombez</i>	<i>0.619*</i>	<i>0.609</i>
Garonne @ Verdun	0.917	0.943

Table 3.10: Calibration results for the gauging sites of the domain.

3.5 Conclusion du chapitre

Ce chapitre avait pour objectif de produire une simulation de crue à haute résolution sur un tronçon de la Garonne, simulation qui servira de référence (“pseudo-observations”) pour la méthode d’inversion à l’aide d’un modèle géométrique simplifié (modèle périodique).

Dans un premier temps, la combinaison des Orthophotos avec des données bathymétriques *in situ* a confirmé le potentiel de la télédétection passive dans la construction de la bathymétrie continue 2D du lit mineur. Elle a permis une identification haute fréquence des unités morphologiques et de leur variabilité longitudinale et transversale.

Cette méthodologie, utilisée dans la construction de la bathymétrie, peut être utilisée en accompagnement d’images dans l’infrarouge thermiques sur lesquelles sont appliquées la méthode LSPIV, pour estimer le débit dans un tronçon donné : la combinaison de la profondeur donnée par l’imagerie visible et proche infrarouge, et de la vitesse donnée par la méthode LSPIV, permet d’estimer le débit. *Legleiter et al.* (2017) ont testé cette approche hybride avec succès sur cinq cours d’eau en Alaska.

Pour aller plus loin, nous pourrions tester la méthodologie présentée dans ce chapitre sur un autre tronçon avec des données moins incertaines (par exemple, Sacramento River, US). La méthodologie présentée avec 4 canaux (RVB + proche infrarouge) pourrait être également être étendue à l’imagerie hyperspectrale, ou à d’autres sources de données multispectrales (par exemple, Landsat 8 ou Sentinel).

Dans la partie 2.3, cette bathymétrie du lit mineur a été combinée avec la bathymétrie du lit majeur (MNT de 1m) pour construire la topographie utilisée dans la simulation de référence. Néanmoins, cette simulation nécessite un calage des coefficients de Manning du lit mineur et du lit majeur. Le lit mineur a été calibré à l’aide de la courbe de tarage en aval à Verdun-sur-Garonne et la dynamique de l’événement non-débordant de la crue de juin 2002, tandis que le lit majeur a été calibré à l’aide des repères et des laisses de crue PHE de la crue de juin 2000, ainsi que l’hydrogramme aval à Verdun-sur-Garonne. Une réflexion spécifique sur la variable à utiliser pour la comparaison repère ponctuel / maille a été menée, conduisant à préférer la comparaison directe sur la variable de cote de la surface libre η .

Les coefficients de Manning trouvés sont $n_{ch} = 0.04 \text{ s} \cdot \text{m}^{-1/3}$ (*channel*) pour le lit mineur et $n_{fp} = 0.059 \text{ s} \cdot \text{m}^{-1/3}$ (*floodplain*) pour le lit majeur. Comme mentionné précédemment, l’ensemble des paramètres du modèle Garonne (bathymétrie et coefficients de rugosité) et simulations hydrauliques résultantes seront considérées dans la suite comme des pseudo-observations.

Cette étude n’est pas encore terminée. Par manque de temps, nous n’avons pas testé une gamme très large de valeurs de Manning en lit majeur. Globalement, le modèle fonctionne bien et donne de bons résultats, malgré quelques points surprenants comme un biais positif systématique sur les cotes simulées aux repères de crue, simultanément à une sous-estimation de la cote à Verdun-sur-Garonne. De petits écarts de volume sont également constatés entre la somme des volumes entrants (Portet + affluents) et le volume sortant d’après l’hydrogramme mesuré à Verdun-sur-Garonne : la compréhension fine de ces écarts reste à mener.

A court terme, le test d’autres valeurs de coefficients de frottement est prioritaire, ainsi qu’une éventuelle spatialisation de celui-ci en lit mineur pour tenir compte du secteur relativement endigué juste à l’aval de Toulouse. Enfin, une étude de sensibilité aux critères de raffinement de la grille pourrait être intéressante pour optimiser le temps de calcul.

Dans le chapitre suivant, nous utilisons les données de bathymétrie/simulations pour identifier des signatures hydrauliques, morphologiques et bathymétriques définies au tronçon. Ces signatures sont utilisées dans le Chapitre 5 dans l’algorithme d’inversion.

Chapitre 4

Signatures morphologiques, bathymétriques et hydrauliques au tronçon

4.1 Introduction

Au Chapitre 3, nous avons construit une bathymétrie détaillée sur le tronçon d'étude de la Garonne, avec comme données *in situ* de base des levés de sections en travers. L'objectif de ce chapitre est maintenant de définir un certain nombre d'indices caractéristiques d'un tronçon, sans nécessairement les rattacher à une section précise. En suivant une définition proche de celle de *Le Coarer* (2007), nous appelons *signature au tronçon* un indice ou une courbe calculée directement à l'échelle du tronçon, sans isoler l'échelle de la ou des section(s) individuelle(s) dans le tronçon. Il peut s'agir typiquement de paramètres de section équivalente, comme définis par *Rebolho et al.* (2018) ou *Zheng et al.* (2018), mais aussi de courbes caractéristiques du tronçon ou d'indicateurs de variabilité d'un paramètre. Ces signatures peuvent être relatives à plusieurs aspects :

- le tracé en plan du tronçon
- sa topographie / bathymétrie globale,
- la topographie plus spécifique du lit majeur,
- son fonctionnement hydraulique.

Les différentes parties de ce chapitre traiteront donc de chacune de ces familles de signatures, qui seront utilisées au chapitre suivant pour définir un tronçon équivalent au tronçon réel.

4.2 Signatures relatives au tracé en plan

4.2.1 Sinuosité

Dans cette partie, nous repérons la position d'un point le long du cours d'eau par son abscisse curviligne s ou, de façon équivalente, par son point kilométrique (PK). En orientant les abscisses curvilignes de façon croissante vers l'aval, la sinuosité d'un bief entre 2 points A et B est définie comme le ratio entre la différence d'abscisse curviligne et la distance euclidienne entre A et B :

$$\sigma = \frac{s_B - s_A}{d_{AB}}$$

La sinuosité est une variable importante du fonctionnement hydraulique, puisqu'elle modifie la pente apparente et donc les frottements : le ratio entre la pente de la plaine d'inondation et la pente du chenal (mesurée en suivant la ligne d'écoulement centrale) est précisément égale à la sinuosité. C'est l'un des degrés de liberté géomorphologiques permettant d'ajuster la charge sédimentaire au débit liquide (*Lane, 1955*). La Figure 4.1 illustre cette variable sur le tronçon de la Garonne entre Toulouse et Verdun-sur-Garonne ($\sigma = 1.3$), et sur un cas plus extrême de la Shenandoah River, dans les Appalaches ($\sigma = 3.8$).

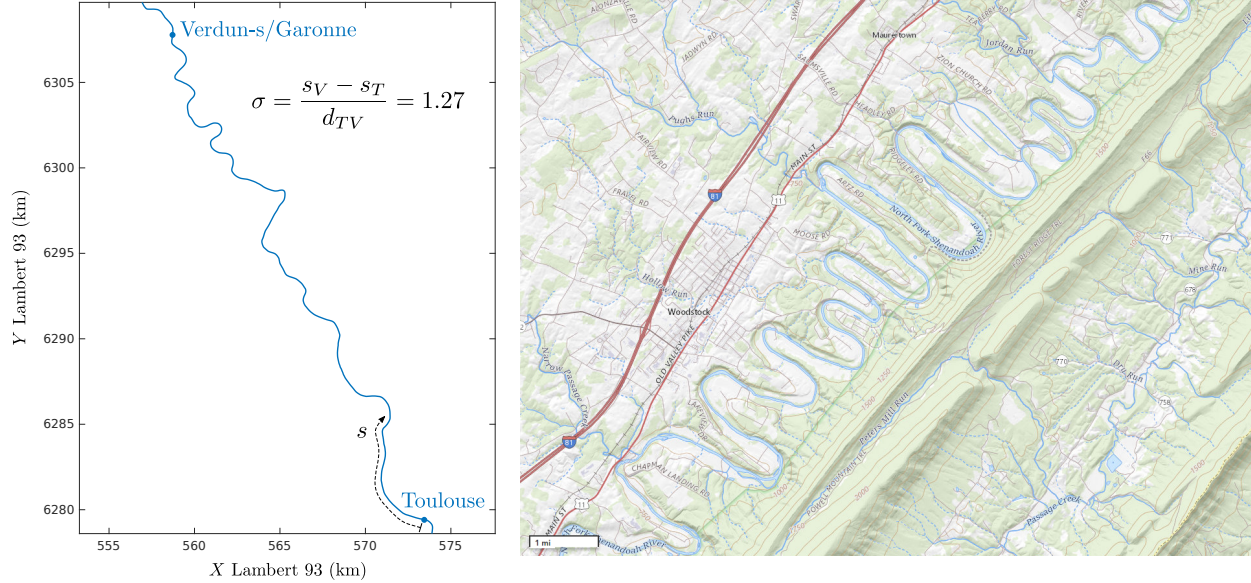


Figure 4.1: Exemples de sinuosité pour le tronçon de la Garonne (à gauche, $\sigma = 1.27$) ou de la North Fork Shenandoah River en Virginie (à droite, $\sigma = 3.8$).

4.2.2 Variance angulaire

Connaissant le tracé du cours d'eau dans le plan cartographique (ligne d'écoulement centrale), nous pouvons caractériser statistiquement sa variabilité angulaire. Cette étape est illustrée dans la suite à l'aide du tronçon de la Garonne entre Toulouse et Verdun-sur-Garonne, ce qui est également l'occasion de présenter les outils de description de la forme en plan en variables complexes qui seront utilisés pour la modélisation au chapitre suivant. Pour commencer, nous définissons la position de la ligne d'écoulement centrale par son affixe complexe ; par souci de clarté nous conservons la notation classique \mathbf{z} pour les affixes complexes, mais en caractère gras comme pour toutes les variables à valeurs complexes et pour éviter la confusion avec la cote topographique (qui sera notée z_b pour *bed* comme dans Basilisk) :

$$\mathbf{z}(s) = x(s) + i y(s) \quad \text{avec } i^2 = -1$$

où x et y sont typiquement en coordonnées Lambert 93. Le vecteur unitaire tangent, définissant la direction locale du chenal, sera défini par :

$$\mathbf{T}(s) = \frac{d\mathbf{z}}{ds}$$

Nous ajustons ensuite une courbe 'basse fréquence' au tracé, typiquement une spline de faible degré en s . Dans le cas présent, nous nous contentons d'une tendance régionale rectiligne,

$$\mathbf{z}_{\text{reg}}(s) = (s - s_T) \frac{\mathbf{z}_V - \mathbf{z}_T}{s_V - s_T} + \mathbf{z}_T$$

où $s_T = 0$ est l'abscisse curviligne à Toulouse, $s_V = 44904$ m celle à Verdun-sur-Garonne, $\mathbf{z}_T = \mathbf{z}(s_T) = x_T + i y_T = 573884 + 6278602i$ est l'affixe complexe correspondant à Toulouse, et $\mathbf{z}_V = \mathbf{z}(s_V) = x_V + i y_V = 556911 + 6309722i$ celle correspondant à Verdun-sur-Garonne. Nous voyons donc apparaître le vecteur direction moyen (régional) :

$$\mathbf{T}_{\text{reg}} = \frac{d\mathbf{z}_{\text{reg}}}{ds} = \frac{\mathbf{z}_V - \mathbf{z}_T}{s_V - s_T} = \frac{-16973 + 31120i}{44904} = -0.378 + 0.693i$$

Nous pouvons constater que ce vecteur n'est pas unitaire, nous pouvons faire apparaître son module et son argument complexe :

$$\mathbf{T}_{\text{reg}} = |\mathbf{T}_{\text{reg}}| e^{i\theta_{\text{reg}}} = 0.789 e^{2.07i}$$

Cette notation fait apparaître l'angle de la tendance régionale, $\theta_{\text{reg}} = 2.07$ rad, ce qui correspond bien à un azimuth Nord-Ouest ; par ailleurs, la propriété $|\mathbf{T}_{\text{reg}}| < 1$ traduit le fait qu'un déplacement de 1 mètre le long du cours d'eau occasionne un déplacement moins long en moyenne selon la direction régionale. On vérifie :

$$\frac{1}{|\mathbf{T}_{\text{reg}}|} = \frac{1}{0.789} = 1.27 = \sigma \quad \text{sinuosité}$$

Nous pouvons alors définir le résidu à la tendance régionale, qui représente la 'divagation' locale du chenal par rapport à la tendance régionale :

$$\mathbf{z}'(s) = \mathbf{z}(s) - \mathbf{z}_{\text{reg}}(s)$$

Le vecteur unitaire tangent se décompose alors en :

$$\mathbf{T}(s) = e^{i\theta(s)} = \frac{d\mathbf{z}'}{ds} + \frac{d\mathbf{z}_{\text{reg}}}{ds} = \frac{d\mathbf{z}'}{ds} + \mathbf{T}_{\text{reg}}$$

L'allure du résidu $\mathbf{z}'(s)$ pour le tronçon de la Garonne est illustré sur la Figure 4.2. Dans l'idéal, nous pourrions appliquer à ce résidu ou à sa dérivée une analyse par ondelettes du même type que celle présentée au Chapitre 2. Par simplicité et étant donnée la longueur du tronçon, nous nous contenterons d'une analyse de Fourier pour obtenir la dérivée $\frac{d\mathbf{z}'}{ds}$, en supposant que le résidu est périodique (il est nul à Toulouse et revient à une valeur nulle à Verdun-sur-Garonne). Nous utilisons le fait que la transformée de Fourier de la dérivée est obtenue simplement en multipliant par (ik) , où k est le nombre d'onde en rad/m (analogie spatiale de la pulsation pour une analyse temporelle ; la transformée consiste à passer d'une fonction de s à une fonction de k) :

$$\mathcal{F} \left[\frac{d\mathbf{z}'}{ds} \right] (k) = ik \mathcal{F} [\mathbf{z}'] (k)$$

La dérivée est ensuite obtenue par transformée inverse. Cette façon de procéder permet également d'accéder à d'autres propriétés de la forme en plan, notamment la courbure locale, qui fait l'objet de la section suivante.

La Figure 4.3 illustre la série $\theta(s) = \arg \{ \mathbf{T}(s) \}$ obtenue pour la Garonne (en pratique, pour éviter les discontinuités de l'angle en $-\pi/\pi$, il est préférable d'aligner préalablement la direction régionale avec l'axe réel par une rotation d'angle $-\theta_{\text{reg}}$, c'est-à-dire travailler sur $\mathbf{z}(s) \frac{|\mathbf{T}_{\text{reg}}|}{\mathbf{T}_{\text{reg}}}$). Une autre signature du tracé en plan du tronçon est la variance ou l'écart-type angulaire de la direction du chenal,

$$\sigma_\theta = \left[\frac{1}{s_V - s_T} \int_{s_T}^{s_V} \theta(s)^2 ds - \theta_{\text{reg}}^2 \right]^{\frac{1}{2}} = 0.68 \text{ rad}$$

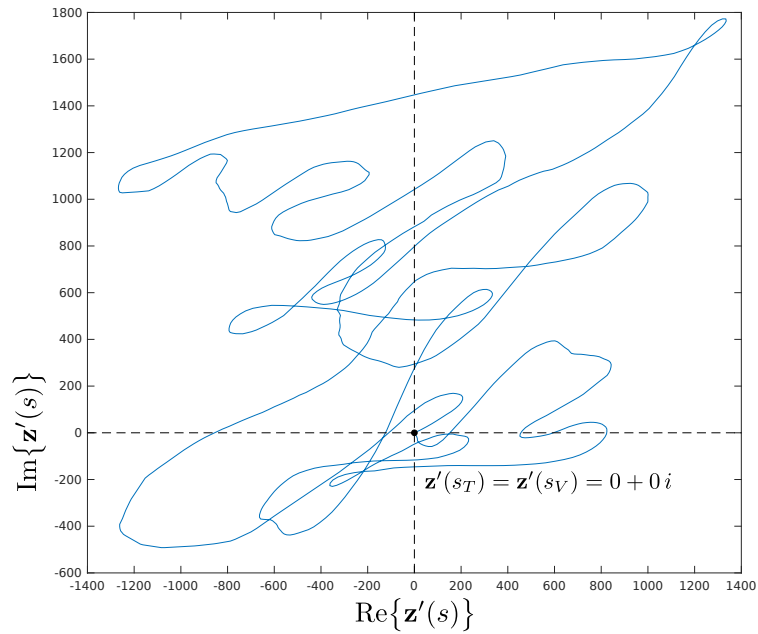


Figure 4.2: Divagation $\mathbf{z}'(s)$ par rapport à la tendance régionale du tracé en plan de la Garonne entre Toulouse (abscisse curviligne s_T) et Verdun-sur-Garonne (abscisse curviligne s_V). La tendance passant exactement par ces deux points, la divagation est égale au vecteur nul en s_T et en s_V et la courbe est fermée.

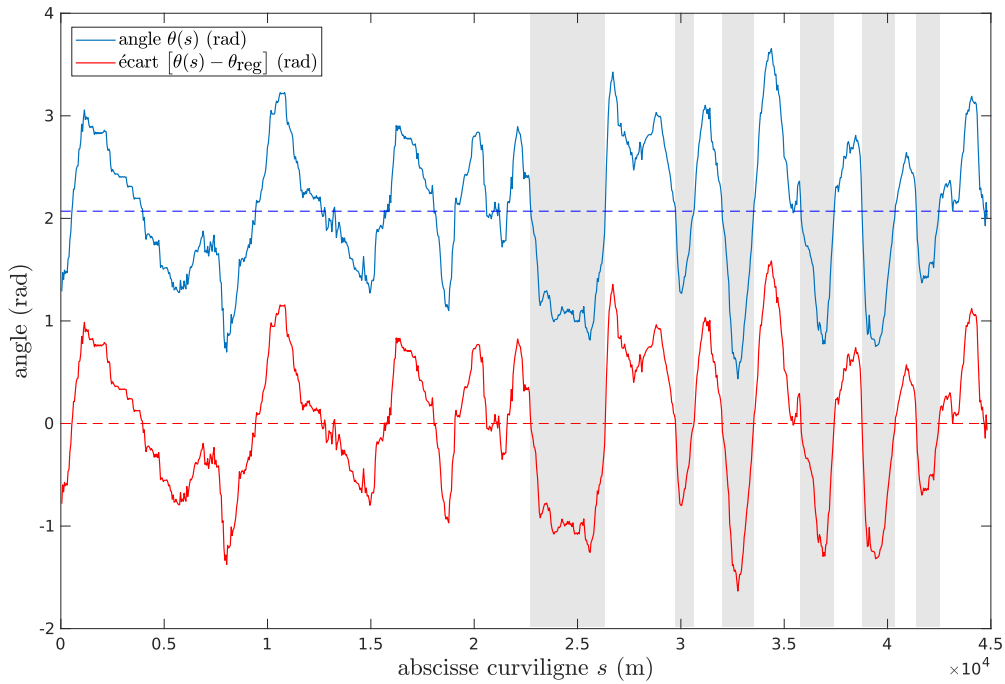


Figure 4.3: Angle de la direction locale du chenal par rapport à l'axe de X des coordonnées Lambert 93 (en bleu), ou relativement à la direction régionale (en rouge).

4.2.3 Longueur d'onde des alternances de faciès hydrauliques

Nous avons analysé en détail au Chapitre 2 les alternances de faciès hydrauliques, en prenant en compte simultanément les variables de formes en plan et les variables relatives à l'écoulement. Dans le cas présent pour la Garonne nous allons nous baser uniquement sur la forme en plan pour définir la longueur d'onde (mesurée en distance le long du chenal) de ces alternances. En effet, dans le cas d'observations télédéetectées, elles sont les seules directement exploitables. Sur la Figure 4.3, nous avons indiqué en gris les alternances dans la parties aval du tronçon, où elles sont bien marquées. Nous comptons 6 motifs complets de méandres pour une longueur de 22 km, ce qui donne une longueur d'onde

$$\lambda = \frac{22 \times 10^3}{6} \simeq 3700 \text{ m}$$

4.2.4 Courbure maximale, rayon de courbure minimal

La courbure locale de la ligne d'écoulement centrale peut être calculée en dérivant une seconde fois la fonction $\mathbf{z}(s)$. En effet, le vecteur tangent unitaire \mathbf{T} ayant une norme constante, sa dérivée lui est toujours orthogonale. En notant $\mathbf{N} = i\mathbf{T}$ le vecteur unitaire normal dirigé vers la rive gauche (rotation anti-horaire de $\frac{\pi}{2}$), on définit la courbe algébrique (signée) γ telle que :

$$\frac{d\mathbf{T}}{ds} = \gamma(s)\mathbf{N} = i\gamma(s)\mathbf{T}$$

De même que précédemment, on calcule $\frac{d\mathbf{T}}{ds}$ à l'aide de la formule :

$$\mathcal{F} \left[\frac{d\mathbf{T}}{ds} \right] (k) = ik \mathcal{F} [\mathbf{T}] (k) = (ik) (ik) \mathcal{F} [\mathbf{T}] (k) = -k^2 \mathcal{F} [\mathbf{z}] (k)$$

suivie d'une transformée de Fourier inverse. Nous pouvons en déduire la valeur du rayon de courbure local, qui est :

$$R(s) = \frac{1}{|\gamma(s)|}$$

Sur le tronçon de la Garonne, le rayon de courbure atteint un minimum au niveau de la confluence avec l'Hers-Mort ; il vaut alors environ 105 m (cf. Figure 4.4).

4.2.5 Voisinage tubulaire et difficulté à positionner des sections en lit majeur

La définition du rayon de courbure permet de mettre en évidence l'une des difficultés en hydraulique lorsque l'on utilise une approche filaire : celle de rattacher un point quelconque du lit majeur à une section du cours d'eau (point unique sur la ligne d'écoulement centrale). Cette bijection n'est en fait possible que si le point est situé dans le *voisinage tubulaire* du chenal, c'est-à-dire un buffer dont la demi-largeur locale est égale au rayon de courbure $R(s)$. Le voisinage du chenal dans lequel on est sûr de pouvoir faire cette bijection a donc comme demi-largeur la valeur minimale du rayon de courbure (en gris sur la Figure 4.4).

Pour palier ce problème, on peut définir un repère orthogonal différent de la base locale (\mathbf{T}, \mathbf{N}) : c'est le principe de la cartographie conforme (voir par exemple *Hilton et al.*, 2019). Un tel système sera étudié au chapitre suivant.

4.3 Définition d'un niveau de plein-bord au tronçon

Le débordement, c'est-à-dire le passage d'un écoulement confiné dans le chenal à un écoulement passant à la fois dans le chenal et dans le lit majeur, est une transition importante dans le fonctionnement hydraulique du cours d'eau. L'identification objective de la cote de plein-bord est une question ouverte (voir par exemple *Navratil*, 2005), différents critères de morphologie des berges pouvant être retenus. Dans notre description, nous allons caractériser les conditions de plein-bord comme celles qui maximisent le rayon hydraulique. Cette

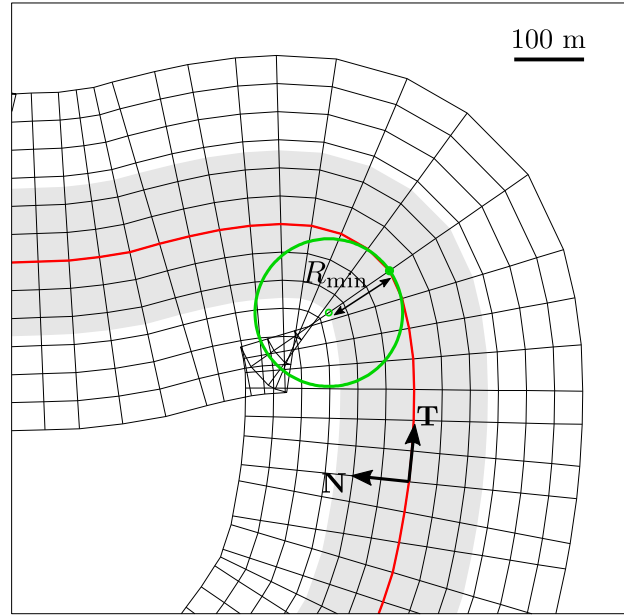


Figure 4.4: Zoom sur le coude de la Garonne au niveau de la confluence avec l'Hers-Mort. La courbure de la ligne d'écoulement centrale (en rouge) y est maximale, et le rayon de courbure minimal ($R_{\min}=105$ m, cercle en vert). On ne peut donc projeter un point en lit majeur de façon univoque sur le chenal que si celui-ci est situé dans un voisinage tubulaire de largeur R_{\min} (en gris).

définition est possible à la section (le rayon hydraulique étant le ratio entre la section mouillée et le périmètre mouillé), mais nous avons mentionné que nous ne souhaitons pas passer par cette échelle. Nous allons donc directement définir un niveau de plein-bord à l'échelle du tronçon.

4.3.1 Analyse de la topographie sans tendance

La première étape de l'analyse morphologique au tronçon va consister à retirer la tendance régionale linéaire de la topographie du tronçon $z_b(x, y)$. Illustrons cette opération sur le tronçon de la Garonne entre Toulouse et Verdun-sur-Garonne (voir la Figure 4.5) :

- la cote topographique de la Garonne donnée par le RGE ALTI 1m est d'environ 126.8 m NGF à Toulouse (Pont des Catalans) ;
- cette cote vaut 91.0 m NGF à Verdun-sur-Garonne ;
- ces deux points étant distants à vol d'oiseau de 31780 m, la pente régionale est de $I_{\text{reg}} = 1.1 \times 10^{-3}$ m/m. Comme mentionné plus haut, sa direction fait un angle $\theta_{\text{reg}}=2.03$ rad avec l'axe cartésien X des coordonnées Lambert 93. La topographie dépouillée de sa tendance régionale s'obtient donc par :

$$z'_b(x_{L93}, y_{L93}) = z_b(x_{L93}, y_{L93}) + I_{\text{reg}} \cos(\theta_{\text{reg}})x_{L93} + I_{\text{reg}} \sin(\theta_{\text{reg}})y_{L93} + \text{constante}$$

Nous pouvons arbitrairement translater cette topographie sans tendance pour obtenir des valeurs minimales proches de zéro. Nous choisissons finalement les constantes suivantes :

$$z'_b(x_{L93}, y_{L93}) = z_b(x_{L93}, y_{L93}) + \underbrace{I_{\text{reg}} \cos(\theta_{\text{reg}})}_{-4.88 \times 10^{-4} \text{ m/m}} (x_{L93} - x_{c0}) + \underbrace{I_{\text{reg}} \sin(\theta_{\text{reg}})}_{+9.86 \times 10^{-4} \text{ m/m}} (y_{L93} - y_{c0}) - 100$$

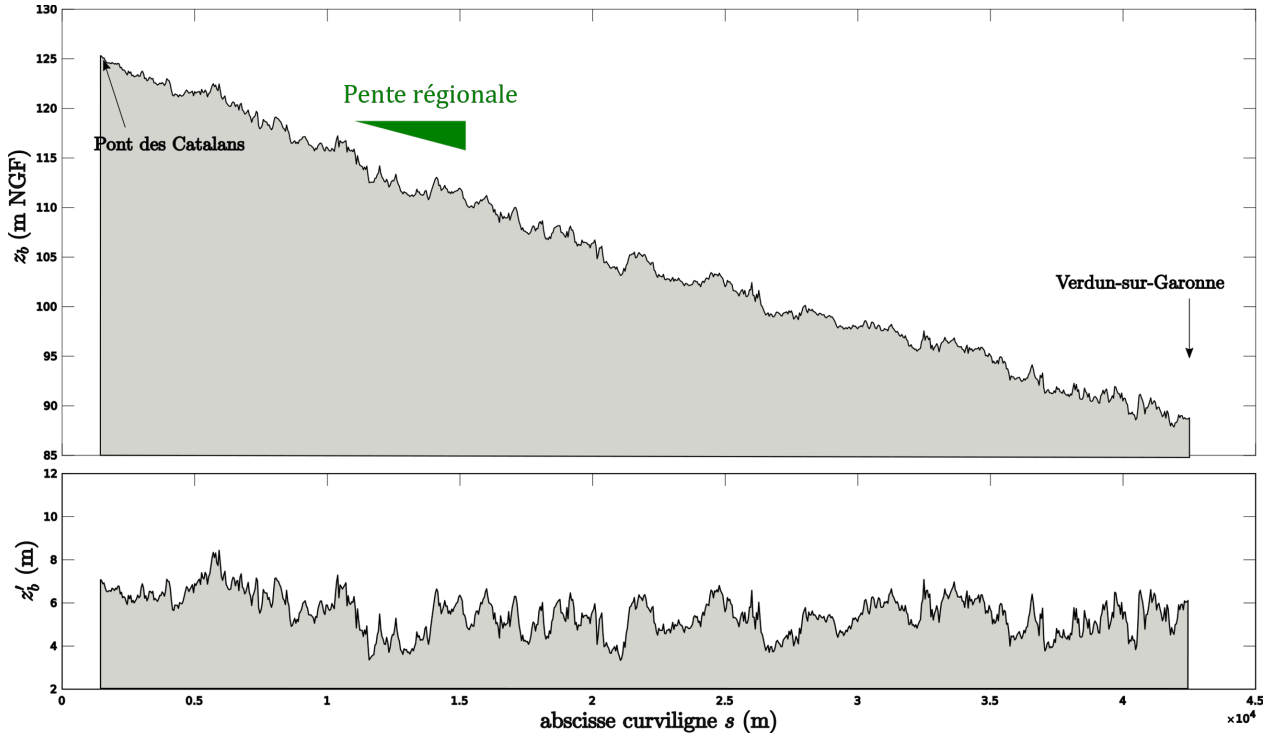


Figure 4.5: Illustration 1D de la construction de la topographie sans tendance régionale I_{reg} entre Verdun-sur-Garonne et Toulouse

où $(x_{c0}, y_{c0}) = (565\,000, 6\,294\,000)$ est un point de repères de coordonnées simples situé au milieu du domaine étudié. Il est à noter que, en suivant le cours de la Garonne, la pente hydraulique est plus faible que I_{reg} puisque la distance développée est plus longue (d'un facteur égal à la sinuosité) : elle vaut 8.7×10^{-4} m/m.

4.3.2 Rayon hydraulique équivalent au tronçon

Une fois obtenue la topographie sans tendance (plate en moyenne), nous pouvons appliquer sur la carte z'_b un seuil uniforme noté η' , qui correspond à une ligne d'eau dont la pente est égale à la pente régionale, et dont on enlève également la tendance. Nous pouvons alors regarder la surface totale des zones 'inondées' par ce seuil, telles que $z'_b(x, y) < \eta'$: nous notons cette aire $A(\eta')$ et nous pouvons la calculer pour différentes valeurs du seuil η' .

Parallèlement, pour chaque valeur de η' nous pouvons calculer sur la carte le volume sous le seuil :

$$V(\eta') = \iint_{A(\eta')} h(x, y) dA = \iint \max\{\eta' - z'_b(x, y); 0\} dx dy$$

Le ratio $\frac{V(\eta')}{A(\eta')}$ est alors analogue à un rayon hydraulique, nous le notons $R_h^*(\eta')$. Cette procédure est illustrée sur la Figure 4.6, en haut à gauche, et la courbe résultante est tracée sur la Figure 4.7 (en bleu). On peut alors déterminer la valeur de η'_{bf} (pour *bankfull*) qui maximise le rayon hydraulique équivalent au tronçon R_h^* : on identifie $\eta'_{\text{bf}} = 10.3$ m et $R_h^*(\eta'_{\text{bf}}) = 2.77$ m. En dessous de η'_{bf} , le volume augmente rapidement quand on augmente η' , alors que la surface inondée n'augmente pas beaucoup (elle reste confinée au lit mineur). En revanche, au-dessus de η'_{bf} , la surface inondée augmente rapidement avec η' alors que le volume stocké, lui, n'augmente pas beaucoup (la hauteur de submersion est faible en lit majeur). Il est donc logique que $\frac{V(\eta')}{A(\eta')}$ passe par un maximum.

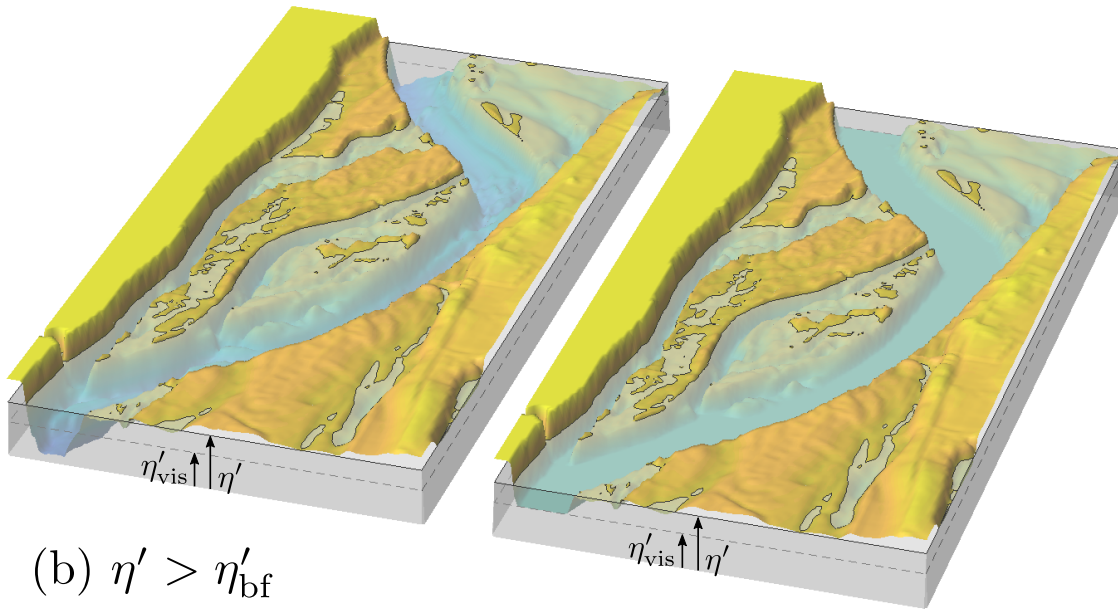
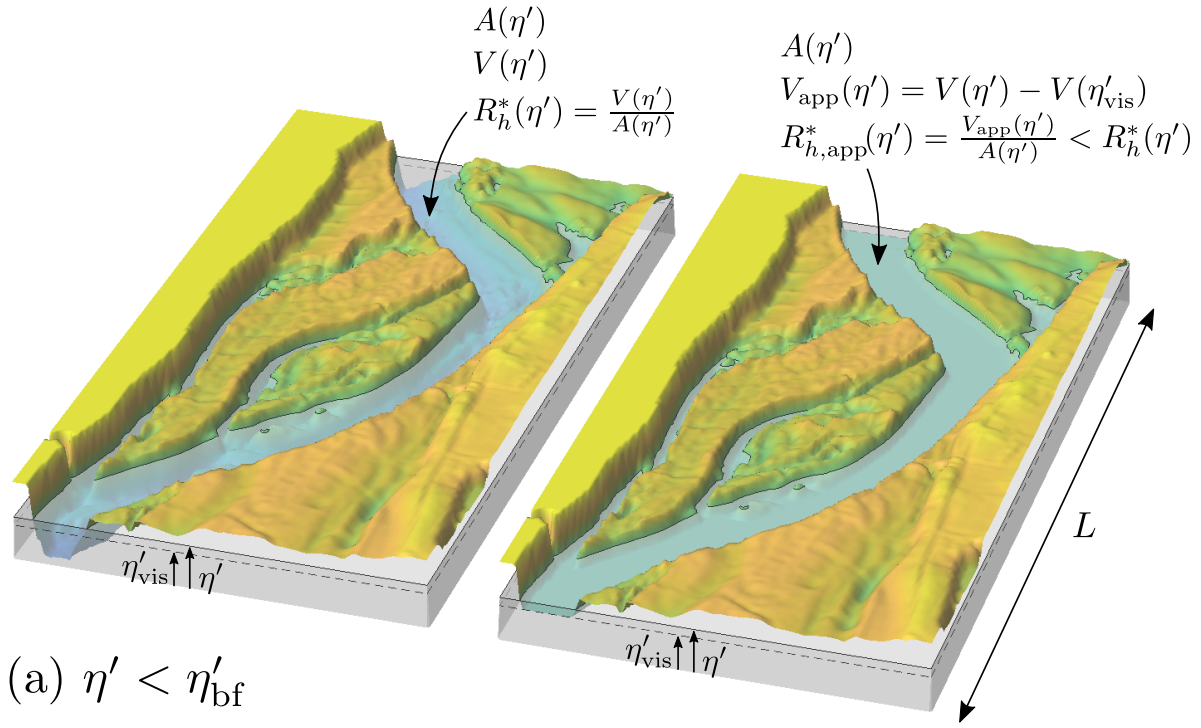


Figure 4.6: Procédure de la calcul du rayon hydraulique équivalent au tronçon. La variable η' représente le niveau d'eau appliqué sur la topographie z'_b débarrassée de sa tendance, $A(\eta')$ est l'aire inondée par ce seuil et $V(\eta')$ le volume sous le seuil. Si la topographie du lit mineur n'est pas accessible sous un niveau de visibilité η'_{vis} , le volume $V(\eta')$ n'est pas intégralement observable et nous n'avons accès qu'à une estimation $V_{\text{app}}(\eta') < V(\eta')$, qui biaise donc par défaut l'estimation du rayon hydraulique équivalent $R_h^*(\eta') = \frac{V(\eta')}{A(\eta')}$ (situation en haut à droite et en bas à droite).

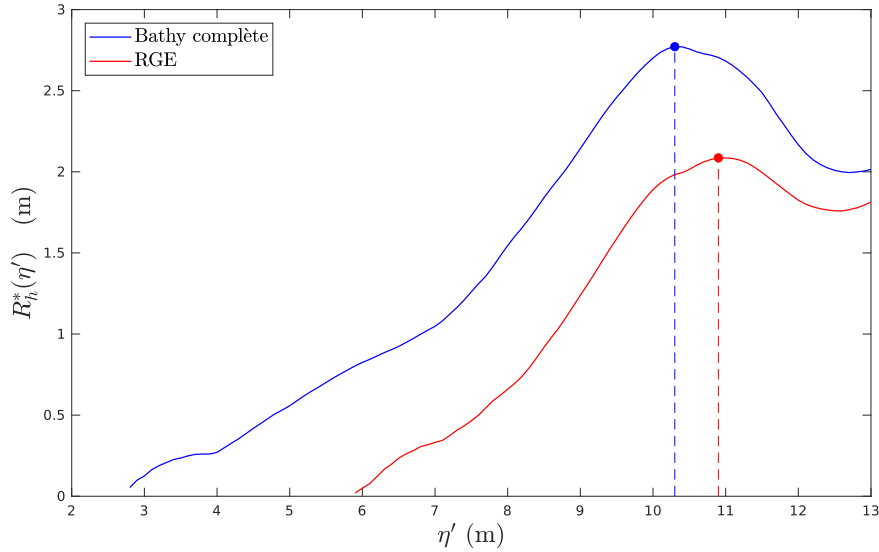


Figure 4.7: Estimation du rayon hydraulique équivalent de plein-bord pour le tronçon de la Garonne. En bleu, le rayon hydraulique équivalent est calculé en supposant que la totalité de la topographie en lit mineur est connue. En rouge, on suppose qu'elle n'est accessible qu'au-dessus d'un seuil de visibilité η'_{vis} . Le seuil de visibilité change les valeurs de R_h^* calculées, mais pas drastiquement la position en η' du maximum.

4.3.3 Largeur équivalente de plein-bord au tronçon

Si nous divisons l'aire inondée au plein-bord, $A_{\text{bf}} = A(\eta'_{\text{bf}})$, par la longueur L du tronçon mesurée selon la direction régionale, nous obtenons une grandeur analogue à une largeur de plein-bord. Cependant, il faut à nouveau prêter attention à l'effet de la sinuosité ; une estimation alternative (et plus conforme à l'idée intuitive) est de diviser A_{bf} par la longueur développée de chenal sur le tronçon, qui vaut σL :

$$W_{\text{bf}}^* = \frac{A(\eta'_{\text{bf}})}{\sigma L} \quad \text{largeur équivalente de plein-bord au tronçon}$$

Pour le tronçon de la Garonne, nous obtenons une valeur très proche de 200 m pour $\eta'_{\text{bf}} = 10.3$ m : $W_{\text{bf}}^* = 197$ m (Figure 4.8).

4.3.4 Influence d'un seuil de visibilité

Les conditions précédentes dans lesquelles nous avons identifié le niveau de plein-bord (connaissance de la bathymétrie complète) ne sont évidemment pas réunies dans le cas d'une observation de la topographie par télédétection seulement (altimétrie de type SWOT mais aussi plus simplement LIDAR terrestre). En effet, dans de telles conditions, la topographie n'est accessible qu'au-dessus des plus basses eaux, ce qui revient à mettre un seuil de visibilité η'_{vis} sur la topographie dépouillée de sa tendance (Figure 4.6 en haut à droite ou en bas à droite). Le volume $V(\eta'_{\text{vis}})$ n'étant pas accessible, le volume apparent sous le seuil est $V_{\text{app}}(\eta') = V(\eta') - V(\eta'_{\text{vis}})$ ce qui conduit à une estimation biaisée par défaut du rayon hydraulique équivalent, $R_{h,\text{app}}^*$. Sur la Figure 4.7, la courbe $R_{h,\text{app}}^*(\eta')$ est tracée en rouge, elle est construite avec les données LIDAR du RGE ALTI 1m (après retrait de la tendance), par opposition à la courbe bleue construite avec la bathymétrie complète présentée au Chapitre 3. Nous voyons que le seuil de visibilité du RGE ALTI se situe autour de $\eta'_{\text{vis}} = 6$ m, qui correspond au minimum de z'_b dans le RGE. Au contraire, cette valeur minimale en lit mineur obtenue par l'interpolation bathymétrique se situe autour de $\eta' = 2.8$ m. L'estimation du rayon hydraulique équivalent est bien sûr affectée par la présence d'un seuil de visibilité : le rayon hydraulique apparent de plein-bord est ainsi estimé à 2.09 m sur le RGE. En revanche, il est intéressant de constater que

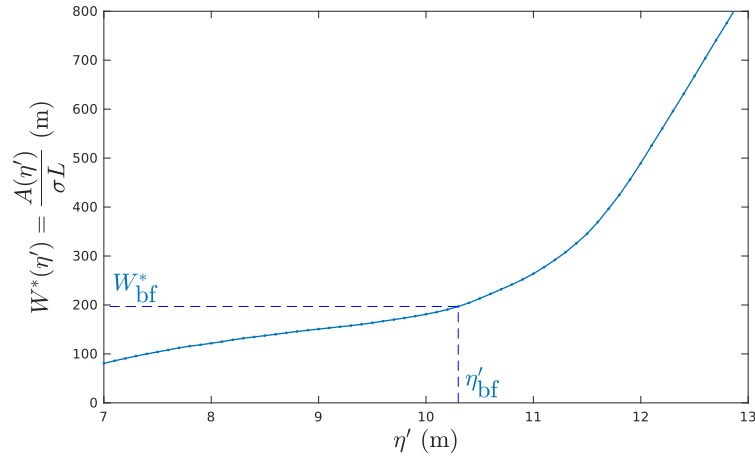


Figure 4.8: Estimation de la largeur équivalent de plein-bord sur le tronçon de la Garonne.

l'estimation de la cote de plein-bord η'_{bf} n'est pas drastiquement impactée : 10.9 m sur le RGE, contre 10.3 m sur la bathymétrie complète. En d'autres terme, la présence du seuil de visibilité change les valeurs de $R^*_h(\eta')$, mais pas radicalement la position du maximum. Nous pouvons donc partir du principe qu'un niveau de plein-bord η'_{bf} est une signature identifiable même sur une topographie possédant un seuil de visibilité.

4.4 Signatures spécifiques à la topographie en lit majeur

4.4.1 Principe et intérêt

Dans certaines approches hydrauliques, comme la décomposition en lits composés telle qu'utilisée dans la formulation DEBORD (*Nicollet et Uan, 1979*), on isole spécifiquement la section mouillée, le périmètre mouillé et le rayon hydraulique en lit majeur, en considérant les écoulements en lit majeur rive gauche, lit majeur rive droite et lit mineur comme relativement indépendants et échangeant de la quantité de mouvement. Nous pouvons reprendre le concept de rayon hydraulique en lit majeur en lui appliquant le même changement d'échelle que le rayon hydraulique classique à l'échelle du tronçon.

4.4.2 Rayon hydraulique équivalent du lit majeur

Les grandeurs nécessaires au calcul de $R^*_{h,fp}(\eta')$ (pour *floodplain*) sont illustrées sur la Figure 4.9 ; pour des raisons de lisibilité de la figure, nous utilisons une topographie synthétique, mais les éléments extraits sur la topographie réelle sont identiques :

- l'aire inondée en lit majeur, obtenue en retranchant l'aire inondée à plein-bord : $A_{fp}(\eta') = A(\eta') - A_{bf}$
- le volume inondé en lit majeur, $V_{fp}(\eta')$, obtenu en retranchant au volume stocké total $V(\eta')$ le volume stocké dans le chenal depuis le fond jusqu'à la cote η' (et non pas seulement jusqu'à la cote de plein-bord η'_{bf}).

Clairement, le seuil de visibilité étant situé bien en-dessous du niveau de plein-bord, il n'affecte pas la qualité de l'estimation de la courbe signature $R^*_{h,fp}(\eta')$, qui ne dépend que de la résolution de la topographie en lit majeur acquise par télédétection.

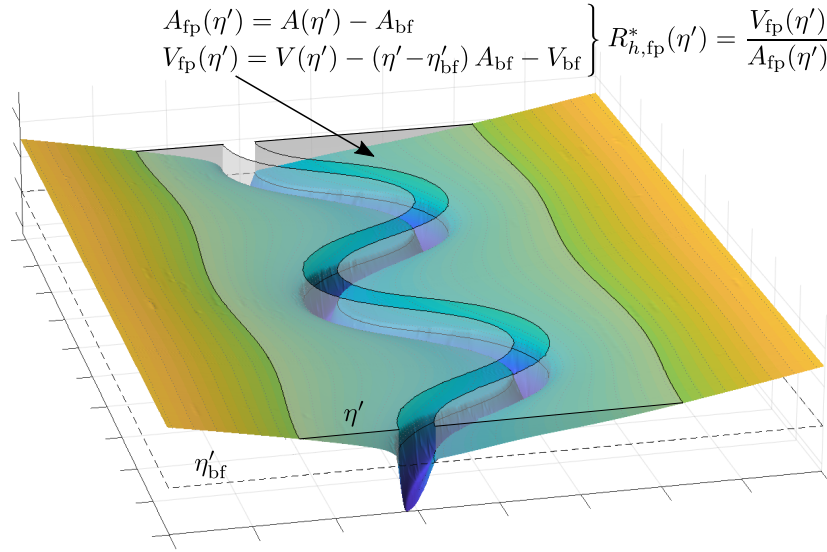


Figure 4.9: Méthode de calcul du rayon hydraulique équivalent en lit majeur spécifiquement. $R_{h,fp}^*$ est toujours défini comme le ratio d'un volume sur une surface inondée, mais en ne considérant que ceux en lit majeur.

4.5 Signatures hydrauliques en écoulement stationnaire

Nous avons construit la plupart des signatures topographiques précédentes en balayant des valeurs de η' , un seuil de cote uniforme appliqué sur la topographie dépouillée de sa tendance. Dans des conditions d'écoulement, la surface libre ne va pas correspondre à un plan parallèle à la tendance topographique régionale, c'est-à-dire qu'elle va différer d'un plan horizontal $\eta' = \text{constante}$ dans la topographie sans tendance : elle va définir une surface $\eta'(x, y, t)$ (voir la Figure 4.10). Cette surface ne sera un plan $\eta' = \text{constante}$ que pour une géométrie uniforme et un régime permanent. Il faut donc maintenant définir des signatures hydrauliques, indicateurs synthétisant certains aspects du fonctionnement du tronçon en présence d'un écoulement.

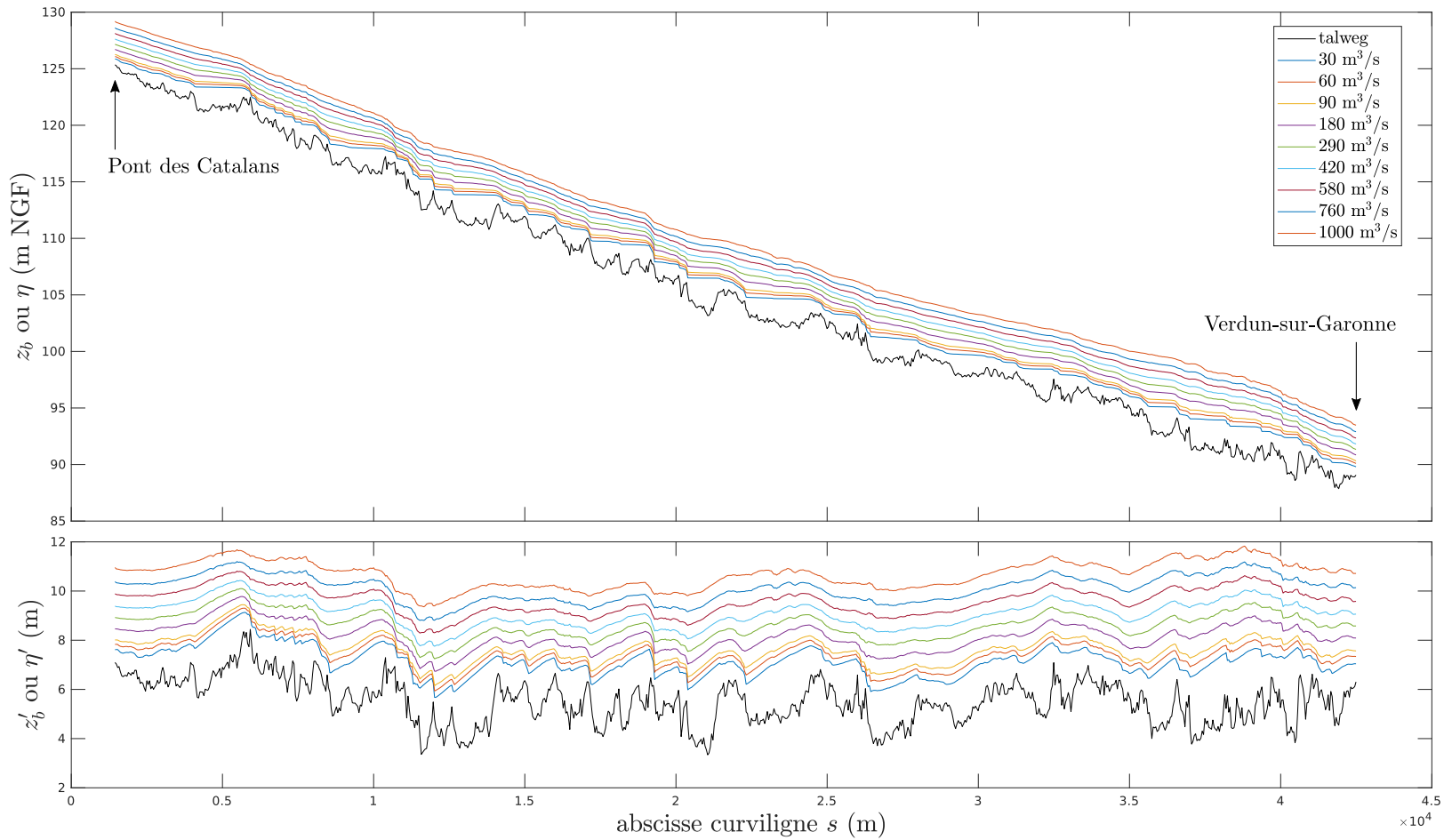


Figure 4.10: Profils en long de la surface libre extraits des états stationnaires obtenus pour le calage du coefficient de Manning en lit mineur. En haut, les profils sont représentés en cote normale NGF (variable z_b ou η), tandis qu'en bas a retiré la composante tendancielle de la topographie de la plaine d'inondation (variable z'_b ou η').

4.5.1 Signatures relatives à la cote de la surface libre

La première signature est $\frac{\tilde{A}(Q)}{\sigma L}$ qui est équivalente à une largeur au miroir au tronçon et qui désigne l'aire inondée par un écoulement stationnaire en fonction du débit Q par unité de longueur développée par le chenal (σL).

\tilde{A} est à distinguer de la signature purement topographique $A(\eta')$, qui est simplement obtenue en appliquant un seuil uniforme sur la topographie sans tendance. La figure 4.11 en haut donne un exemple l'aire inondée pour un débit donné.

La deuxième signature relative à la cote de la surface libre est $\sigma_{\eta'}(\bar{\eta}')$ qui représente l'écart type de la cote de la surface libre $\eta'(x, y)$ d'un écoulement stationnaire sans tendance régionale en fonction de sa valeur moyenne au tronçon $\bar{\eta}'(Q)$ (Figure 4.11 en bas). Elle permet de quantifier le comportement qualitatif identifié sur les lignes d'eau : l'écart-type de la cote $\eta'(x, y)$ diminue lorsque l'on se rapproche des conditions de plein-bord, la différence de comportement hydraulique entre mouilles et radiers s'estompant à mesure que le débit, et donc la cote moyenne $\bar{\eta}'$, augmente (Figure 4.12).

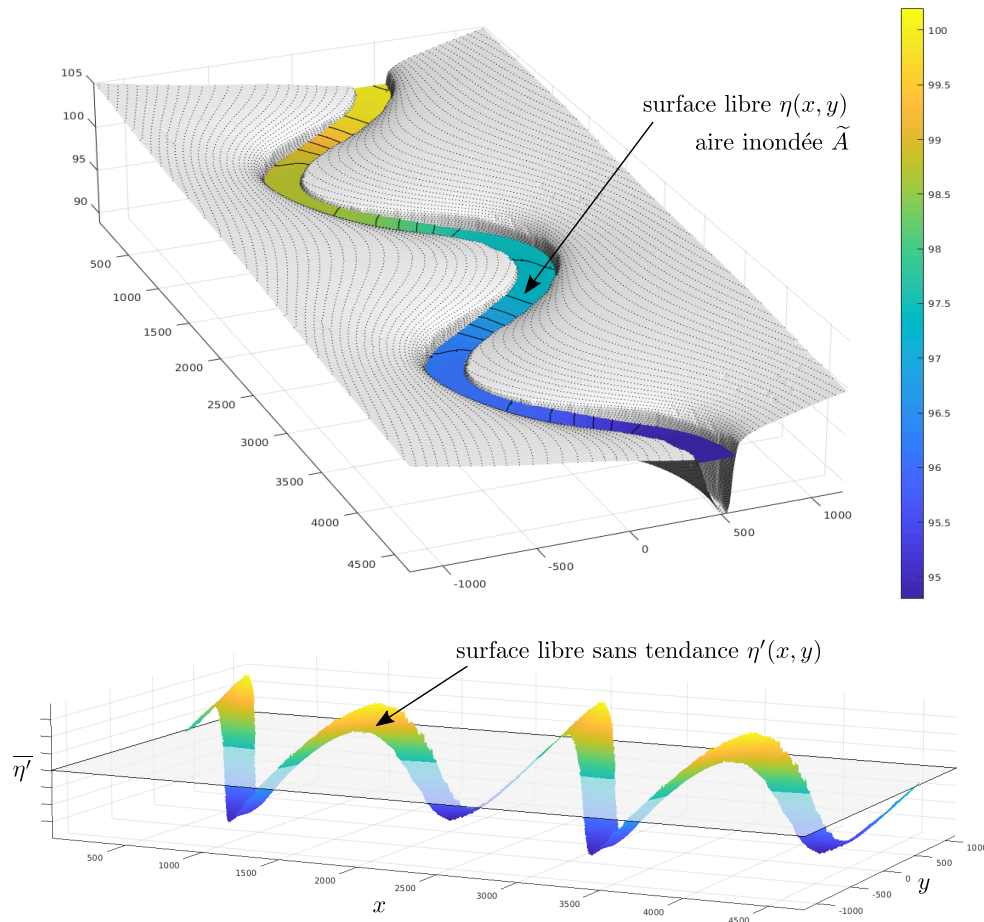


Figure 4.11: Calcul de la cote sans tendance de la surface libre η' d'un écoulement, en retirant la tendance régionale de la topographie.

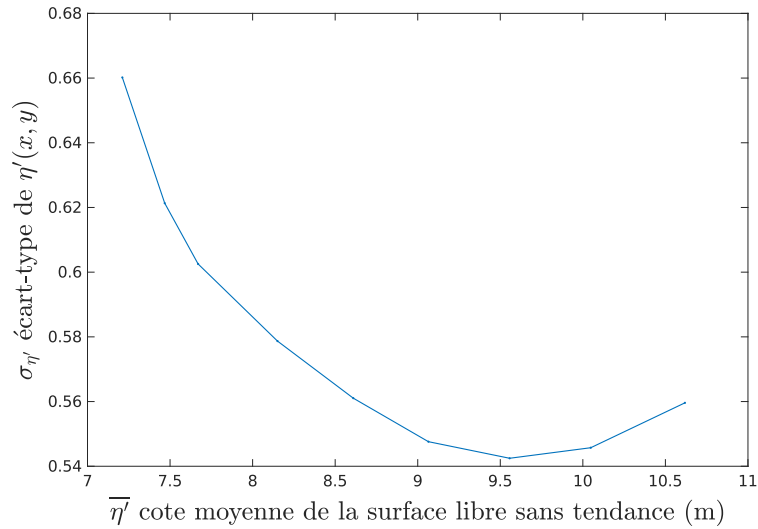


Figure 4.12: Signature $\sigma_{\eta'}(\bar{\eta}')$, courbe reliant l'écart-type de la cote sans tendance $\eta'(x, y)$ de la surface libre d'un écoulement stationnaire à la moyenne de cette cote, $\bar{\eta}'$. Cette signature est ici limitée à la gamme de débits non-débordants simulés pour le calage du coefficient de Strickler en lit mineur.

4.5.2 Signature hydrométrique : courbe de tarage au tronçon

Finalement, la signature hydraulique qui nous intéressera principalement pour l'estimation des débits est la courbe de tarage au tronçon, qui relie le débit sortant du tronçon à la cote moyenne sans tendance $\bar{\eta}'$. Pour que cette courbe soit une relation univoque, elle ne peut être en toute rigueur définie qu'en régime permanent. Pendant un événement de crue, le maximum de débit et le maximum de cote de la surface libre ne sont pas synchrones à une section donnée : la relation hauteur-débit va alors présenter un hystérésis entre la crue et la décrue (voir par exemple, *Muste et al.*, 2020), comme illustré sur la Figure 4.13. Cet effet va bien entendu être amplifié en passant de l'échelle de la section à l'échelle du tronçon, le décalage entre le maximum temporel de la moyenne spatiale $\bar{\eta}'$ et le maximum temporel de Q en sortie du tronçon étant encore accentué. L'objectif de cette thèse se limite cependant à la recherche d'une relation univoque, et en analysant par la suite les états hydrauliques observés pendant un événement de crue comme une succession d'états quasi-stationnaires. La pertinence de cette hypothèse devra bien entendu être évaluée en la testant sur des états hydrauliques extraits des simulations transitoires effectuées sur la Garonne.

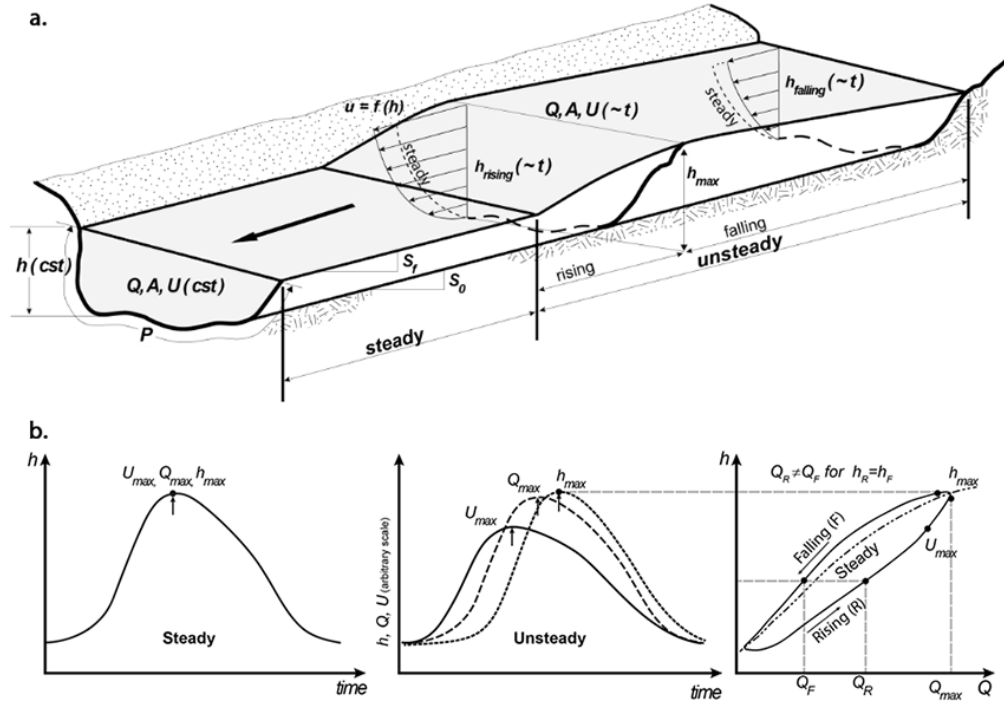


Figure 4.13: Illustration du caractère non-univoque de la relation hauteur-débit à la section (extrait de Muste, et al. 2020). Pour une crue très lente, les maxima des variables de hauteur et de débit sont synchrones (courbe en bas à gauche). Pour un évènement plus rapide, la maximum de débit est atteint avant le maximum de hauteur (courbes en bas au milieu), ce qui provoque une boucle d'hystérésis en variables (Q, h) (courbe en bas à droite) avec un débit plus fort en montée de crue qu'en décrue pour une même hauteur observée.

4.6 Tableau synthétique des signatures définies

Le Tableau 4.1 ci-dessous résume les différentes signatures définies dans ce chapitre. Notons que la relation débit-cote moyenne de la surface libre sans tendance est une bijection et par conséquent : $Q(\bar{\eta}') = \bar{\eta}'(Q)$. Intuitivement, les signatures s'écrivent en fonction du débit Q , mais pour une signature donnée on peut toujours faire le type de transformation :

$$\tilde{A}(Q) = \tilde{A}(Q(\bar{\eta}')) = \tilde{A}(\bar{\eta}'(Q)) = \tilde{A}(\bar{\eta}')$$

Nous pouvons appliquer cette équivalence pour toutes les signatures. Ainsi, exprimer des signatures en fonction du débit est équivalent à les exprimer en fonction de $\bar{\eta}'$. On garde dans la suite les signatures en fonction de $\bar{\eta}'$, qui est la variable observable par télédétection.

Signature		Définition	Type
Forme en plan	σ	Sinuosité	valeur unique
	σ_θ	Ecart-type angulaire	valeur unique
	λ	Longueur d'onde des alternances de faciès hydrauliques	valeur unique
Topographie générale	I_{reg}	Pente régionale du lit majeur (m/m)	valeur unique
	$A(\eta')$	Aire inondée plane pour le niveau uniforme η' de topographie sans tendance	courbe
	$V(\eta')$	Volume inondé pour le niveau uniforme η' de topographie sans tendance	courbe
	$R_h^*(\eta') = \frac{V(\eta')}{A(\eta')}$	Rayon hydraulique équivalent pour le niveau uniforme η' de topographie sans tendance	courbe
	η'_{bf}	Niveau η' maximisant $R_h^*(\eta')$	valeur unique
	$W_{\text{bf}}^* = \frac{A(\eta'_{\text{bf}})}{\sigma_L}$	Aire inondée à plein-bord par unité de longueur développée de chenal \Rightarrow largeur équivalente de plein-bord	valeur unique
Topographie en lit majeur (<i>floodplain</i>)	$A_{\text{fp}}(\eta')$	Aire inondée plane en lit majeur pour le niveau η'	courbe
	$V_{\text{fp}}(\eta')$	Volume inondé en lit majeur pour le niveau η'	courbe
	$R_{h,\text{fp}}^*(\eta') = \frac{V_{\text{fp}}(\eta')}{A_{\text{fp}}(\eta')}$	Rayon hydraulique équivalent en lit majeur pour le niveau η'	courbe
Ecoulement stationnaire	$\tilde{A}(\bar{\eta}')$	Aire inondée ($\eta(x, y) > z_b(x, y)$) fonction de la valeur moyenne $\bar{\eta}'$	courbe
	$\sigma_{\eta'}(\bar{\eta}')$	Ecart-type de $\eta'(x, y)$ fonction de sa valeur moyenne $\bar{\eta}'$	courbe
	$Q(\bar{\eta}')$	Courbe de tarage au tronçon : débit aval fonction de la moyenne $\bar{\eta}'$	courbe

Table 4.1: Tableau récapitulatif des différentes signatures (valeurs uniques ou courbes) définies dans ce chapitre.

Chapitre 5

Modèle équivalent stationnaire en géométrie périodique : formulation et premiers tests d'inversion

5.1 Introduction

Au chapitre précédent, nous avons défini un certain nombre de signatures topographique et hydrauliques synthétisant certaines propriétés à l'échelle du tronçon, calculables à partir d'observations et notamment, pour certaines, d'observations par télédétection. Bien sûr, ces signatures peuvent être calculées sur des résultats de simulation produites par un modèle hydraulique direct du tronçon étudié. Dans ce chapitre, nous allons tenter de répondre à deux questions :

- Quel est le modèle de complexité minimale permettant de reproduire la plupart des signatures définies précédemment, et notamment des signatures hydrauliques $\tilde{A}(\overline{\eta'})$, $\sigma_{\eta'}(\overline{\eta'})$ et $Q(\overline{\eta'})$?
- Ce modèle minimal défini, la signature $Q(\overline{\eta'})$ (courbe de tarage au tronçon) peut-elle être déduite indirectement sur un tronçon non-jaugé, en exploitant l'information portée par d'autres signatures pour identifier les paramètres spécifiant ce modèle direct minimal ? En d'autres termes, le modèle minimal peut-il aider à résoudre le problème inverse hauteur-débit constitué par l'estimation de débits à l'aide de mesures altimétriques par télédétection ?

L'approche proposée repose sur la notion de tronçon équivalent ou géométrie équivalente, telle que le modèle stationnaire uniforme proposé par *Rebolho et al.* (2018) ou *Zheng et al.* (2018). Les motivations pour complexifier cette géométrie équivalente uniforme viennent de la première partie : nous faisons l'hypothèse qu'il existe des corrélations exploitables entre la forme en plan du cours d'eau, observable par télédétection, et la variabilité longitudinale de la bathymétrie. Ces hypothèses conduisent à construire une géométrie équivalente qui n'est plus uniforme, mais périodique, tout en conservant l'hypothèse de stationnarité. L'avantage de l'approche périodique est qu'elle permet, comme l'approche uniforme, de s'affranchir de toute condition à la limite, tout en permettant de représenter une certaine variabilité interne au tronçon, et donc le calcul de certaines signatures.

La première étape de la construction de ce modèle équivalent consiste à paramétriser la forme en plan (qui correspondra à la ligne de talweg ou profondeur maximale dans chaque section). Une fois fixé le cadre géométrique dans le plan, nous proposerons une paramétrisation de la bathymétrie des sections en lit mineur et de la topographie en lit majeur. Enfin, nous testerons la possibilité d'inverser les paramètres de ce modèle uniquement sur la base d'observations de la surface libre et de la topographie au dessus d'un seuil

de visibilité, correspondant au plus bas niveau de la surface libre, en utilisant des signatures hydrauliques calculées à l'échelle du tronçon.

5.2 Construction du modèle équivalent périodique d'un tronçon

5.2.1 Paramétrisation de la forme en plan

La description et la modélisation du tracé en plan des cours a fait l'objet de très nombreuses études depuis les travaux fondateurs de *Leopold et Langbein* (1966). Ils montrent que, pour une sinuosité donnée (prescrite à plus large échelle par les contrôles morphologiques), la forme locale du tracé est loin d'être aléatoire. Le cours d'eau tend à minimiser la variance angulaire permettant de réaliser la sinuosité imposée, de façon analogue à une lame d'acier-ressort dont la position est fixée en 2 points (distance euclidienne d_{AB} imposée), mais pouvant par ailleurs se déformer librement entre ces deux points (Figure 5.1). Un autre exemple plus dramatique cité par Leopold et Langbein est celui de l'accident de train de Greenville, survenu le 31 mai 1965 : un train de marchandises transportant entre autres une cargaison de rails de chemins de fer en acier a percuté un bulldozer, provoquant le déraillement du convoi (Figure 5.2). Sur toute la longueur du convoi occupée par le chargement de barres d'acier (structuré en 'fagots' de 200 mètres de longueur et répartis sur la longueur 30 wagons), la violente contrainte compressive provoquée par la décélération brutale a forcé un raccourcissement dans la direction de déplacement. Du fait des propriétés élastiques de l'acier, le chargement de barres a adopté une configuration conservant la longueur développée initiale, mais minimisant la courbure totale selon une forme similaire à une série de méandres.

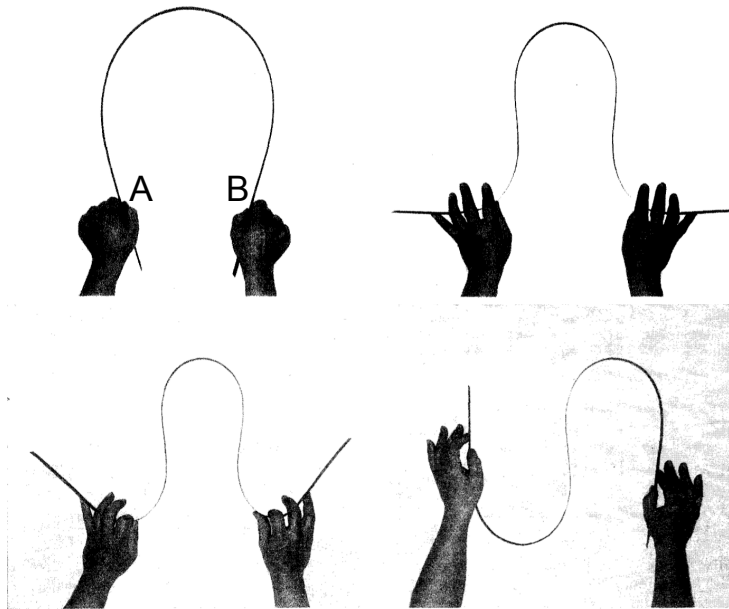


Figure 5.1: Source : *Leopold et Langbein* (1966)



Figure 5.2: Photos de l'accident ferroviaire de Greenville (Caroline du Sud), le 31 mai 1965. Suite au déraillement en tête de convoi (photo de droite), le chargement de barres d'acier situé plus à l'arrière a adopté une configuration conservant sa longueur initiale mais minimisant la courbure totale, suivant une courbe de type méandre (photo de gauche).

5.2.1.1 Paramétrisation : modèle de Kinoshita à 3 paramètres

Selon *Leopold et Langbein* (1966), la courbe minimisant la variance angulaire pour une sinuosité donnée est la *courbe générée par sinus* (*sine-generated curve*), dans laquelle le vecteur unitaire tangent a pour angle

$$\theta(s) = \theta_0 \sin\left(\frac{2\pi}{\lambda}s\right)$$

où θ_0 est un paramètre représentant la déviation angulaire maximale par rapport à la direction moyenne, et λ la longueur d'onde des méandres mesurée selon les abscisses curvilignes. Cette équation revient à dire que le vecteur tangent unitaire est :

$$\mathbf{T}(s) = e^{i\theta(s)} = e^{i\theta_0 \sin\left(\frac{2\pi}{\lambda}s\right)}$$

Il s'agit donc d'une courbe périodique, répétition à l'identique d'un même motif. Il n'existe pas d'expression littérale pour $\mathbf{z}(s)$ la courbe elle-même dans le plan complexe, il faut donc procéder numériquement par intégration de la fonction (à valeurs complexes mais d'une variable réelle) $\mathbf{T}(s)$. Dans notre modèle, nous utilisons une formulation légèrement plus générale pour la forme en plan, proposée par Kinoshita (*Kinoshita*, 1961) :

$$\theta(s) = \theta_0 \sin(k_0 s) + \alpha_s \theta_0^3 \cos(3k_0 s) - \alpha_f \theta_0^3 \sin(3k_0 s)$$

où l'on note $k_0 = \frac{2\pi}{\lambda}$ le nombre d'onde du motif en $\text{rad}\cdot\text{m}^{-1}$. L'introduction de deux nouveaux paramètres, l'asymétrie α_s (*skewness*) et l'aplatissement α_f (*flattening*), permet de représenter une plus grande diversité de formes, comme celle illustrée sur la Figure 5.3.

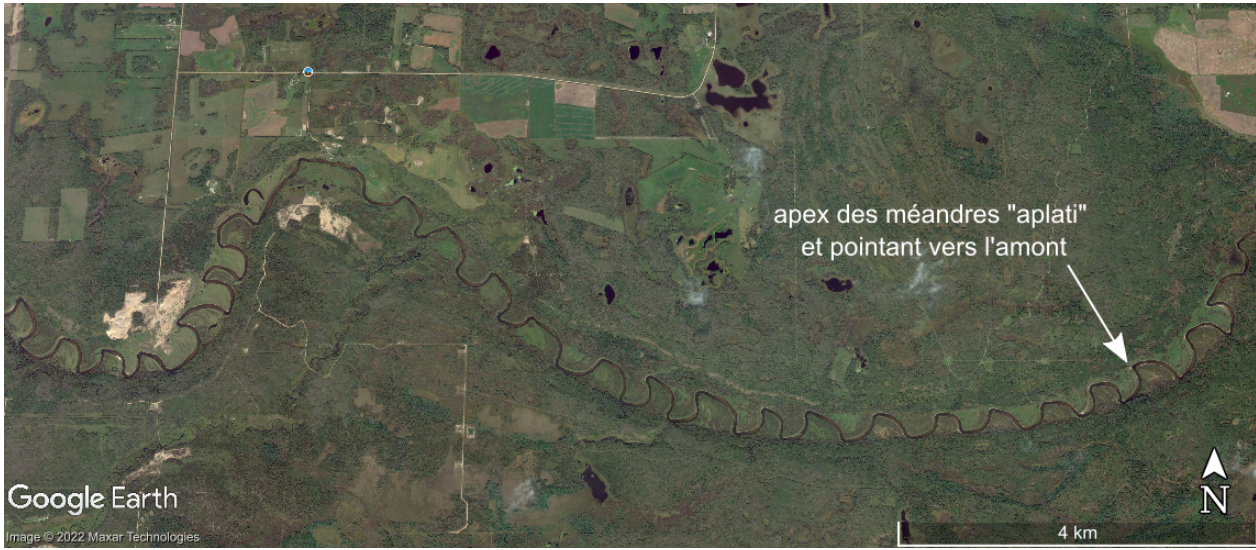


Figure 5.3: Méandres sur la Beaver River (Alberta / Saskatchewan) : on note une tendance des méandres à l'asymétrie (*skewness*) et à l'aplatissement (*flattening*).

Par définition du vecteur unitaire tangent à la ligne de talweg, on a :

$$\frac{dz}{ds} = \mathbf{T}(s) = e^{i\theta(s)}$$

et donc

$$\mathbf{z}(s) = \int_0^s e^{i\theta(s)} ds$$

La courbe de Kinoshita peut donc être construite simplement en intégrant par transformée de Fourier $\mathbf{T}(s)$, selon les formules inverses de celles de dérivation utilisées au chapitre précédent. Nous procédons également à rebours du calcul de \mathbf{T} ou θ à partir de \mathbf{z} , en commençant par construire la grandeur sans dérive $\mathbf{z}'(s)$ (divagation locale) telle que :

$$\mathcal{F}[\mathbf{z}'](k) = \begin{cases} \frac{1}{ik} \mathcal{F}[\frac{dz}{ds}](k) & \text{pour } k \neq 0 \\ 0 & \text{pour } k = 0 \text{ (valeur moyenne nulle)} \end{cases}$$

On obtient $\mathbf{z}'(s)$ par transformée inverse : il s'agit d'une courbe fermée périodique (voir Figure 5.4) analogue à celle présentée pour la Garonne (Figure 4.2). Il reste, pour finir, à rajouter la tendance linéaire en s (terme en $k = 0$ dans la dérivée), qui est la moyenne sur une période du vecteur \mathbf{T} à laquelle on a accès dans la transformée de Fourier rapide (FFT) :

$$\langle \mathbf{T} \rangle = \frac{1}{\lambda} \int_0^\lambda e^{i\theta(s)} ds = \frac{1}{\lambda} \int_0^\lambda \cos(\theta(s)) ds + i \frac{1}{\lambda} \int_0^\lambda \sin(\theta(s)) ds$$

On peut noter que $\theta(s + \frac{\lambda}{2}) = -\theta(s)$ et donc que $\sin(\theta(s + \frac{\lambda}{2})) = -\sin(\theta(s))$: la fonction $\sin(\theta(s))$ est donc de moyenne nulle, il n'y a pas de dérive dans la direction y . La dérive est uniquement selon la direction x et on peut l'exprimer en fonction de la sinuosité σ :

$$\langle \mathbf{T} \rangle = \frac{1}{\lambda} \int_0^\lambda \left(\frac{dx}{ds} \right) ds = \frac{1}{\lambda} [x(s)]_{s=0}^{s=\lambda} = \frac{1}{\lambda} \frac{\lambda}{\sigma} = \frac{1}{\sigma}$$

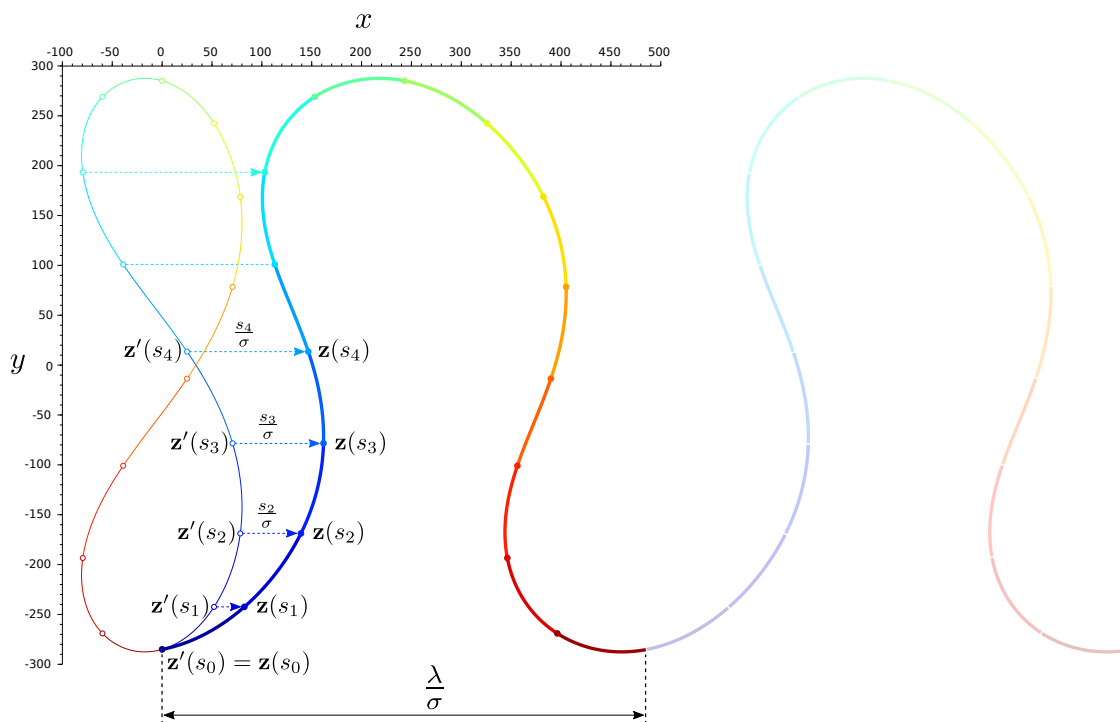


Figure 5.4: Construction de la courbe de Kinoshita : on commence par construire la courbe “en 8” périodique $\mathbf{z}' * (s)$, puis on ajoute la dérive linéaire en s : $\mathbf{z}(s) = \mathbf{z}'(s) + \frac{s}{\sigma}$. Sur une période λ mesurée en suivant le talweg, le déplacement selon l’axe des x est donc $\Delta x = \frac{\lambda}{\sigma}$.

et donc finalement

$$\mathbf{z}(s) = \mathbf{z}'(s) + \frac{s}{\sigma}$$

5.2.2 Bathy-topographie

5.2.2.1 Construction de coordonnées conformes (Φ)

Considérons le système de coordonnées cartésiennes (x, y) avec l’axe x aligné selon la direction moyenne (régionale) du cours d’eau, et l’axe y orienté par conséquent dans la direction perpendiculaire. Considérons également la topographie à modéliser ‘dépouillée’ de sa tendance régionale, c’est-à-dire plate en moyenne. Ces coordonnées sont très peu pratiques pour paramétriser la topographie à la fois en lit mineur (chenal) mais également en lit majeur : sauf dans le cas particulier d’un chenal rectiligne, la coordonnée y ne représente pas la distance à la ligne de talweg. Au contraire, nous aimerions précisément disposer d’un système de coordonnées dont l’une serait toujours nulle pour un point situé sur la ligne de talweg, mais qui se comporterait asymptotiquement comme la coordonnée y loin du chenal en rive droite ou en rive gauche. La Figure 5.5 (panneau de gauche) illustre cette propriété : la coordonnée y est illustrée par le plan semi-transparent, tandis que la coordonnée Φ souhaitée est illustrée en couleur. Dans cette coordonnée transverse conforme, la ligne de talweg est une équipotentielle “rugueuse” (*Vandembroucq et Roux, 1997*) définie par $\Phi = 0$. Une topographie symétrique et périodique en lit majeur pourrait alors être une simple fonction de $|\Phi|$ (Figure 5.5, panneau de droite). Dans un but de concision, la construction de ces coordonnées conformes n’est pas détaillée dans le manuscrit ; elle est basée sur la méthode des éléments analytiques (par exemple *Strack, 2017*).

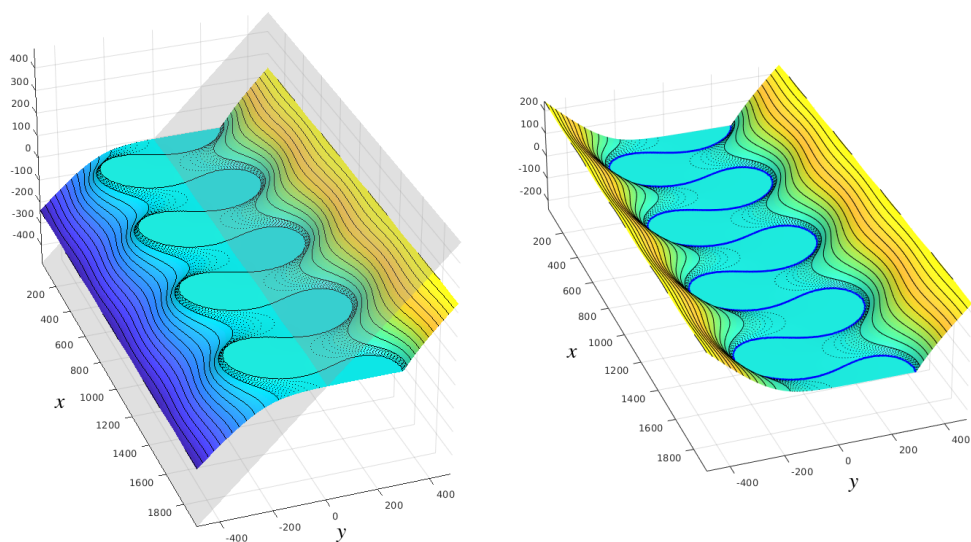


Figure 5.5: Principe de construction d'un jeu de coordonnées conformes adaptées à la géométrie du chenal. La coordonnée cartésienne transverse y présente l'inconvénient de ne pas être constante le long du chenal. Nous construisons donc un jeu de coordonnées conformes dont la composante transverse Φ est constante le long du chenal (i.e., le chenal constitue une équipotentielle de la surface $\Phi(x, y)$ en couleur sur le panneau de gauche). La topographie équivalente en lit majeur peut alors être définie comme une fonction de $|\Phi|$ seulement (panneau de droite).

En particulier, nous choisissons ce cadre géométrique car il semble bien adapté à la modélisation statistique de la topographie des barres de méandres, déposées par accretion latérale de sédiments dans la partie interne des méandres (cf. Figure 5.6). Ces structures vont déterminer la topographie juste au-dessus du seuil de débordement (débit / niveau de plein bord), et définir une transition importante dans la courbe de tarage.



Figure 5.6: Barres de méandres nettement visibles sur la Beaver River, du fait de la couverture végétale très rase (les flèches indiquent la direction de construction de la barre). Ces zones étant les premières à être submergées au-delà du régime de plein-bord, la représentation fine de leur hypsométrie (au niveau statistique) est importante.

5.2.2.2 Fonction largeur de plein-bord $W_{\text{bf}}(s)$, demi-largeurs de plein-bord rive gauche $W_{\text{bf}}^{(l)}(s)$ et rive droite $W_{\text{bf}}^{(r)}(s)$

Nous avons vu au Chapitre 2 que la largeur de plein-bord co-varie avec d'autres variables géométriques comme la profondeur maximale à la section. Dans le modèle périodique proposé, nous utiliserons une largeur de plein-bord uniforme, qui sera fixée à celle identifiée d'après le maximum de rayon hydraulique équivalent au tronçon R_h^* . Pour la Garonne, nous fixons donc

$$W_{\text{bf}}(s) = W_{\text{bf}}^* = \frac{A(\eta'_{\text{bf}})}{\sigma L} \quad \forall s$$

Cependant, la position en plan de la ligne de talweg (i.e. de profondeur maximale) étant fixée par le modèle de Kinoshita, il n'est pas très réaliste de considérer que les sections en travers sont symétriques par rapport à cette ligne. Si la fonction $W_{\text{bf}}(s)$ peut raisonnablement être considérée comme constante, il n'en est pas de même pour la distance séparant la ligne de talweg de la ligne de plein-bord rive gauche (notée $W_{\text{bf}}^{(l)}(s)$ pour *left*) et la distance séparant cette même ligne de talweg de la ligne de plein-bord rive droite (notée $W_{\text{bf}}^{(r)}(s)$ pour *right*). Dans les zones de forte courbure comme les apex de méandres, la pente des berges est en effet bien plus forte dans la partie externe que dans la partie interne, et la demi-largeur est donc bien plus grande du côté de la berge vers laquelle le chenal s'incurve. Cette asymétrie est illustrée sur la Figure 5.7.

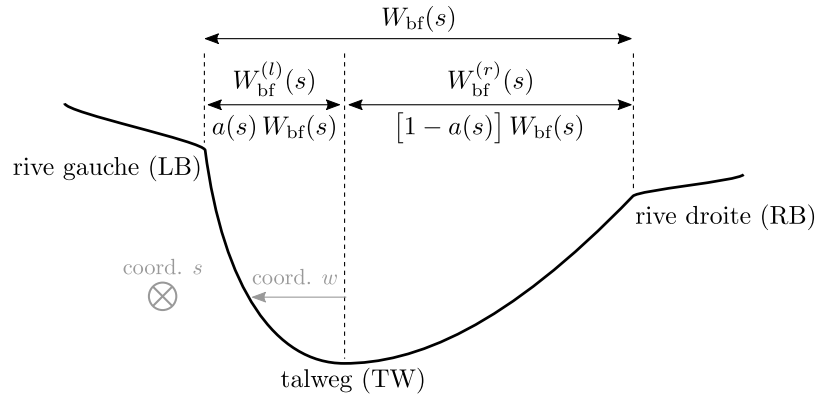


Figure 5.7: Notations utilisées pour la description de la géométrie des sections en travers. L'abscisse curviligne s augmente le long de la ligne de talweg dans la direction perpendiculaire au plan de la figure, tandis que la coordonnée w varie perpendiculairement à cette ligne de talweg. A chaque section positionnée par s , on définit une demi-largeur à plein-bord en rive gauche, $W_{\text{bf}}^{(l)}$, et une demi-largeur en rive droite, $W_{\text{bf}}^{(r)}$. La demi-largeur rive gauche représente une fraction $a(s)$ de la largeur totale, et la demi-largeur rive droite la fraction complémentaire $(1 - a(s))$.

La paramétrisation choisie ici s'inspire de celle proposée par *Deutsch et Wang* (1996), dans laquelle chaque demi-largeur est fonction de la courbure locale (signée) $\gamma(s)$ de la ligne de talweg :

$$\begin{cases} W_{\text{bf}}^{(l)}(s) = a(s)W_{\text{bf}}^* = \frac{W_{\text{bf}}^*}{2} \left(1 + \frac{\gamma(s)}{\gamma_{\text{max}}} \right) \\ W_{\text{bf}}^{(r)}(s) = (1 - a(s))W_{\text{bf}}^* = \frac{W_{\text{bf}}^*}{2} \left(1 - \frac{\gamma(s)}{\gamma_{\text{max}}} \right) \end{cases}$$

où $\gamma_{\text{max}} = \max_s \{|\gamma|\}$ est le maximum en valeur absolue sur tout le chenal de la courbure $\gamma(s)$. L'inconvénient de cette paramétrisation est qu'il y a toujours une section pour laquelle l'une des demi-largeur est nulle, et

de même si le chenal présente une très faible sinuosité. En effet, si $\gamma(s) = +\gamma_{\max}$, alors la demi-largeur rive droite est nulle ($a(s) = 1$ et $(1 - a(s)) = 0$, la berge droite est verticale), et inversement si $\gamma(s) = -\gamma_{\max}$ c'est la demi-largeur rive gauche qui est nulle et la berge gauche qui est verticale ($a(s) = 0$ et $(1 - a(s)) = 1$). La simulation de berges très abruptes en rive externe des méandres à forte courbure est tout à fait réaliste, mais ce comportement n'est pas souhaitable pour des chenaux dont la courbure maximale n'est pas très élevée. Nous pouvons donc réécrire le modèle de Deutsch et Wang en introduisant une valeur maximale pour $a(s)$, paramètre noté $a_{\max} < 1$:

$$\begin{cases} W_{\text{bf}}^{(l)}(s) = a(s)W_{\text{bf}}^* = W_{\text{bf}}^* \left[\frac{1}{2} + \frac{\gamma(s)}{\gamma_{\max}} \left(a_{\max} - \frac{1}{2} \right) \right] \\ W_{\text{bf}}^{(r)}(s) = (1 - a(s))W_{\text{bf}}^* = W_{\text{bf}}^* \left[\frac{1}{2} - \frac{\gamma(s)}{\gamma_{\max}} \left(a_{\max} - \frac{1}{2} \right) \right] \end{cases}$$

De cette façon, chaque demi-largeur est bornée dans l'intervalle $[1 - a_{\max}; a_{\max}]$. L'idée est ensuite d'éliminer le degré de liberté a_{\max} en en faisant une fonction déterministe de la courbure maximale de la ligne de talweg. Un second avantage de la coordonnée transverse conforme Φ présentée précédemment, outre le fait que le chenal soit défini par l'isovaleur $\Phi = 0$, est qu'elle est également très bien adaptée pour quantifier ces effets de courbure non pas en termes relatifs, mais bien en termes absolus.

Pour comprendre comment est calculé a_{\max} , prenons l'exemple du chenal de sinuosité moyenne illustré en haut à droite sur la Figure 5.8, avec une largeur de plein-bord souhaitée $W_{\text{bf}}^* = 200$ m (il s'agit des paramètres de sinuosité et de largeur de plein-bord du tronçon de la Garonne). Sur cette figure, les niveaux de couleurs et les lignes pointillées correspondent à des iso-valeurs de la coordonnée conforme transverse Φ . Nous commençons par identifier la position du maximum de courbure de la ligne de talweg (tracée en blanc, iso-valeur $\Phi = 0$) : la section en travers correspondante est tracée en noir. En s'éloignant du talweg (TW) à la fois vers la rive gauche ($\Phi > 0$) et vers la rive droite ($\Phi < 0$), nous recherchons la valeur absolue Φ_{bf} telle que les courbes iso-valeur $\Phi = +\Phi_{\text{bf}}$ et $\Phi = -\Phi_{\text{bf}}$ soient distantes de la largeur totale de plein-bord W_{bf}^* . Nous obtenons ici le point LB (*left bank*) situé sur l'iso-valeur $\Phi = +\Phi_{\text{bf}} = +40.5$, et de l'autre côté le point RB situé sur l'iso-valeur $\Phi = -\Phi_{\text{bf}} = -40.5$ avec la distance $d(\text{LB}, \text{RB}) = 200$ m. Le paramètre a_{\max} est alors donné par le ratio entre le maximum des 2 demi-largeurs obtenues d'une part, et la largeur totale d'autre part :

$$a_{\max} = \frac{\max\{d(\text{TW}, \text{LB}), d(\text{TW}, \text{RB})\}}{W_{\text{bf}}^*}$$

Nous obtenons dans ce cas médian $a_{\max} = 0.88$, c'est-à-dire que la demi-largeur externe dans les méandres ne sera jamais inférieure à 12% de la largeur totale. Les deux autres cas illustrés sur la figure 5.8 permettent de vérifier le comportement asymptotique pour les très fortes ou très faibles sinuosités :

- pour une très forte sinuosité, comme dans le tracé en haut à gauche, les iso-valeurs de Φ au maximum de courbure sont extrêmement dissymétriques : l'échelle est logarithmique, et l'on voit que les iso-valeurs sont extrêmement rapprochées côté externe et extrêmement espacées côté interne. La valeur Φ_{bf} identifiée est de 0.253, avec l'iso-valeur $\Phi_{\text{RB}} = -0.253$ située à quasiment 200 m du talweg côté interne de la courbure, et l'iso-valeur $\Phi_{\text{LB}} = +0.253$ quasiment confondue avec la ligne de talweg $\Phi = 0$ côté externe. Il en résulte $a_{\max} \simeq 1$, le talweg est donc bien "collé" à la rive externe des méandres pour ce cas de très forte sinuosité, et la totalité du chenal s'étend alors intégralement côté rive interne ;
- en revanche, quand la sinuosité devient très faible (figure en bas), la coordonnée conforme Φ diffère très peu de la coordonnée cartésienne y . Les iso-valeurs Φ_{LB} et Φ_{RB} correspondent alors quasiment avec les valeurs $\pm \frac{W_{\text{bf}}^*}{2} = 100$ m, et l'on obtient $a_{\max} \simeq \frac{1}{2}$: la ligne de talweg se maintient toujours à distance presque égale du bord de berge rive gauche et du bord de berge rive droite. Le modèle initial de Deutsch et Wang aurait produit une berge verticale au maximum de courbure même pour ce chenal de faible sinuosité.

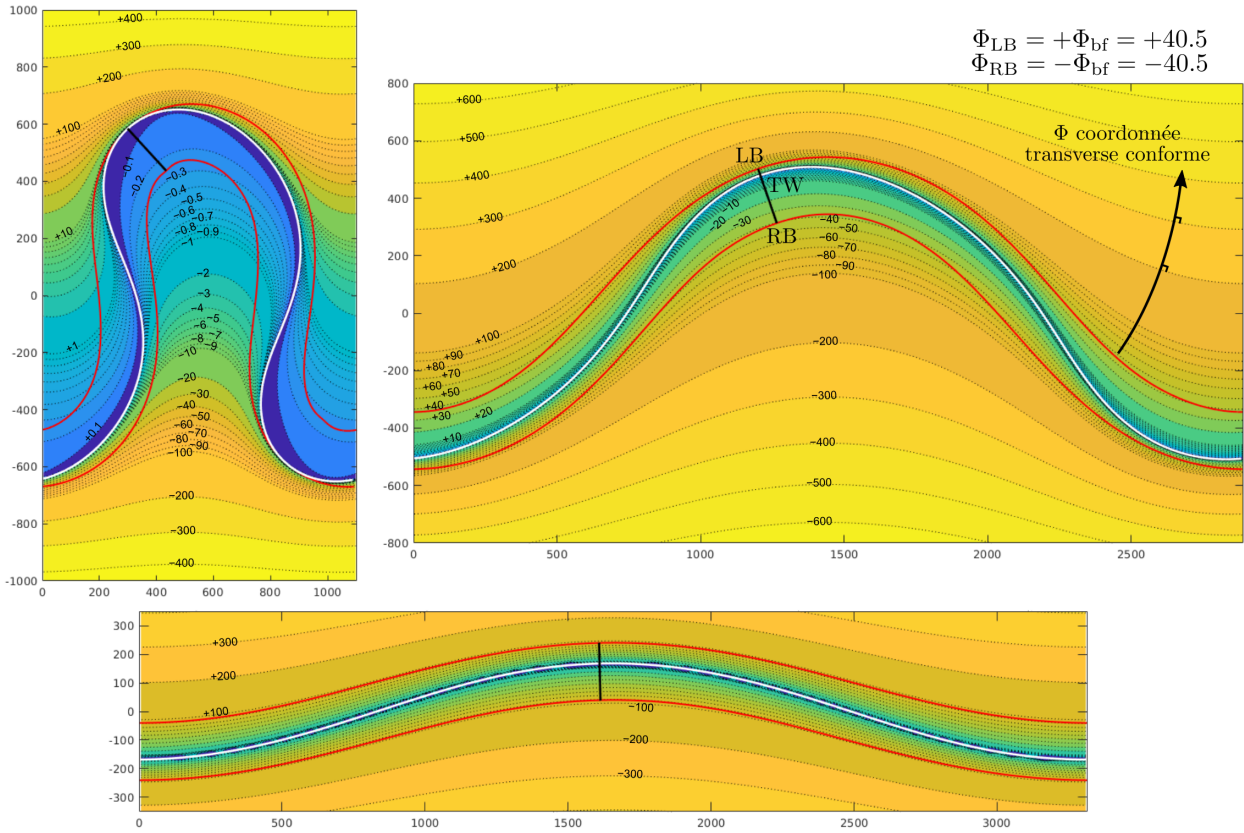


Figure 5.8: Illustration de la méthode de calcul de l'asymétrie maximale des demi-largeurs rive gauche et rive droite. Les niveaux de couleur et les lignes pointillées sont des iso-valeurs de la coordonnée transverse conforme Φ , avec un découpage par décades logarithmiques permettant de visualiser plus finement les iso-valeurs en cas de forte courbure, et une échelle de couleur symétrique pour les valeurs $\Phi > 0$ côté rive gauche et $\Phi < 0$ côté rive droite (la direction d'écoulement étant de gauche vers la droite).

5.2.2.3 Bathymétrie en lit mineur

Deux éléments importants sont à paramétrer pour la bathymétrie en lit mineur :

- la profondeur maximale à chaque section (profondeur de talweg h_{talweg}),
- la forme des sections, en fonction de la coordonnée w distance à la ligne de talweg.

La bathymétrie est donc construite en utilisant les coordonnées (s, w) respectivement selon les vecteurs (\mathbf{T}, \mathbf{N}) , système valable seulement dans un voisinage tubulaire du chenal comme expliqué au chapitre précédent (Figure 4.4). Nous procéderons différemment en lit majeur. De façon à limiter le nombre de degrés de liberté, nous faisons l'hypothèse, forte, que le niveau de plein-bord correspond à un niveau $\eta' = \text{constante}$ en topographie sans tendance, que nous ferons correspondre au niveau η'_{bf} défini au chapitre précédent. La bathymétrie du lit mineur sera donc définie par le domaine $\eta' - \eta'_{\text{bf}} < 0$, tandis que la topographie du lit majeur correspondra au domaine $\eta' - \eta'_{\text{bf}} > 0$.

Les modèles usuels pour la forme de la section en travers sont de type paraboliques (par exemple, *Deutsch et Wang*, 1996) ou une forme très similaire en cosinus (par exemple, *Abramian*, 2015). Conformément à la

définition des deux demi-largeurs $W_{\text{bf}}^{(l)}$ et $W_{\text{bf}}^{(r)}$, nous définissons la bathymétrie sans tendance $z'_b(s, w)$ dans la section pour les deux parties à gauche ($w > 0$) et à droite ($w < 0$) de la ligne de talweg à l'aide d'un cosinus :

$$z'_b(s, w) = \begin{cases} -h_{\text{talweg}}(s) \cos\left(\frac{\pi w}{2 W_{\text{bf}}^{(l)}(s)}\right) & \text{pour } 0 \leq w \leq W_{\text{bf}}^{(l)}(s) \\ -h_{\text{talweg}}(s) \cos\left(\frac{\pi w}{2 W_{\text{bf}}^{(r)}(s)}\right) & \text{pour } -W_{\text{bf}}^{(r)}(s) \leq w < 0 \end{cases}$$

De façon à minimiser les degrés de liberté, nous utilisons les corrélations mises en évidence au Chapitre 2 pour imposer un déphasage nul entre le maximum de profondeur et l'apex des méandres. En d'autres termes, nous positionnons le maximum de profondeur à l'endroit où la direction du chenal est parallèle à la direction régionale et où la pente est donc maximale. Une variante de positionnement du maximum de profondeur serait de le localiser au niveau du maximum de courbure ; ces deux choix donnent des résultats très proches l'un de l'autre. Dans le modèle de Kinoshita, l'angle $\theta(s)$ est nul au voisinage des positions telles que $\frac{2\pi s}{\lambda} = 0, \pi, 2\pi, 3\pi, \text{etc.}$, c'est-à-dire chaque fois que la phase $k_0 s = \frac{2\pi s}{\lambda}$ augmente de π ou que l'abscisse curviligne augmente d'une demi-longueur d'onde $\frac{\lambda}{2}$. Nous prenons donc comme fonction :

$$h_{\text{talweg}}(s) = (h_{\text{pool}} - h_{\text{run}}) \left[\frac{1 + \cos(k_0 s)}{2} \right] + h_{\text{run}}$$

Cette fonction a donc deux degrés de liberté, h_{pool} la profondeur de talweg dans les mouilles et h_{run} la profondeur de talweg dans les radiers, le phasage avec la forme en plan étant imposé. La bathymétrie finale, résultant à la fois de la paramétrisation des demi-largeurs, de la forme des sections et de la variabilité longitudinale de la profondeur de talweg, a l'aspect illustré sur la Figure 5.9.

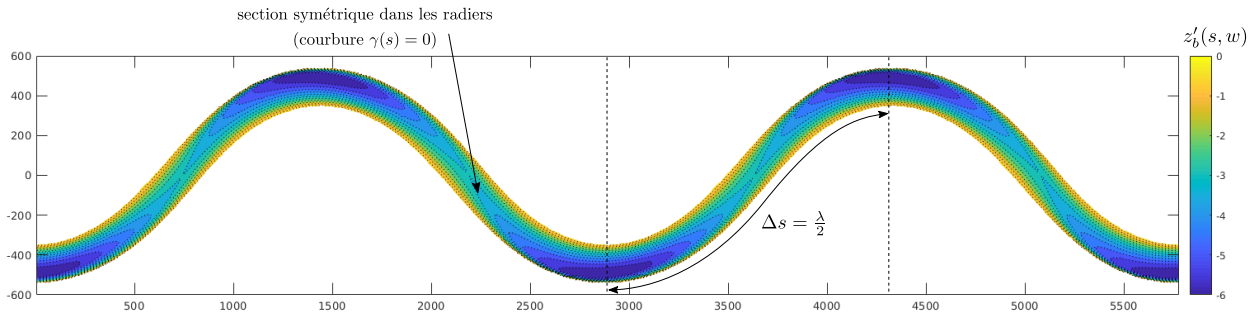


Figure 5.9: Bathymétrie-type résultant de la paramétrisation des demi-largeurs gauche et droite, de la forme en cosinus des sections, et de la variabilité longitudinale de la profondeur de talweg. Les valeurs utilisées ici sont $h_{\text{run}} = 3$ m pour la profondeur des radiers et $h_{\text{pool}} = 6$ m pour la profondeur des mouilles à l'apex des méandres.

5.2.2.4 Topographie en lit majeur

Une fois définie l'emprise plane du chenal à plein bord, nous pouvons terminer la paramétrisation du modèle topographique par le lit majeur (*floodplain*). Là encore nous allons nous baser sur la coordonnée transverse conforme Φ pour proposer le modèle le plus parcimonieux possible.

La Figure 5.10 illustre la structure de cette coordonnée dans le lit majeur uniquement, c'est-à-dire en masquant les valeurs dans l'emprise du chenal. Nous voyons en particulier qu'il existe une valeur minimale

de $|\Phi|$, que nous notons Φ_{\min} . Les points vérifiant $|\Phi| = \pm\Phi_{\min}$ normalement sont localisés côté intérieur des méandres, nous avons déjà mentionné plus haut que les iso-valeurs de la coordonnée Φ “mimaient” en quelque sorte la forme des barres de méandres. Le principe de modélisation de la topographie équivalente en lit majeur est de construire une relation univoque croissante entre $|\Phi|$ et η' une cote dans la topographie réelle dépouillée de sa tendance, et ce pour toutes les valeurs de $|\Phi| \geq \Phi_{\min}$ en lit majeur.

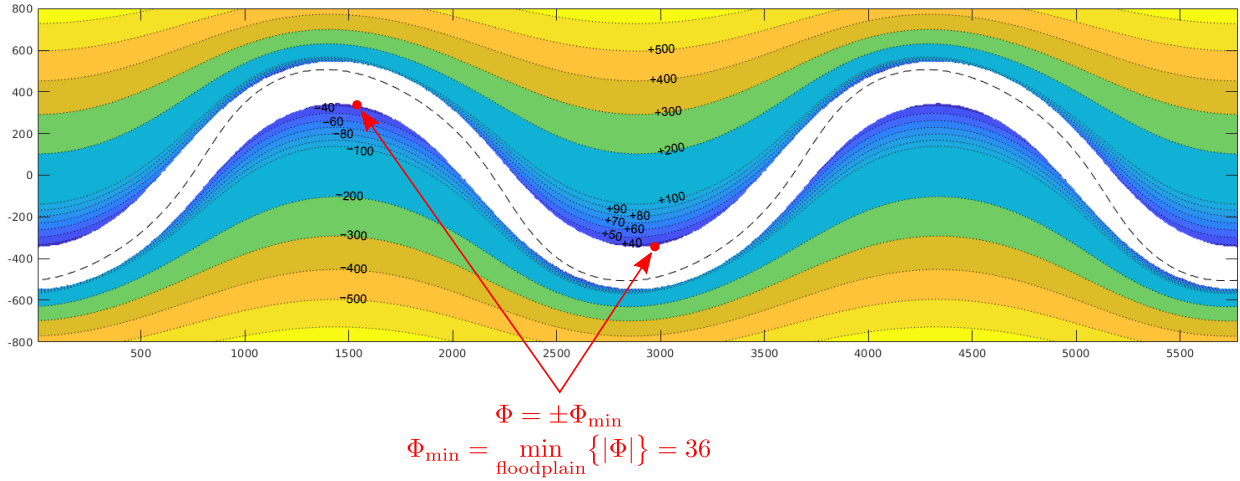


Figure 5.10: Localisation du minimum de valeur absolue de la coordonnée transverse conforme Φ en lit majeur.

La façon la plus simple de procéder est de passer par la variable d'aire inondée en lit majeur par unité de longueur de chenal, $\frac{1}{\sigma L} A_{fp}(\eta')$, définie au chapitre précédent. En traitant $|\Phi|$ comme une pseudo-élévation, nous pouvons également construire une courbe $A_{fp}(|\Phi|)$ dans le modèle géométrique équivalent. Pour chaque valeur de $|\Phi|$, nous pouvons alors associer une valeur de η' , par la correspondance :

$$|\Phi| \longrightarrow \underbrace{\frac{1}{\sigma L} A_{fp}(|\Phi|)}_{\text{tronçon modèle}} = \underbrace{\frac{1}{\sigma L} A_{fp}(\eta')}_{\text{tronçon réel}} \longrightarrow \eta'$$

La Figure 5.11 illustre cette correspondance. Si l'on souhaite associer par exemple une valeur de cote η' aux courbes iso-valeurs $\Phi = \pm 300$ m, il suffit de regarder l'aire en lit majeur comprise entre ces deux courbes dans le tronçon périodique (en enlevant l'aire à plein-bord, par définition) : on obtient la valeur de 593 m^2 par mètre de chenal. Nous pouvons finalement regarder, sur la topographie réelle dépouillée de sa tendance, pour quel niveau η' on obtient la même valeur d'aire inondée en lit majeur : on obtient la valeur $\eta' = 12.8$ m. Bien évidemment, pour assurer la continuité avec la bathymétrie en lit mineur, la valeur $\Phi = \pm\Phi_{\min}$ doit être associée à la valeur $\eta' = \eta'_{bf}$ niveau de plein-bord au tronçon.

La relation empirique obtenue est tracée sur la Figure 5.12 (courbe de gauche). Sur le graphique de droite, on a tracé la dérivée $\frac{d\eta'}{d\Phi}$ (évaluée numériquement) de cette fonction. La valeur de $\frac{d\eta'}{d\Phi}$ s'interprète comme une pente du lit majeur perpendiculairement au chenal. On constate que pour les grandes valeurs de la coordonnée Φ (c'est-à-dire quand celle-ci se comporte quasiment comme la coordonnée cartésienne y), la pente est d'environ 5% (0.005).

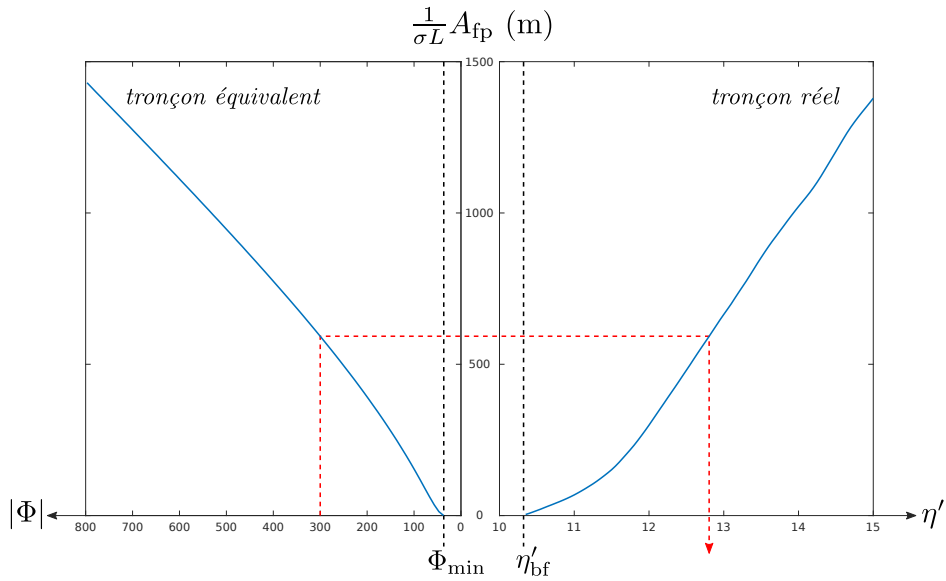


Figure 5.11: Table de correspondance permettant d'associer une valeur de cote sans tendance η' à chaque valeur $|\Phi|$ de la coordonnée transverse conforme. Pour une valeur de $|\Phi|$, on commence par regarder l'aire inondée en lit majeur par unité de largeur de chenal correspondant à ce niveau (à gauche), puis, en utilisant la courbe bijective observée entre $\frac{A_{fp}}{\sigma L}$ et η' , on identifie la valeur de η' (à droite).

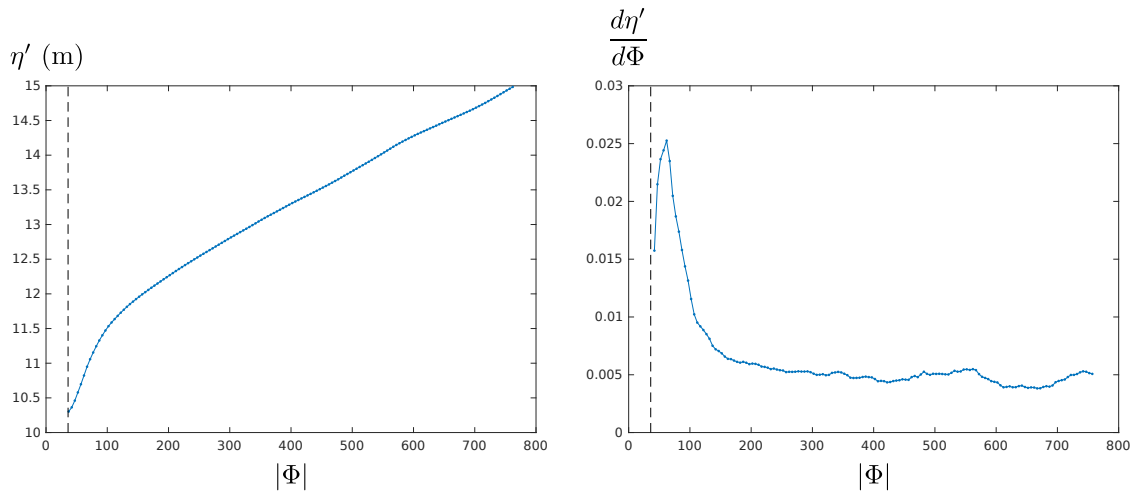


Figure 5.12: Courbe $\eta'(|\Phi|)$ donnée par la méthode de correspondance (à gauche), et sa dérivée évaluée numériquement (à droite).

Cette approche par table de correspondance a l'avantage de la simplicité, mais l'inconvénient de ne se focaliser que sur l'adéquation entre les signatures $A_{fp}(\eta')$ simulées et observées. Cette signature est évidemment importante, mais elle n'est pas la seule : sur le plan hydraulique, la signature $R_{h,fp}^*(\eta')$ (rayon hydraulique équivalent en lit majeur) est tout aussi importante.

Si on veut réaliser un ajustement de ce type, il faut donc se donner une forme explicite pour la fonction

$\eta'(\Phi)$ ou sa dérivée. La dérivée numérique $\frac{d\eta'}{d\Phi}$ précédente permet d'intuiter une paramétrisation flexible :

$$\begin{cases} \frac{d\eta'}{d\Phi} = a_0 \exp\left(-\left(\frac{\Phi - \Phi_{\min}}{\Phi_c}\right)^2\right) + a_\infty \\ \eta'(|\Phi|) = \frac{a_0 \Phi_c \sqrt{\pi}}{2} \operatorname{erf}\left(\frac{|\Phi| - \Phi_{\min}}{\Phi_c}\right) + a_\infty |\Phi| + \eta'_{\text{bf}} \end{cases}$$

où les paramètres a_0 , a_∞ et Φ_c sont à ajuster pour reproduire au mieux les courbes $V_{\text{fp}}(\eta')$, $A_{\text{fp}}(\eta')$ et/ou $R_{h,\text{fp}}^*(\eta')$ déduites du MNT en lit majeur, conjointement plus pertinentes que la seule hypsométrie $A_{\text{fp}}(\eta')$ pour approcher le comportement hydraulique du lit majeur. Les paramètres a_0 et a_∞ s'interprètent comme des pentes transversalement à l'écoulement : à proximité du chenal ($\Phi \rightarrow 0$, $\eta' \rightarrow \eta'_{\text{bf}}$), une variation $d\Phi$ de la coordonnée transverse conforme donnera une variation de cote $d\eta' = a_0 d\Phi$, tandis que très loin du chenal ($\Phi \rightarrow \infty$) une variation $d\Phi$ donnera une variation $d\eta' = a_\infty d\Phi$. La transition entre ces deux comportements aux limites s'effectue sur une distance caractéristique Φ_c (que l'on peut interpréter en mètres puisqu'il s'agit d'une coordonnée). D'autres modèles paramétriques sont envisageables, il ne s'agit que d'un exemple assez parcimonieux. Les courbes $\frac{A_{\text{fp}}(\eta')}{\sigma L}$ (aire inondée en lit majeur par unité de longueur de chenal) et $R_{h,\text{fp}}^*(\eta')$ observées et modélisées pour le tronçon de la Garonne sont illustrées sur la Figure 5.13. Les paramètres optimisés sont les suivants :

$$\begin{cases} a_0 = 0.018 \text{ m/m} \\ a_\infty = 0.0052 \text{ m/m} \\ \Phi_c = 73 \text{ m} \end{cases}$$

Le lit majeur étant par définition observable la plupart du temps, nous faisons l'hypothèse que ces paramètres (ou ceux d'une paramétrisation similaire) peuvent être identifiés sur n'importe quel tronçon hydraulique sur lequel on veut faire des estimations de débit. Ils ne feront donc pas partie des paramètres à identifier dans la méthode inverse.

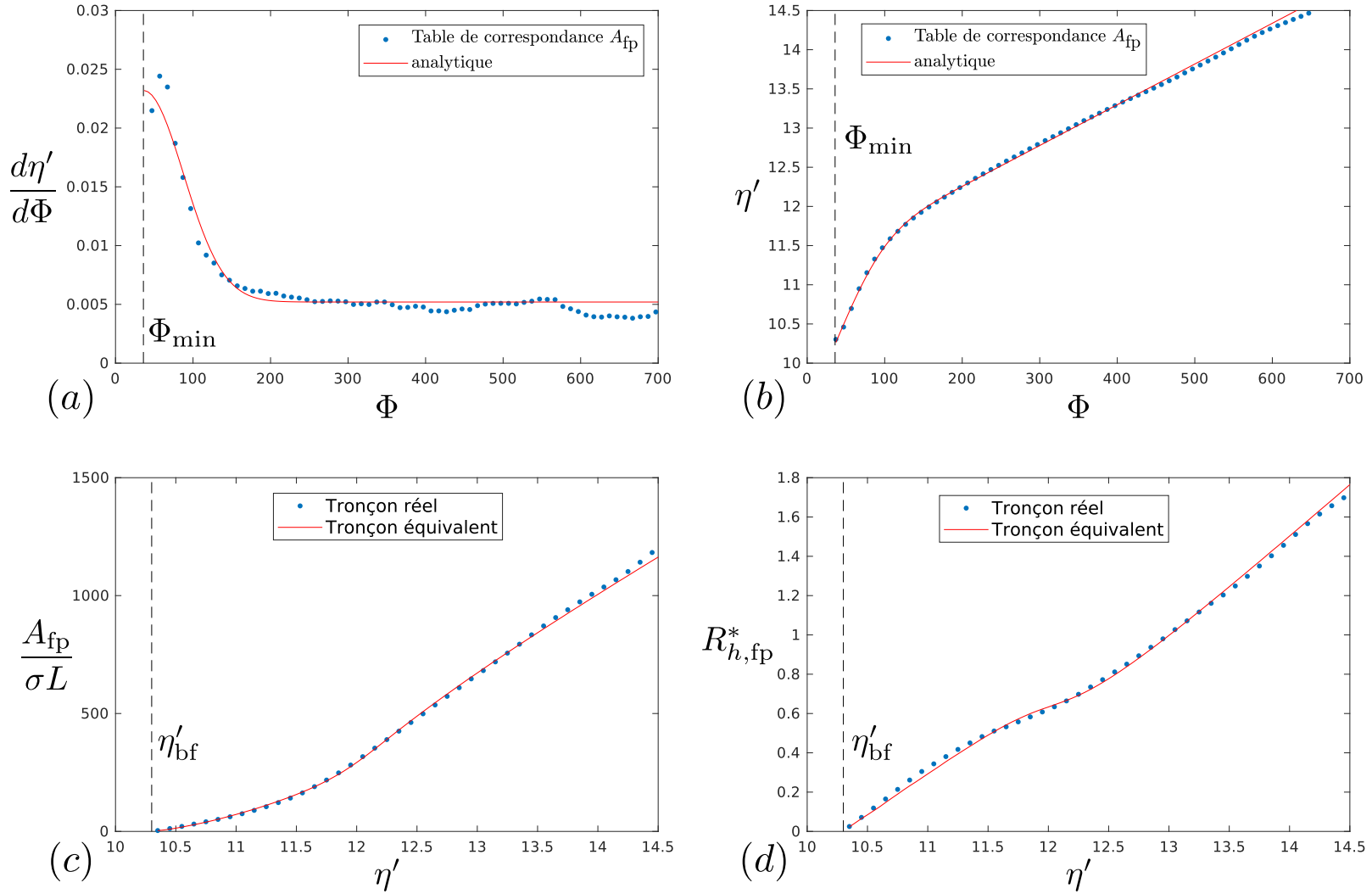


Figure 5.13: Modélisation des signatures topographiques en lit majeur, par ajustement des paramètres (a_0, a_∞, Φ_c) décrivant la relation $\eta'(|\Phi|)$, où Φ est la coordonnée transverse conforme. (a) Ajustement de $\frac{d\eta'}{d\Phi}$. (b) Courbe $\eta'(|\Phi|)$ résultante, comparée à celle obtenue par la méthode plus simple de table de correspondance. (c) et (d) Courbes observées et modélisées d'aire inondée en lit majeur par unité de longueur de chenal ($A_{fp}(\eta')/\sigma L$) et de rayon hydraulique équivalent en lit majeur $R_{h,fp}^*(\eta')$.

5.2.3 Résumé des paramètres du modèle géométrique périodique

Paramètre		Signification	Inversion	Valeur / Plage
Forme en plan	λ	Longueur d'onde la forme en plan (m)	non	3700
	θ_0	Déviations angulaire max. du chenal (rad)	non	0.96
	α_s	Asymétrie	non	$\frac{1}{10}$
	α_f	Aplatissement	non	$\frac{1}{96}$
	W_{bf}	Largeur de plein-bord (m)	non	200
Topographie lit majeur	I_{reg}	Pente régionale du lit majeur (m/m)	non	0.0011
	a_0	Pente transverse proche du chenal (m/m)	non	0.018
	a_∞	Pente transverse loin du chenal (m/m)	non	0.0052
	Φ_c	Distance caractéristique de transition (m)	non	73
Bathymétrie lit mineur	h_{run}	Profondeur max. des radiers (m)	oui	[2.6,8]
	Δh_{pool}	Approf. max. des mouilles relativement aux radiers (m)	oui	[0, 8]
Frottement	K_{min}	Strickler lit mineur ($m^{1/3} \cdot s^{-1}$)	oui	[10,40]
	K_{maj}	Strickler lit majeur ($m^{1/3} \cdot s^{-1}$)	oui	[5,30]

Table 5.1: Résumé des paramètres du modèle géométrique périodique.

5.2.4 Implémentation sous Basilisk

5.2.4.1 Procédure de génération de la topographie équivalente périodique

Comme nous l'expliquerons plus en détail dans la partie suivante, l'objectif de la procédure d'inversion est de tester un grand nombre de variantes de la topographie équivalente, en faisant varier les méta-paramètres permettant de la générer. Plutôt que de générer chaque variante à l'extérieur de Basilisk, on fournit au contraître en entrée du code toujours le même semis de points, dont les attributs sont :

- les coordonnées cartésiennes (x, y)
- la coordonnée conforme transverse Φ
- le cas échéant, si le point est dans le voisinage tubulaire du chenal, les coordonnées (s, w)

Sur la base de ce semis de points, et avec la donnée des paramètres h_{run} , h_{pool} , K_{min} et K_{maj} , le code génère "à la volée" (dans l'évènement `init`) le terrain équivalent périodique ainsi que la carte de coefficient

de rugosité. Cette procédure fait appel à une structure KDT (k -dimensional tree) classiquement utilisée par Basilisk pour gérer la topographie sur un maillage adaptatif, et héritée de son prédécesseur Gerris (<http://basilisk.fr/src/terrain.h>).

Comme pour le modèle Garonne, le modèle équivalent fonctionne avec la topographie sans tendance, la pente régionale étant prise en compte sous forme d'un terme source de quantité de mouvement, à la manière d'un vent mettant en mouvement la surface d'un lac horizontal. Cette méthode de résolution est illustrée en Figure 5.14 dans le cas particulier de conditions aux limites périodiques : ici, seuls les régimes permanents sont bien sûr pertinents à analyser. L'eau qui sort à la limite aval du domaine "rentre" en quelque sorte à nouveau côté amont, l'état stationnaire étant obtenu quand le frottement compense le terme moteur fictif dû à la pente régionale. Le modèle est initialisé avec un certain volume, obtenu en initialisant η' à une valeur uniforme, et avec une vitesse nulle partout. En fin de simulation, on récupère les signatures désirées, telles que la moyenne spatiale $\overline{\eta'}$ de la cote de la surface libre $\eta(x, y)$ stabilisée, sa variance, le débit circulant dans le tronçon, etc.

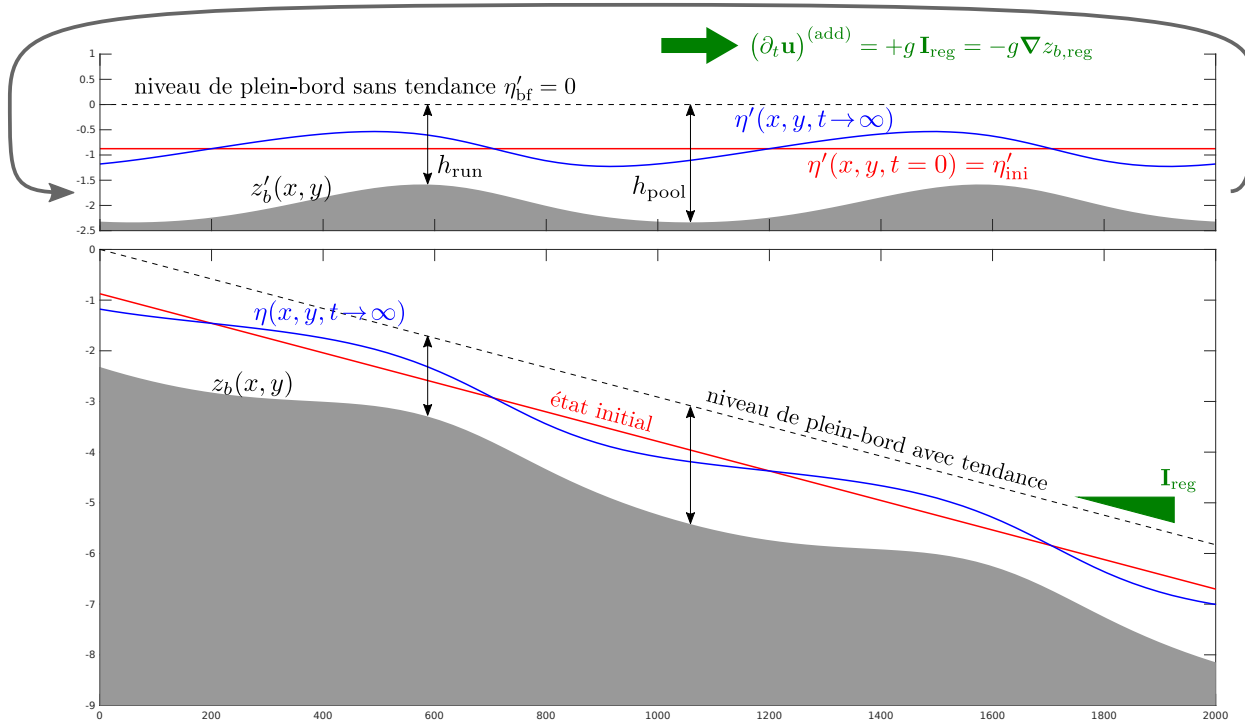


Figure 5.14: Méthode de calcul en géométrie périodique avec Basilisk (illustration 1D). Le calcul s'effectue dans le système de cote / altitude sans tendance (schéma du haut) : le domaine est initialisé avec un volume d'eau associé à une cote sans tendance η'_{ini} uniforme (en rouge). Cela correspond, dans la topographie avec tendance, à une surface libre plane inclinée dans la direction de la pente régionale \mathbf{I}_{reg} qui est aussi parallèle au niveau de plein-bord *bankfull* (par construction, ce niveau est supposé parallèle lui aussi à la tendance de la topographie régionale et correspond au niveau $\eta'_{bf} = 0$ dans la construction du modèle périodique). Algorithmiquement, la pente régionale est prise en compte dans le calcul Basilisk en topographie sans tendance par le biais d'un terme d'accélération fictif additionnel. La simulation converge vers la solution stationnaire $\eta'(x, y, t \rightarrow \infty)$ tracée en bleu, avec son équivalent $\eta(x, y, t \rightarrow \infty)$ dans la topographie avec tendance. Nous représentons ici 2 répétitions du motif périodique de base pour illustrer l'effet de la suppression de la tendance régionale, mais l'intérêt est bien évidemment de limiter le domaine de calcul à un seul motif.

5.2.4.2 Approximation de l'onde diffusante

Le calcul en géométrie périodique peut bien évidemment être effectué avec le solveur Saint-Venant complet. Cependant, du fait des contraintes propres à l'inversion (test d'un grand nombre de variantes), il peut être utile d'accélérer les calculs en utilisant l'approximation de l'onde diffusante. Cette solution est d'autant plus attractive que seul l'état stationnaire final nous intéresse : le transitoire n'ayant pas de sens physique sur la géométrie périodique, on peut tirer parti du caractère implicite du solveur des problèmes d'advection-diffusion dans Basilisk (le solveur Saint-Venant est, lui, explicite). La dérivation de cette approximation et son implémentation sont décrites en Annexe B, et l'impact de l'utilisation de cette approximation à la place du système de Saint-Venant complet est étudiée dans la section suivante avec le premier essai de modélisation inverse sur une géométrie synthétique. Elle sera conservée dans le second essai de modélisation inverse des régimes stationnaires simulés sur la Garonne. Le gain en temps de calcul est de l'ordre d'un facteur 50 à 100.

5.3 Modélisation inverse

Dans la première partie de ce chapitre, nous avons présenté la structure générale du modèle équivalent périodique qui permet, via un petit nombre de paramètres, de créer une géométrie synthétique simplifiée reproduisant certaines des signatures topographiques du tronçon réel. Le but de cette dernière partie est maintenant d'évaluer la capacité du modèle à reproduire les signatures hydrauliques du tronçon réel, et en particulier de tenter de répondre à la question suivante : si le modèle est calé de façon à reproduire les courbes signatures de l'aire inondée $\bar{A}(\bar{\eta}')$ en fonction de la moyenne de la cote de la surface libre de l'écoulement $\bar{\eta}'$, et de la variance $\sigma_{\bar{\eta}'}^2(\bar{\eta}')$ de cette cote, reproduit-il la signature non-observable à distance de courbe de tarage au tronçon $Q(\bar{\eta}')$?

L'obtention des résultats de la simulation de référence ayant pris du retard au cours des dernières semaines, cette question n'a pas pu être abordée en s'appuyant sur un jeu cohérent de variables hydrauliques en transitoire. Néanmoins, nous présentons dans la partie suivante 2 tests d'inversion :

1. un test d'inversion sur une géométrie synthétique, où la géométrie à inverser est elle-même générée selon le modèle périodique avec des paramètres connus, et où les signatures hydrauliques sont générées en simulant des régimes permanents avec le modèle de Saint-Venant complet. Ce test a pour but d'évaluer le contenu informatif des signatures hydrauliques concernant la valeur des paramètres sous-jacents, et également d'évaluer l'impact de l'utilisation de l'approximation de l'onde diffusante dans la modélisation inverse.
2. un test d'inversion sur le tronçon de la Garonne, les signatures hydrauliques étant extraites des simulations en régime permanent effectuées pour caler le coefficient de Strickler en lit mineur. La gamme de débits disponibles ne couvre donc que les débits non-débordants, mais cela constitue un premier test d'inversion en topographie réelle.

Nous commençons cette partie par une courte introduction sur la méthode d'inversion, l'algorithme génétique CaRaMEL.

5.3.1 Procédure d'inversion stochastique

5.3.1.1 Principe

Etant donnée une structure de modèle direct, c'est-à-dire l'ensemble des équations utilisées pour produire des variables de sortie à partir de données d'entrée, différentes approches existent pour résoudre le problème inverse, c'est-à-dire pour remonter aux valeurs des paramètres connaissant certaines variables de sortie. Selon la structure du modèle direct (régression statistique, modèle conceptuel, ensemble d'équations différentielles modélisant un système dynamique, etc.), deux approches sont globalement possibles :

- des approches basées sur le calcul de la dérivée d’une fonction de coût (erreur) par rapport aux variables contrôlant la réponse (par exemple, *Plessix*, 2006) : ici, ce sont les paramètres de géométrie, mais cela peut être également les états du modèle en assimilation de données. L’idée est alors de modifier de façon itérative ces variables de contrôle de la réponse en inversant l’opérateur dérivé local (matrice jacobienne), selon une procédure déterministe.
- dans certains cas, la difficulté à évaluer la dérivée de la fonction de coût fait préférer une méthode stochastique : on renonce alors à faire “remonter” l’information d’erreur sur les sorties par le biais d’une procédure déterministe, et on se limite à une procédure d’essais-erreurs, qui a un caractère exploratoire et moins localisé dans l’espace des variables de contrôle.

Les deux approches ont des avantages et des inconvénients : l’approche par dérivée de l’erreur est en général bien adaptée pour des problèmes dans lesquels le modèle direct est très gourmand en temps de calcul (comme par exemple la résolution des équations de la circulation atmosphérique), et où l’on cherche plutôt à modifier à la marge les valeurs de variables de contrôle assez bien contraintes. À l’inverse, si l’on se trouve dans un cas où un *run* du modèle direct est assez rapide, et/ou si la gamme des valeurs possibles des variables de contrôle est large et peu contrainte, l’approche stochastique peut être préférable.

Le modèle périodique stationnaire proposé dans ce chapitre a été conçu pour être utilisé dans ce second type d’approche : le petit nombre de paramètres contrôlant une géométrie simplifiée, la recherche uniquement d’un état stationnaire, et l’utilisation de l’approximation de l’onde diffusante font qu’il est possible de faire “foisonner” des paramétrisations de la structure dans une approche par essais-erreur. Par ailleurs, nous avons mentionné que l’approche suivie était d’utiliser 2 des signatures calculables par le modèle, la courbe signature de l’aire inondée $\tilde{A}(\eta')$ et de la variance $\sigma_{\eta'}^2(\eta')$. Pour chaque combinaison testée des 4 paramètres contrôlant la réponse du tronçon équivalent, nous pouvons donc calculer 2 scores ou erreurs :

- un score d’écart entre la courbe signature $\tilde{A}(\eta')$ du tronçon réel, et celle du tronçon équivalent,
- de même, un score d’écart entre la courbe signature $\sigma_{\eta'}^2(\eta')$ du tronçon réel, et celle du tronçon équivalent.

Nous pourrions combiner ces deux scores en une moyenne pondérée, mais cela reviendrait à perdre de l’information : en effet, une paramétrisation peut donner un bon score pour l’une des deux signatures, et un score médiocre sur l’autre, et l’on masque les possibilités d’amélioration sur les deux axes indépendamment en faisant la moyenne. En outre, les deux scores ne sont pas forcément commensurables, et la question des valeurs des poids n’est pas nécessairement triviale : un outil adapté à ce genre de situation est alors l’optimisation multi-objectif. Plutôt que de chercher à maximiser une fonction de coût scalaire (qui peut être la norme 2 d’un vecteur de grande dimension, mais cela n’en reste pas moins un scalaire), nous pouvons définir l’optimalité au sens de Pareto : on dira qu’une solution est Pareto-optimale si on ne peut pas améliorer l’un des scores sans dégrader strictement au moins l’un des autres. On voit que cette définition n’est pas limitée à deux scores, elle se généralise à n’importe quelle dimension (cependant, quand le nombre de dimensions / d’axes augmente, toute solution tend à devenir Pareto-optimale puisqu’il devient facile qu’elle se “spécialise” sur un seul des scores, en étant très mauvaise sur les autres).

La Figure 5.15 illustre le concept de Pareto-optimalité sur un cas simple : on suppose un modèle à 3 paramètres θ_1 , θ_2 et θ_3 , évalué sur deux scores f_1 et f_2 que l’on cherche à maximiser. Idéalement, on souhaiterait trouver un jeu de paramètres unique qui maximise simultanément f_1 et f_2 , et donc localisé dans le coin en haut à droite dans le graphe (f_1, f_2) (si f_1 et/ou f_2 représentent une erreur à minimiser, cela ne fait qu’inverser les axes, on visera alors le coin bas-gauche). En pratique, c’est rarement le cas, et on observe souvent le type de compromis illustré sur la figure. Les solutions marquées par un point noir sont dites *non-dominées* : on ne peut pas passer d’un point noir à un autre i.e., améliorer l’une des fonctions score, sans dégrader l’autre. À l’inverse, certaines solutions sont *dominées* (points blancs) : pour chacune de ces solutions, il est possible de trouver une autre solution (un point noir) qui améliore les deux scores à

la fois. L'ensemble des solutions non-dominées (ou Pareto-optimales, de façon synonyme) forment le front de Pareto approximé. Si les paramètres et les scores prennent des valeurs continues, l'ensemble des solutions non-dominées est infini (ensemble en rouge) ; ce qu'on attend d'un algorithme d'optimisation multi-objectif est alors de proposer un échantillonnage correct de la variabilité de cet ensemble.

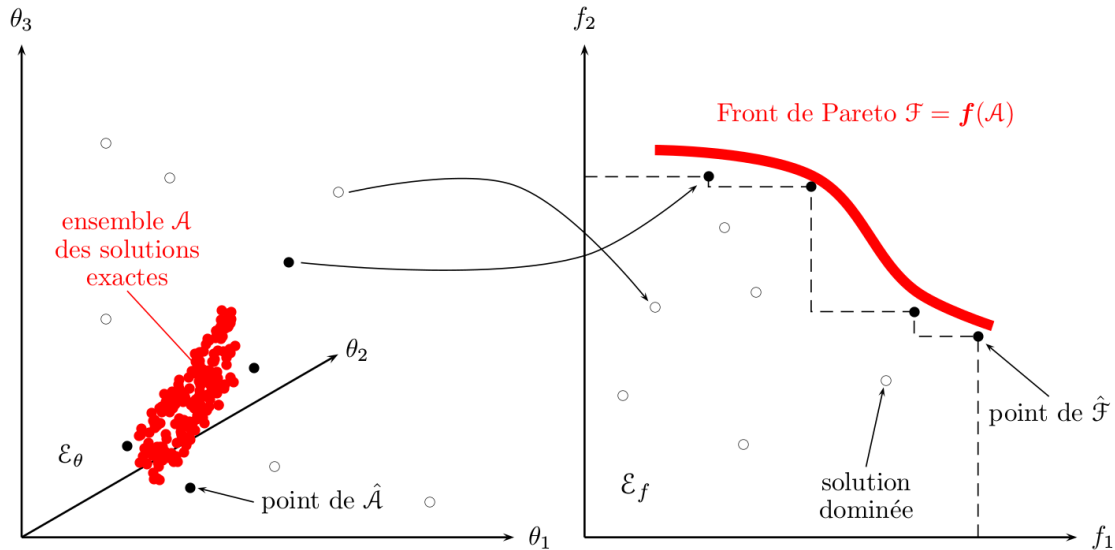


Figure 5.15: Illustration des concepts de dominance et de Pareto-optimalité sur un cas simple d'un espace des paramètres \mathcal{E}_θ à 3 dimensions, et un espace des scores/objectifs \mathcal{E}_f à 2 dimensions.

5.3.1.2 Mise en oeuvre avec l'optimiseur CaRaMEL

Pour l'inversion des paramètres du modèle périodique, nous utilisons l'algorithme CaRaMEL (*Monteil et al., 2020*), dans l'environnement Scilab, avec des appels au code Basilisk par l'intermédiaire de la fonction `host()` qui crée une session de l'interpréteur de commande de l'environnement Linux. Cet algorithme appartient à la famille des algorithmes dit génétiques, qui proposent de nouveaux paramètres candidats par recombinaison de vecteurs de paramètres déjà évalués. Même si l'algorithme CaRaMEL inclue justement des règles de génération de nouveaux candidats mimant des méthodes de type gradient, il reste fondamentalement un algorithme stochastique fonctionnant par essais-erreurs. Le lecteur est renvoyé à l'article décrivant l'algorithme pour plus de détails sur les différentes règles de génération de nouveaux candidats.

L'avantage de l'approche multi-objectif est double : elle permet de prendre en compte de l'information provenant de réponses multiples et pas nécessairement commensurables et, comme elle ne produit pas une solution unique mais un ensemble de solution, elle permet de quantifier l'incertitude sur la paramétrisation, quoique de façon non-formelle (*Beven, 2006*). Ce point sera illustré plus loin.

Si on souhaite cependant résumer l'optimalité d'une solution par une valeur numérique unique scalaire, une méthode d'indexation est présentée sur la Figure 5.16. A partir de l'ensemble des solutions testées ("population" complète), la stratégie est la suivante :

- on identifie dans l'ensemble de la population les solutions non-dominées, qui forment le front de Pareto approximé d'ordre 1,
- on retire ces solutions de l'ensemble initial, et on identifie les solutions non-dominées dans les solutions restantes ; elles forment le front de Pareto d'ordre 2,

- on répète itérativement cette procédure jusqu'à avoir indexé l'ensemble des solutions.

On obtient ainsi une indexation des solutions en “couches” ou “pelures d'oignon”, les solutions les plus superficielles étant les plus proches de l'optimalité. A nouveau, la procédure est ici illustrée pour 2 scores, mais elle est généralisable à un nombre arbitraire de scores.

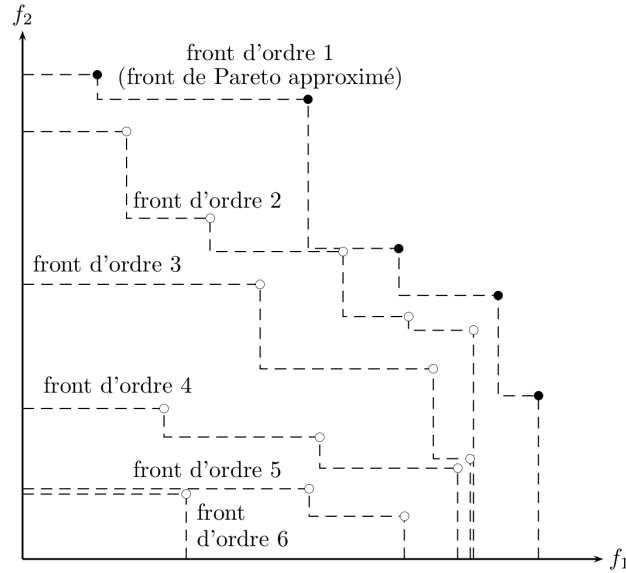


Figure 5.16: Indexation des solutions dans l'espace des scores (f_1, f_2) en couches successives. On commence par identifier les solutions non-dominées (front de Pareto à l'ordre 1), puis on itère : la couche 2 est l'ensemble des solutions non-dominées après avoir enlevé la couche d'indice 1, etc.

5.3.2 Test d'inversion sur un cas synthétique

Dans un premier temps, nous allons travailler sur une topographie contrôlée, en cherchant à inverser les signatures produites par une modèle de référence lui-même généré par le modèle périodique. Cette simulation de référence a les caractéristiques suivantes :

- les paramètres sont $h_{run} = 3$ m, $h_{pool} = 6$ m, $K_{min} = 25 \text{ m}^{1/3} \cdot \text{s}^{-1}$, $K_{maj} = 17 \text{ m}^{1/3} \cdot \text{s}^{-1}$,
- les régimes permanents à différents volumes (fonction du niveau η'_{ini} uniforme imposé comme condition initiale) sont produits par le solveur de Saint-Venant complet.

Le premier objectif de ce test est d'une part de voir si les signatures (en excluant donc la signature courbe de tarage au tronçon) sont suffisamment informatives pour estimer la valeur des paramètres, et donc estimer la courbe de tarage au tronçon. Le second objectif est également d'évaluer l'impact de l'utilisation de l'approximation de l'onde diffusante pour inverser les signatures produites avec le solveur de Saint-Venant complet.

5.3.2.1 Simulation de référence en géométrie périodique

La Figure 5.17 présente les cartes de la surface libre sans tendance $\eta'(x, y)$ pour les 13 régimes stationnaires simulés sur la géométrie périodique de référence.

Le Tableau 5.2 donne les valeurs des signatures pour les différents états stationnaires obtenus, et la Figure 5.18 illustre graphiquement les courbes signatures $\frac{\tilde{A}(\bar{\eta}')}{\sigma_L}$, $\sigma_{\eta'}(\bar{\eta}')$, et la courbe de tarage au tronçon $Q(\bar{\eta}')$.

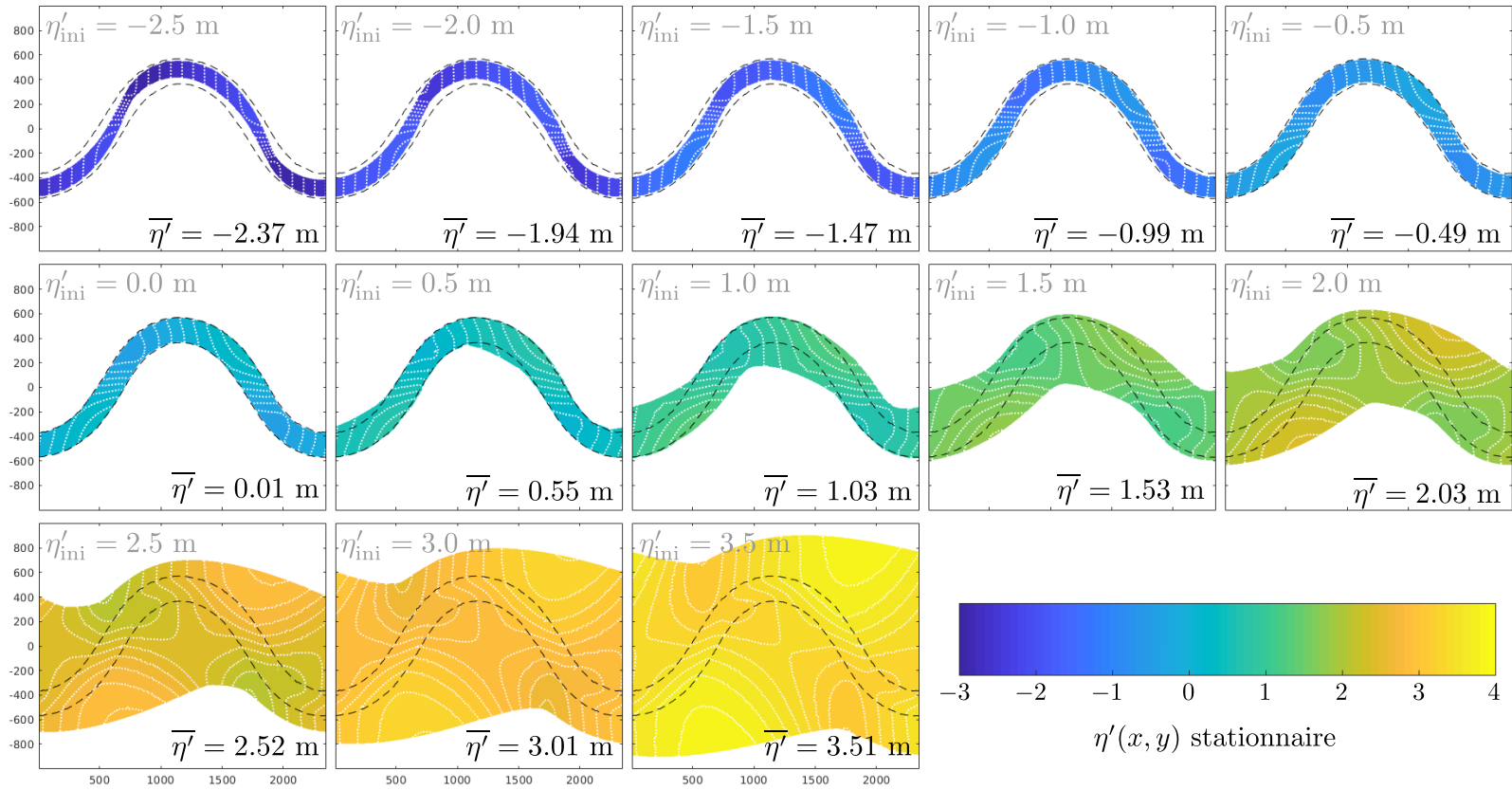


Figure 5.17: Cartes de la cote de la surface libre sans tendance $\eta'(x, y)$ obtenues après convergence vers un régime permanent, pour différents volumes prescrits dans le tronçon. Le volume est prescrit sous la forme de la condition initiale $\eta'(x, y, t = 0) = \eta'_{ini}$ uniforme. Les courbes en pointillés blancs sont les iso-valeurs de η' à intervalle de 0.1 m, et les courbes en pointillés noirs indiquent l'emprise du chenal (lit mineur). La valeur moyenne $\bar{\eta}'$ de la cote de la surface libre stabilisée est également indiquée dans chaque cas.

η'_{ini} (m)	$\bar{\eta}'$ à l'état stationnaire final	$\tilde{A}/(\sigma L)$ aire inondée par unité de longueur de chenal (m ² /m)	$\sigma_{\eta'}$ (m)	Q (m ³ /s)
-2.5	-2.37	138	0.452	42
-2.0	-1.94	145	0.354	106
-1.5	-1.47	154	0.297	206
-1.0	-0.99	165	0.263	341
-0.5	-0.49	177	0.241	513
0.0	0.01	192	0.227	720
0.5	0.55	261	0.222	961
1.0	1.03	355	0.190	1245
1.5	1.53	472	0.192	1591
2.0	2.03	634	0.201	2028
2.5	2.52	817	0.201	2586
3.0	3.01	999	0.203	3320
3.5	3.51	1183	0.198	4269

Table 5.2: Signatures hydrauliques calculées sur la simulation de référence en géométrie périodique.

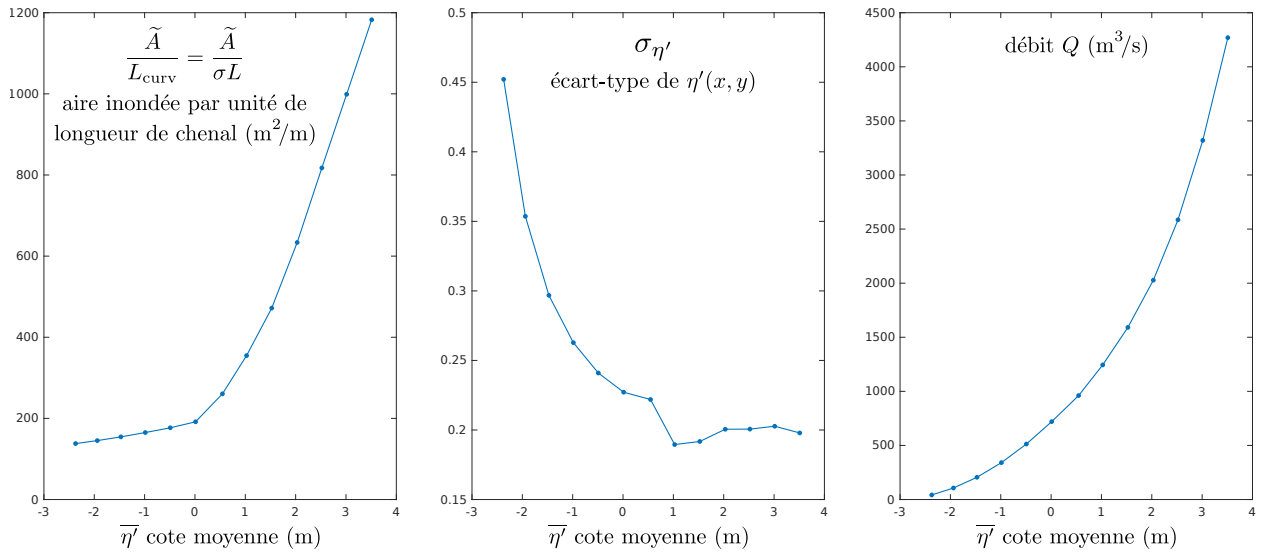


Figure 5.18: Tracé des courbes signatures pour la simulation de référence en géométrie périodique

5.3.2.2 Critères d'optimisation

Pendant l'optimisation, nous cherchons à maximiser l'adéquation entre les signatures "observables à distance" que sont la courbe d'aire inondée par unité de longueur de chenal en fonction de la moyenne $\bar{\eta}'$, $\tilde{A}(\bar{\eta}')$, et la courbe de variance de η' en fonction de $\bar{\eta}'$, $\sigma_{\eta'}(\bar{\eta}')$, en faisant comme si on ne connaissait pas la signature de courbe de tarage au tronçon $Q(\bar{\eta}')$: pour chaque solution testée, l'erreur sur cette signature est calculée comme critère de validation mais n'intervient pas dans l'optimisation.

La Figure 5.19 illustre la méthode de calcul des critères d'optimisation : il s'agit de l'erreur quadratique moyenne (RMSE) sur les variables exprimées en log, de façon à donner le même poids en relatif à toutes les gammes de valeurs de la signature. En ordre de grandeur, si le critère vaut 0.1, cela signifie que l'erreur sur la variable vaut de l'ordre de 10% ($e^x \simeq 1 + x$).

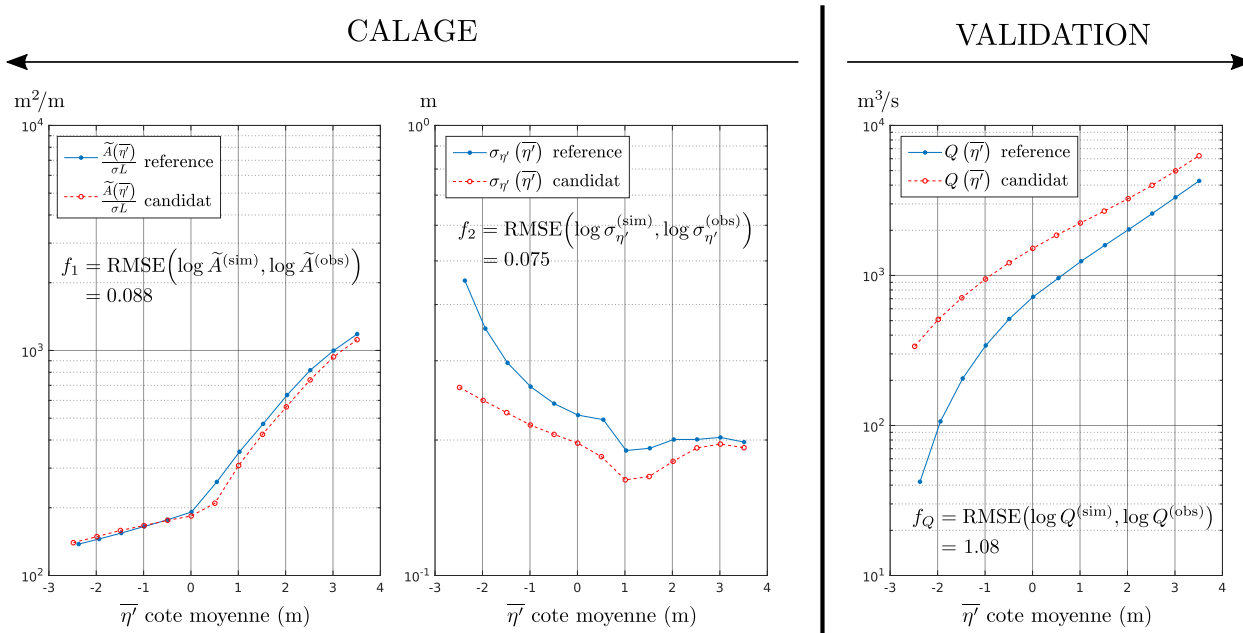


Figure 5.19: Critères utilisés pour évaluer chaque jeu de paramètres ($h_{run}, h_{pool}, K_{min}, K_{maj}$) : pour chaque courbe signature \tilde{A} , $\sigma_{\eta'}$ ou Q , on calcule une erreur quadratique moyenne sur la signature en log.

5.3.2.3 Résultats

La Figure 5.20 montre l'ensemble des solutions obtenues dans l'espace des critères. Sur le graphique de gauche, on a reporté l'ensemble des solutions dans la population en fin d'optimisation (316 jeux de paramètres), en distinguant en gris clair les solutions dominées et en gris foncé les solutions non-dominées (front de Pareto comptant 55 solutions ; contrairement à d'autres algorithmes, le critère d'arrêt dans CaRaMEL porte sur l'amélioration des fonctions objectifs, et non pas sur la condition que l'ensemble des jeux de paramètres de la population soient Pareto-optimaux). Le graphique de droite montre les mêmes points, mais affectés d'une taille et d'une couleur dépendant du critère de validation sur les débits. On constate que les points en bas à gauche, proches ou dans le front de Pareto, ont globalement des signatures sur le débit nettement meilleures que les points plus éloignés du front de Pareto.

La Figure 5.21 est une autre façon de visualiser ces résultats, en individualisant les signatures produites par chaque jeu de paramètres. Les signatures produites par les solutions non-dominées (Pareto-optimales) sont tracées en gris foncé, tandis que les signatures produites par les solutions dans le reste de la population

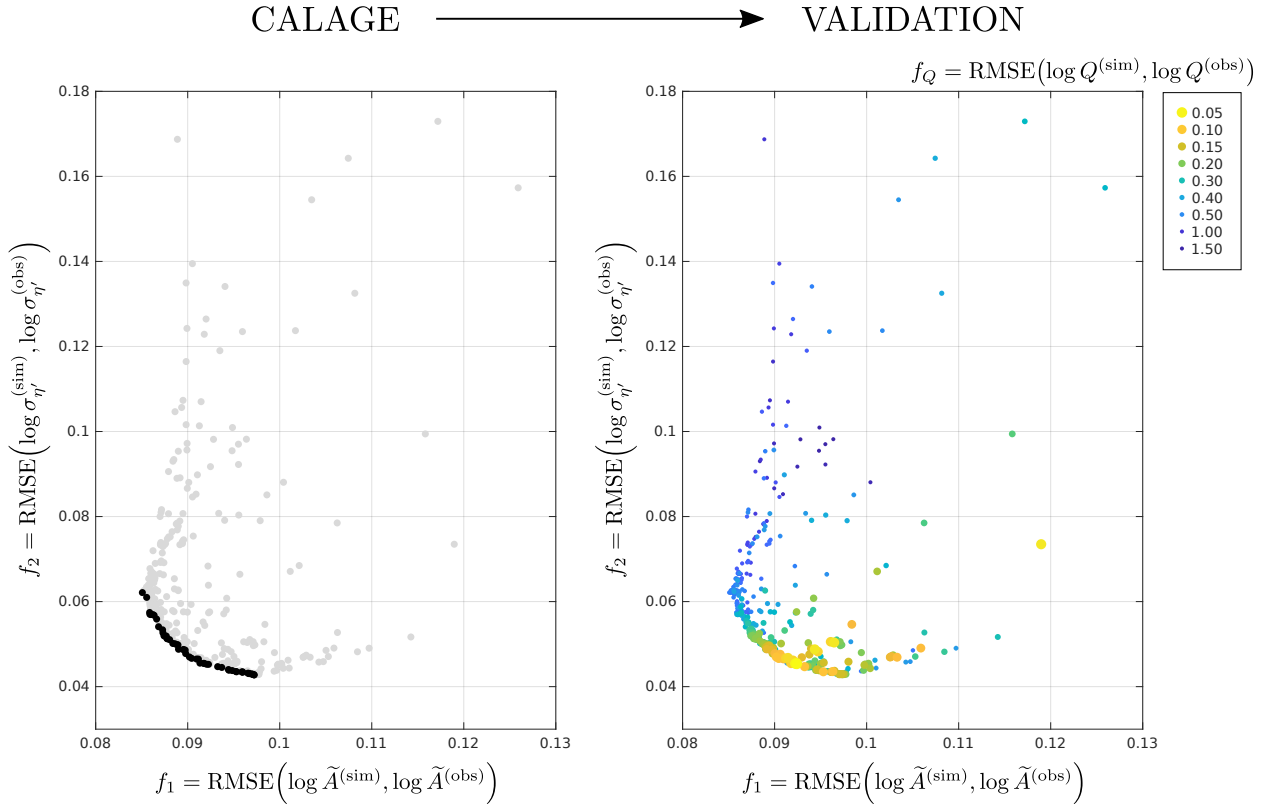


Figure 5.20: Résultats du test d'inversion en géométrie de référence périodique. A gauche, population et front de Pareto en fin d'optimisation : chaque point représente un jeu de paramètres testé, les points noirs indiquent les solutions non-dominées, optimales au sens de Pareto. A droite, on évalue chaque point en validation sur la courbe de tarage au tronçon. La taille et la couleur des points indique la RMSE sur le log du débit : plus le point est gros et vers les couleurs chaudes, moins la RMSE est importante et plus le vecteur de paramètres est performant dans la restitution de la courbe de tarage au tronçon.

sont tracées en gris claire. On constate clairement que l'ensemble produit par les solutions non-dominées est plus fin et plus précis que celui constitué des autres solutions : c'est bien sûr vrai pour les signatures $\tilde{A}(\bar{\eta}')$ et $\sigma_{\eta'}(\bar{\eta}')$, mais cela l'est également pour la signature courbe de tarage au tronçon $Q(\bar{\eta}')$ qui, rappelons-le, n'est pas prise en compte au calage. On constate par ailleurs une difficulté du modèle à reproduire la variance de la cote de la surface libre pour les très basses eaux. Cela peut s'expliquer par l'utilisation de l'approximation de l'onde diffusante : en effet, pour ces faibles débits, le nombre de Froude atteint des valeurs proches de 0.5–0.6, et l'absence de prise en compte de l'inertie peut devenir sensible (*Moussa et Bocquillon, 2000*). Cet effet est à approfondir.

La Figure 5.22 illustre quant à elle la distribution des paramètres optimisés. Idéalement, nous aimerions que la distribution de chaque paramètre soit le plus resserrée possible et centrée sur la valeur utilisée pour produire la simulation de référence :

$$\begin{cases} h_{\text{run}} = 3 \text{ m} & \text{profondeur maximum des radiers} \\ h_{\text{pool}} = 6 \text{ m} & \text{profondeur maximum des mouilles} \\ K_{\text{min}} = 25 \text{ m}^{\frac{1}{3}} \cdot \text{s}^{-1} & \text{Strickler lit mineur} \\ K_{\text{maj}} = 17 \text{ m}^{\frac{1}{3}} \cdot \text{s}^{-1} & \text{Strickler lit majeur} \end{cases}$$

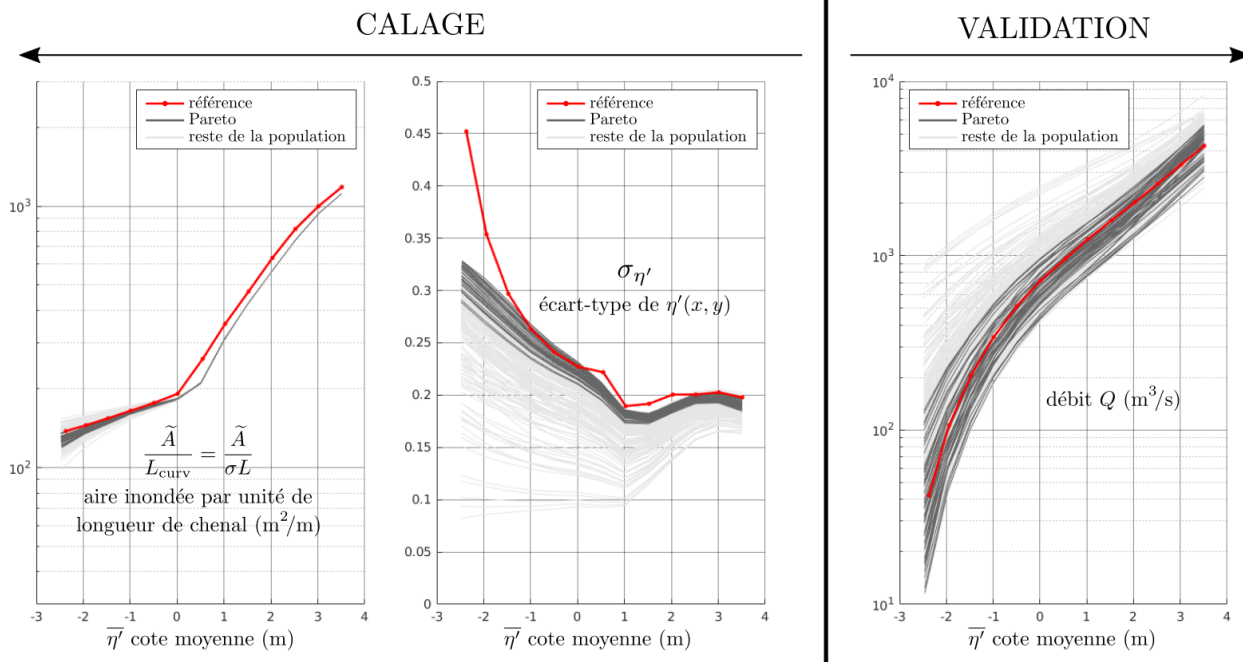


Figure 5.21: Signatures individuelles produites par l’ensemble des solutions testées. Les signatures en gris foncés sont produites par les solutions du front de Pareto, celles des autres solutions sont tracées en gris clair. Les signatures de la simulation de référence sont superposées en rouge.

Pour chaque paramètre, on a reporté sur l’axe horizontal (points) les valeurs individuelles obtenues dans la population, en distinguant comme précédemment les solutions dominées (en gris clair) et les solutions non-dominées, Pareto-optimales (en gris foncé). Sur chaque échantillon, une densité est ajustée par la méthode de *kernel smoothing* (fonction `ksdensity()` sous Matlab): il ne faut pas sur-interpréter leur aspect très lissé, mais elles donnent malgré tout une idée de la concentration des valeurs. On constate que les paramètres de géométrie sont bien contraints (profondeur maximale des radiers h_{run} et des mouilles h_{pool}), alors que les paramètres de rugosité restent très dispersés, avec une distribution peu resserrée comparée à la distribution uniforme a priori initiale. La distribution du paramètre h_{pool} est décalée vers des valeurs sensiblement supérieures à la valeur de la simulation de référence (9 m contre 6 m) : ce résultat n’est pas surprenant, à partir du moment où la profondeur des mouilles est assez supérieure à celle des radiers, elles deviennent des zones de pentes très faibles (dans la topographie réelle), et la valeur devient peu sensible. On constate en revanche que la profondeur des radiers, qui contrôle fortement l’écart-type $\sigma_{\eta'}$ pour les faibles débits, est très bien identifiée (distribution resserrée et centrée sur une valeur proche de 3 m).

5.3.3 Test d’inversion sur les simulations en régime stationnaire de la Garonne

Dans le second test d’inversion mené, nous avons cherché à reproduire les signatures hydrauliques sur le tronçon de la Garonne en considérant les régimes stationnaires simulés pour le calage du coefficient de frottement en lit mineur. Nous utilisons les simulations effectuées avec la valeur optimale identifiée de $K_{min} = 25 \text{ m}^{1/3} \cdot \text{s}^{-1}$, la valeur en lit majeur étant fixée à $K_{maj} = 15 \text{ m}^{1/3} \cdot \text{s}^{-1}$ (cette valeur n’a pas d’importance puisque seuls des débits non-débordants inférieurs à $1000 \text{ m}^3/\text{s}$ sont simulés). Ce second test d’inversion porte donc sur 3 paramètres : la profondeur de talweg dans les radiers h_{run} , la profondeur de talweg dans les mouilles h_{pool} , et le coefficient de Strickler en lit mineur K_{min} .

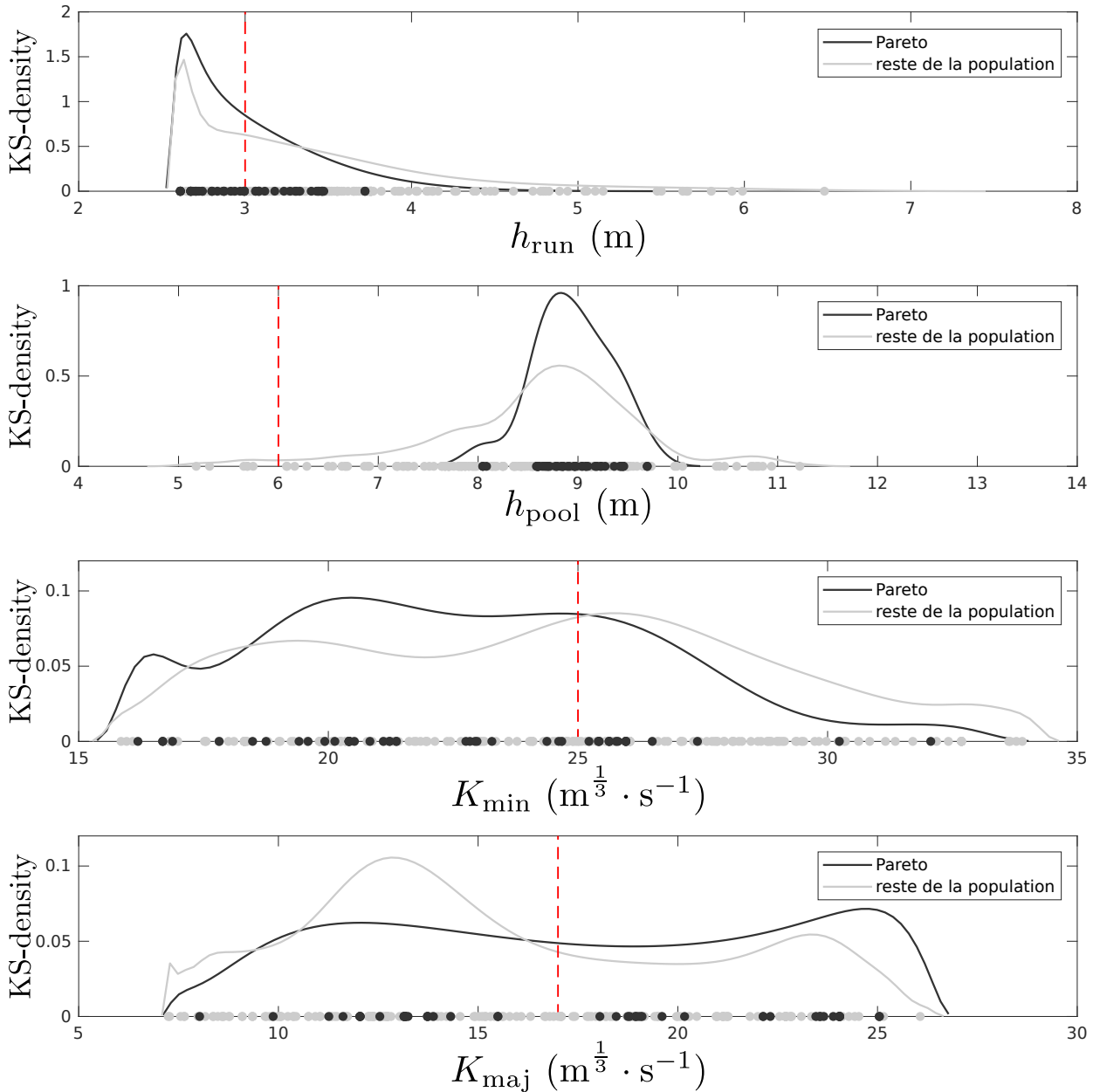


Figure 5.22: Distributions des paramètres du modèle dans la population en fin d'optimisation (gris clair), et dans le sous-ensemble des solutions non-dominées (gris foncé). La valeur utilisée dans la simulation de référence est indiquée par un pointillé rouge.

5.3.3.1 Analyse des signatures à reproduire

Les signatures hydrauliques d'aire inondée par unité de longueur de chenal $\frac{\tilde{A}}{\sigma_L}$, d'écart-type spatial $\sigma_{\eta'}$ et de tarage au tronçon ayant déjà été présentées au Chapitre 5 pour le tronçon de la Garonne, nous illustrons directement la comparaison avec les signatures simulées par un modèle périodique avec approximation de l'onde diffusante. Lorsque l'on conserve les critères utilisés pour le test précédent, on obtient typiquement les allures de signatures tracées sur la Figure 5.23 :

- l'exploration rapide de valeurs typiques des paramètres (h_{run} de l'ordre de 3 m, h_{pool} de l'ordre de 6 m et K_{min} entre 20 et 30) permet de reproduire assez facilement les courbes d'aire inondée et de débit,
- en revanche, ces valeurs de paramètres dans les gammes attendues ne permettent pas du tout de reproduire la signature d'écart-type spatial $\sigma_{\eta'}$ de la cote de la surface libre sans tendance η' . Alors que les valeurs observées atteignent 50 à 60 cm, le modèle périodique simule difficilement un écart-type de l'ordre de 20 cm.

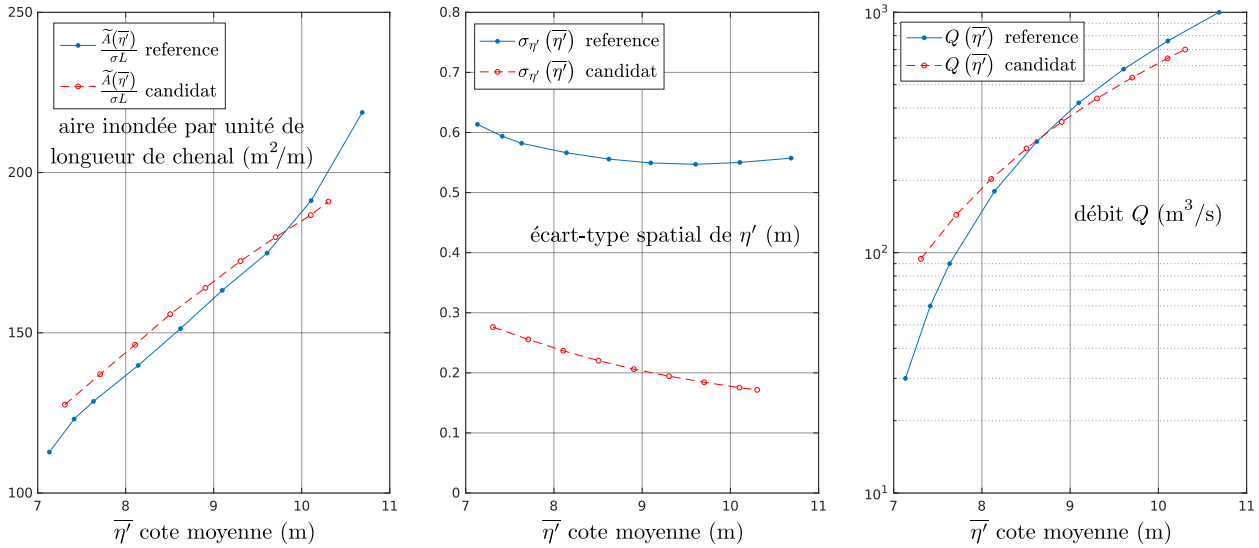


Figure 5.23: Signatures observées pour le tronçon de la Garonne et allure typique des signatures simulées par le modèle périodique (le jeu de paramètres illustré ici est généré aléatoirement et n'a aucune caractéristique optimale). On constate que l'écart-type spatial de η' simulé est très inférieur aux valeurs observées.

5.3.3.2 Quelle variance pour la surface sans tendance η' ?

Pour comprendre l'origine du problème identifié à la section précédente et envisager une solution, il faut revenir à l'allure de la surface libre sans tendance $\eta'(x, y, Q)$ pour un débit donné ou bien, en simplifiant en 1D en ne regardant que la cote le long de la ligne de talweg, $\eta'(s, Q)$. Nous reproduisons à la Figure 5.24 les courbes déjà illustrées au Chapitre 4 (Figure 4.10) des états stationnaires simulés sur la Garonne pour des débits non-débordants, en y ajoutant un élément supplémentaire d'analyse.

Jusqu'à présent, nous avons décomposé la cote η en retirant la tendance linéaire de la topographie et en gardant l'écart η' à cette tendance, en supposant que les variations de η' (et les variations sous-jacentes de z'_b la cote topographique sans tendance) étaient principalement explicables par les alternances de faciès morphologiques (mouilles-radiers). Cette situation est bien sûr très simplificatrice : en réalité, il existe toute une gamme de variabilité basse fréquence qui se manifeste à des longueurs d'onde inférieures à la longueur totale du tronçon (44 km en abscisse curviligne), mais bien supérieure à la longueur d'onde de référence ($\lambda \approx 3700$ m) des alternances de faciès. Sur la Figure 5.24, nous avons ajusté à l'ensemble des courbes la meilleure tendance basse fréquence commune avec 6 harmoniques de la longueur totale ; insistons sur le fait que cette tendance est par construction identique à toutes les courbes et de moyenne nulle. Mathématiquement, cela revient à écrire :

$$\eta'(s, Q) = \bar{\eta}'(Q) + \eta'_{\text{BF}}(s) + \eta'_{\text{facies}}(s, Q)$$

avec les composantes $\eta'_{\text{BF}}(s)$ basse fréquence et $\eta'_{\text{facies}}(s, Q)$ toutes deux de moyenne nulle. Il apparaît clairement que lorsque la cote de talweg est durablement sous sa tendance régionale, alors la cote de la surface libre va être également durablement sous sa tendance régionale, quel que soit le faciès hydraulique dans lequel on se trouve (mouille ou radier) et quel que soit le débit. On peut alors exprimer la variance spatiale de η' avec la décomposition proposée, fonction du débit Q (\mathbb{E} désigne l'espérance de la variable aléatoire, dans un sens spatial ici) :

$$\begin{aligned} \sigma_{\eta'}^2(Q) &= \mathbb{E} \left[\left(\eta' - \bar{\eta}' \right)^2 \right] \\ &= \mathbb{E} \left[\left(\eta'_{\text{BF}}(s) + \eta'_{\text{facies}}(s, Q) \right)^2 \right] \\ &= \mathbb{E} \left[\eta_{\text{BF}}'^2(s) + \eta_{\text{facies}}'^2(s, Q) + 2 \eta'_{\text{facies}}(s, Q) \eta'_{\text{BF}}(s) \right] \\ &= \mathbb{E} \left[\eta_{\text{BF}}'^2(s) \right] + \mathbb{E} \left[\eta_{\text{facies}}'^2(s, Q) \right] + 2 \mathbb{E} \left[\eta'_{\text{facies}}(s, Q) \eta'_{\text{BF}}(s) \right] \end{aligned}$$

Les composantes η'_{BF} et η'_{facies} n'évoluant pas sur les mêmes échelles spatiales, il est clair le dernier terme dans l'équation précédente est quasi nul (espérance du produit de deux termes de moyenne nulle et très peu corrélés entre eux). On en déduit donc :

$$\sigma_{\eta'}^2(Q) \simeq \sigma_{\eta'_{\text{facies}}}^2(Q) + \sigma_{\eta'_{\text{BF}}}^2 \simeq \sigma_{\eta'_{\text{facies}}}^2(Q) + \text{constante}$$

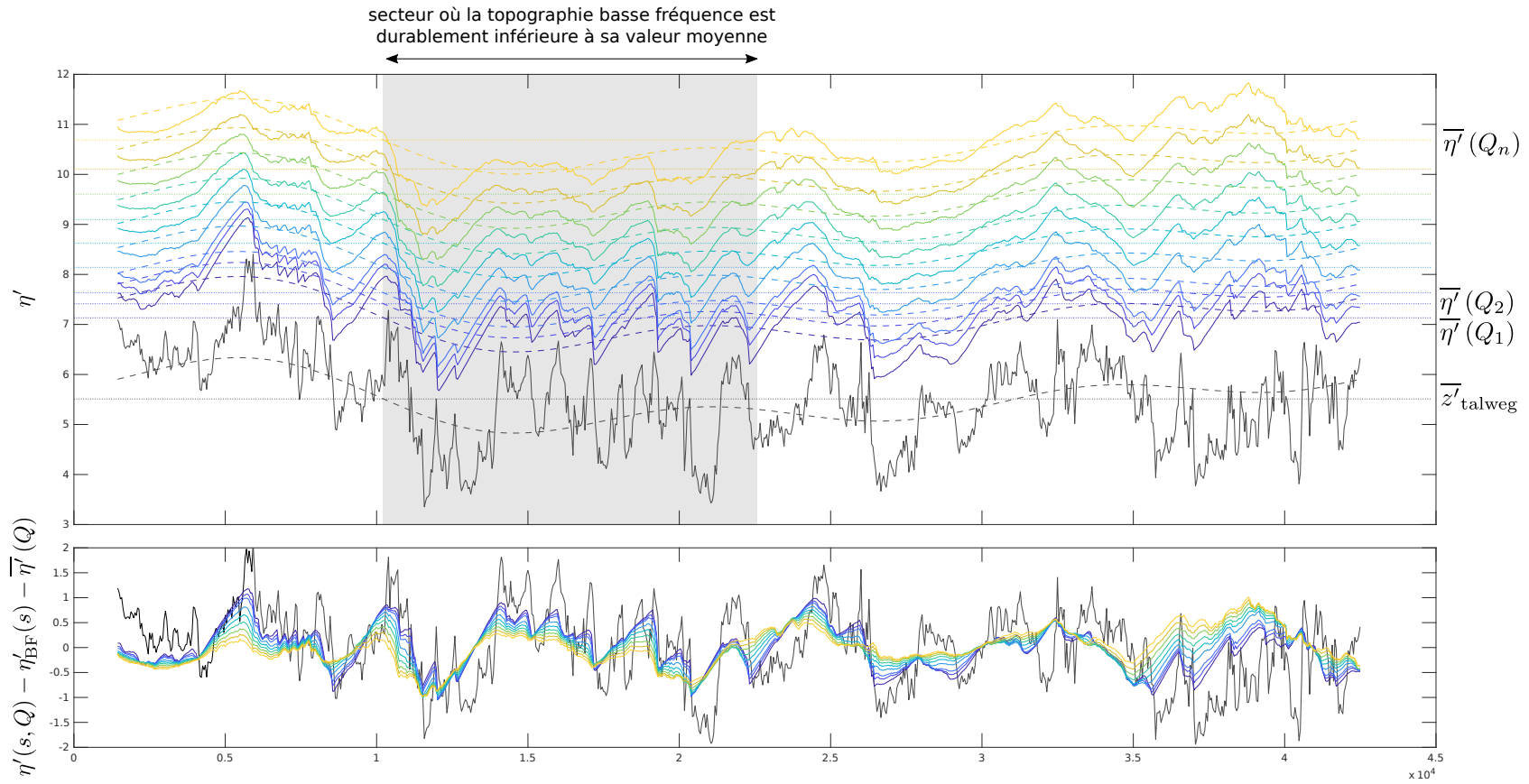


Figure 5.24: Lignes d'eau simulées en régime permanent sur la Garonne pour des débits non-débordants, en variable η' sans tendance régionale. Sur le graphique du haut, une tendance basse fréquence commune à toutes les lignes d'eau est ajustée (les lignes en tiretés sont des translations verticales les unes des autres). Cela illustre qualitativement que cette tendance basse fréquence (à des échelles supérieures à la longueur d'onde des alternances mouilles-radriers) est relativement indépendante du débit. Le graphique du bas montre alors la composante $\eta'_{\text{facies}}(s, Q) = \eta'(s, Q) - \overline{\eta'}(Q) - \eta'_{\text{BF}}(s)$ majoritairement liée aux alternances morphologiques.

Pour illustrer le problème d'augmentation de la variance de η' , nous avons ajusté un modèle de la variabilité basse fréquence sur les lignes d'eau simulées, mais ce n'est en fait pas nécessaire (ce ne serait de toute façon pas réalisable de façon robuste sur des données de type SWOT) : la conclusion est simplement que l'augmentation de variance par rapport à la seule variance produite par les alternances de faciès peut raisonnablement être considérée comme indépendante des conditions d'écoulement, définies soit en termes de débit Q au tronçon soit en termes de cote moyenne $\bar{\eta}'(Q)$ correspondante. Comme le modèle périodique ne peut simuler que la composante $\sigma_{\eta'_{\text{facies}}}^2(x, y, Q)$ de la variance spatiale totale, une solution simple est de forcer les deux signatures de variance observée et simulée à coïncider en un point. Un point pivot naturel est de les faire coïncider à l'abscisse correspondant aux conditions de plein-bord, c'est-à-dire de comparer la courbe $\sigma_{\eta'}^2(\bar{\eta}') - \sigma_{\eta'}^2(\bar{\eta}' = \eta'_{\text{bf}})$ observée, avec cette même courbe tirée de la simulation périodique. Pour mémoire, le niveau η'_{bf} (*bankfull*) est une signature exclusivement topographique (non liée à une surface en eau réelle) qui peut être déterminée de façon relativement robuste même en présence d'un seuil de visibilité. Le nouveau critère est alors la RMSE sur cette courbe signature (cf. Figure 5.25).

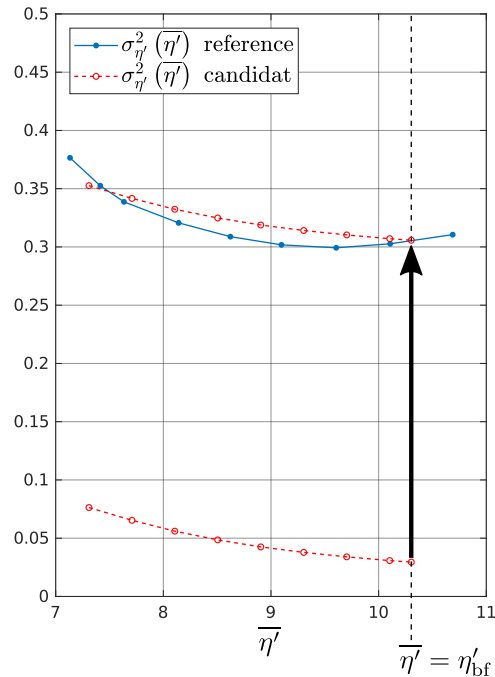


Figure 5.25: Redéfinition du critère de calage sur signature de variabilité spatiale de la cote de la surface libre sans tendance η' . Ce sont maintenant les courbes de variance observée et simulée qui sont comparées, et non plus les courbes d'écart-type : en effet, ce sont les variances des composantes spatiales de $\eta'(x, y, Q)$ qui sont additives, et non pas les écarts-type. On corrige le biais de variance dû à l'existence d'une composante de variabilité basse fréquence en forçant les variances simulées et observées à la même valeur pour l'abscisse $\bar{\eta}' = \eta'_{\text{bf}}$ correspondant au niveau de plein bord déduit de la topographie.

5.3.3.3 Résultats

Comme dans le cas de géométrie synthétique précédent, nous effectuons une optimisation multi-objectif à l'aide de CaRaMEL sur les signatures d'aire inondée et de variance de η' . La présentation suit le même ordre que pour le premier test d'inversion où la géométrie de référence (à inverser) était périodique : analyse du front de Pareto, analyse des faisceaux de signatures simulées par les solutions individuelles, et finalement distribution des jeux de paramètres.

La Figure 5.26 illustre le front de Pareto obtenu en fin d’optimisation, avec comme précédemment :

- sur la partie gauche, la distinction entre solutions dominées d’une part (en gris clair), et solutions non-dominées optimales au sens de Pareto d’autre part (en noir),
- sur la partie droite, l’évaluation de chaque solution sur le critère de restitution de la courbe de tarage au tronçon : plus les points sont gros et de couleur chaude, meilleure est la solution en termes de relation hauteur-débit.

Nous pouvons constater comme précédemment que les signatures hydrauliques $\frac{\tilde{A}(\bar{\eta}')}{L_{\text{curv}}}$ et $\sigma_{\eta'}^2(\bar{\eta}')$ contiennent bien une information exploitable pour l’estimation des débits, puisque les points situés dans le front de Pareto ont des scores sur le débit qui sont globalement meilleurs que les autres solutions. Cela se confirme lorsque l’on regarde les faisceaux de signatures simulées par l’ensemble des solutions, illustrés sur la Figure 5.26.

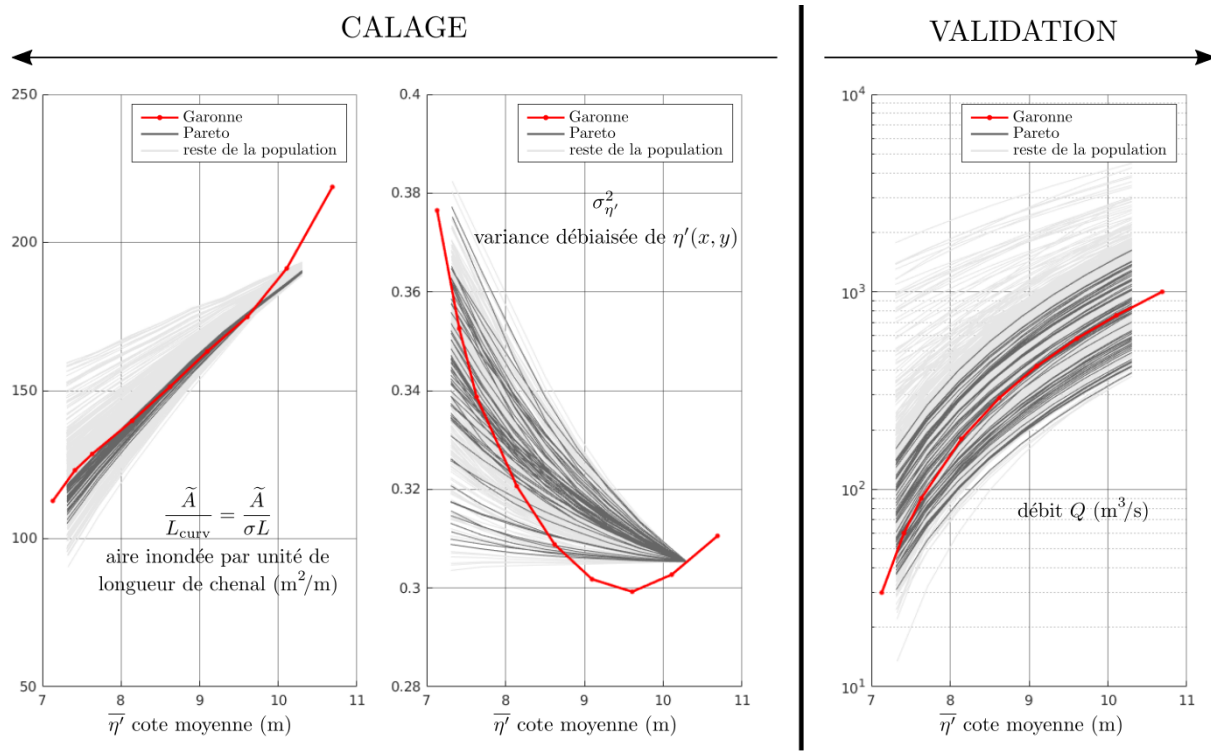


Figure 5.26: Signatures individuelles produites par l’ensemble des solutions testées. Les signatures en gris foncé sont produites par les solutions du front de Pareto, celles des autres solutions sont tracées en gris clair. Les signatures données par les simulations stationnaires sur la Garonne sont superposées en rouge.

Enfin, sur la Figure 5.27 nous montrons, comme pour le test en géométrie de référence synthétique périodique, la distribution des paramètres en fin d’optimisation, en distinguant la distribution parmi les solutions dominées (points et densités *kernel-smoothed* en gris clair) et parmi les solutions non-dominées Pareto-optimales (points et densités en gris foncé). Comme pour le test précédent, la convergence est très bonne pour les paramètres de géométrie, avec des valeurs tout à fait représentatives de la variabilité morphologique du lit de la Garonne : profondeur au talweg centrée sur une valeur $h_{\text{run}} = 3.8$ m pour les radiers, $h_{\text{pool}} = 5.7$ m pour les mouilles. Pour mémoire, ces valeurs sont définies par rapport au niveau de plein, qui vaut $\eta'_{\text{bf}} = 10.3$ m dans la topographie sans tendance pour la Garonne. Nous pouvons donc convertir ces valeurs en niveau z'_b :

$$\begin{cases} z'_{talweg,run} \approx \eta'_{bf} - h_{run} = 6.5 \text{ m} \\ z'_{talweg,pool} \approx \eta'_{bf} - h_{run} = 4.6 \text{ m} \end{cases}$$

Ces valeurs et l'amplitude correspondante d'environ 2 mètres entre la crête des radiers et le fond des mouilles correspondent bien à ce que l'observe sur la figure 5.24 par exemple. En revanche, comme pour le cas synthétique, l'optimisation échoue à produire une estimation resserrée du coefficient de Strickler ; la distribution de K_{min} a posteriori ne diffère que très peu de la distribution uniforme utilisée a priori, que ce soit parmi les solutions dominées ou non-dominées en fin d'optimisation.

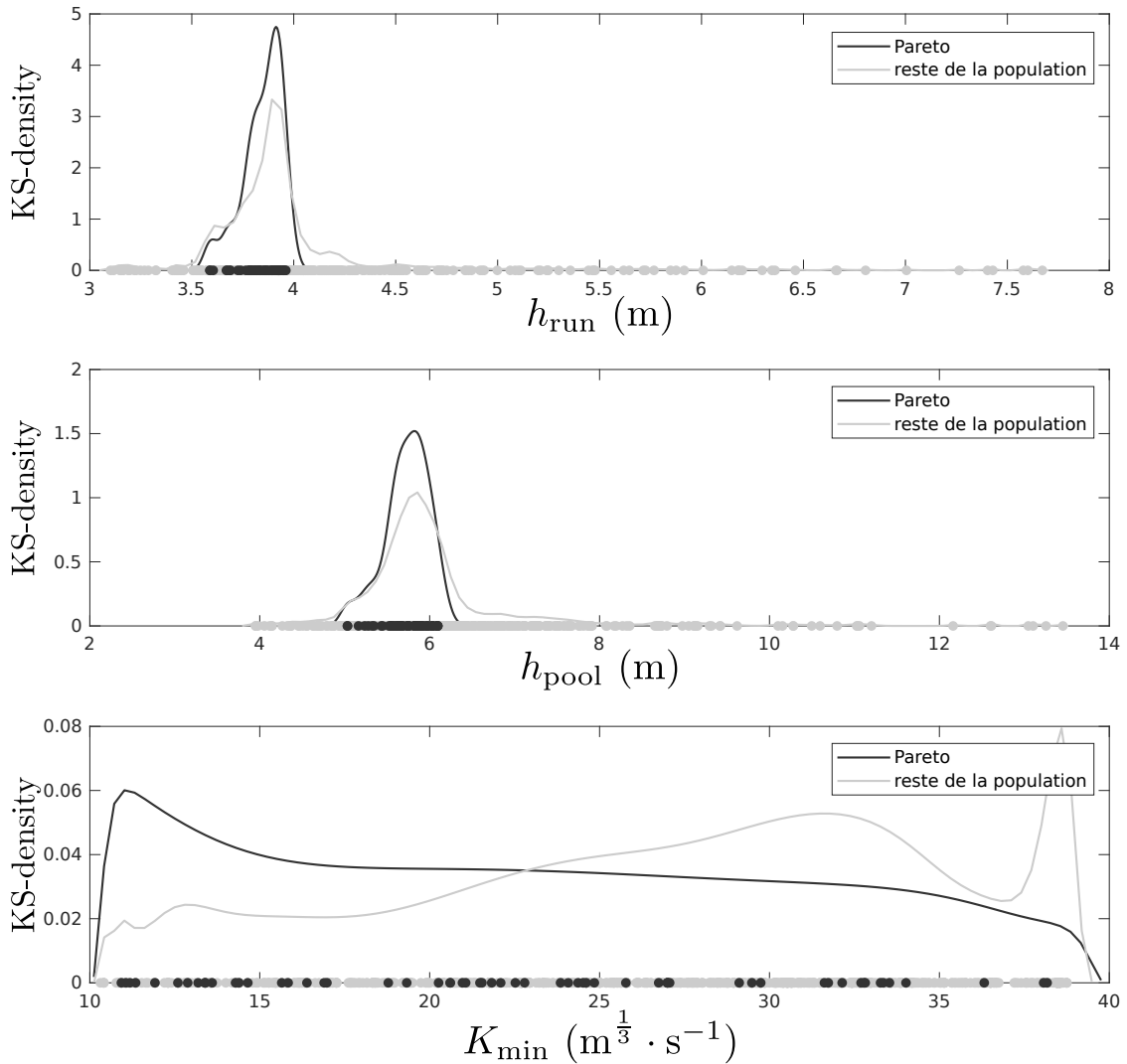


Figure 5.27: Distributions des paramètres du modèle dans la population en fin d'optimisation (gris clair), et dans le sous-ensemble des solutions non-dominées (gris foncé). La valeur $K_{min} = 25$ utilisée dans la simulation de référence sur la Garonne est indiquée par un pointillé rouge. K_{maj} n'était pas optimisé dans ce test ne portant que sur des débits non-débordants.

5.4 Conclusions et perspectives

Dans ce chapitre, nous avons proposé une modélisation au tronçon en régime stationnaire reposant sur des hypothèses de périodicité spatiale de la géométrie, à la fois en lit mineur et en lit majeur. La paramétrisation choisie a pour but de reproduire avec peu de degrés de liberté les signatures topographiques définies au Chapitre 4 : sinuosité, distribution statistique des cotes bathymétrique / topographique en lit mineur / majeur, etc. Ce modèle se focalise sur la reproduction de la variabilité morphologique associée à la longueur d’onde caractéristique λ des alternances de faciès morphologiques (mouilles, radiers) : les variations plus hautes fréquences (“bruit” sur cette régularité moyenne) ainsi que plus basses fréquences (sur des longueurs caractéristiques de plusieurs fois λ) sont ignorées. Mais les analyses menées aux chapitres 4 et 5 semblent montrer que c’est bien cette variabilité à l’échelle caractéristique λ qui contrôle les variations de la signature de variance $\sigma_{\eta'}^2$, à une constante près. L’optimisation multi-objectif sur les signatures d’aire inondée par unité de longueur de chenal d’une part, et de variance de η' d’autre part, a permis d’identifier des paramètres vraisemblables pour la profondeur au talweg h_{run} des radiers et celle h_{pool} des mouilles. L’identification des paramètres de rugosité (K_{min} et K_{maj} dans le premier test en géométrie synthétique, K_{min} seulement pour le cas des débits non-débordants de la Garonne) s’est en revanche avérée être un échec, la distribution a posteriori de ces paramètres dans les fronts de Pareto n’étant pas très différente de la distribution uniforme utilisée pour générer les solutions initiales. Il semblerait plus raisonnable d’utiliser un prior plus resserré pour ces paramètres (ou des valeurs issues d’une classification régionale) pour limiter la dispersion des courbes de tarage au tronçon résultantes ; malgré tout, cet ensemble des signatures $Q(\overline{\eta'})$ générées par les solutions non-dominées des fronts de Pareto ont une dispersion nettement plus réduite que la dispersion du reste des solutions (quantifiant la dispersion a priori).

L’un des points encourageants du second test d’inversion en bathymétrie naturelle sur la Garonne est la capacité de la méthode d’optimisation à capturer la variabilité morphologique de faciès en exploitant la signature de variance de η' , après avoir corrigé cette variance du biais lié à l’existence de variabilités à d’autres échelles. Une telle décomposition sera sans aucun doute indispensable pour exploiter la variance de η' acquise par SWOT ; en effet, une composante de variance de l’erreur instrumentale viendra s’ajouter :

$$\sigma_{\eta'}^2(Q) \simeq \sigma_{\eta'_{\text{facies}}}^2(Q) + \sigma_{\eta'_{\text{BF}}}^2 + \sigma_{\text{instru}}^2$$

La possibilité d’exploiter cette signature sur les données SWOT dépendra fortement de l’importance en ordre de grandeur de cette composante, mais aussi de sa dépendance aux conditions d’écoulement. Si cette variance s’avère assez stable ou prédictible a priori indépendamment des caractéristique de la surface d’eau, alors la signature pourrait être exploitable. Si en revanche elle est elle-même une fonction $\sigma_{\text{instru}}^2(Q)$ (ou $\sigma_{\text{instru}}^2(\overline{\eta'}(Q))$) en variable $\overline{\eta'}$, l’identifiabilité des paramètres sera très réduite.

Enfin, toute la procédure d’inversion doit encore être évaluée sur des états observés transitoires, extraits de la simulation de référence sur la crue de juin 2000 de la Garonne en cours de finalisation au moment de l’écriture de ce chapitre.

Chapitre 6

Conclusion générale

Malgré des siècles de progrès dans la mesure, la compréhension et la prévision du cycle de l'eau, les hydrologues d'aujourd'hui se heurtent à un obstacle majeur que représente la difficulté d'accès aux données de débit des rivières du monde entier. En effet, les réseaux de jaugeage *in situ* dans de nombreuses régions ont stagné ou diminué en taille au cours des dernières décennies. Par ailleurs, l'étude des cours d'eau de bassins versants transfrontaliers est gênée par le manque d'un accès libre aux mesures (*Hannah et al.*, 2011; *Gerlak et al.*, 2011). Les mesures *in situ* du débit des cours d'eau fournissent des jeux de données précises, fiables et temporellement continues pour la gestion des ressources, la prévision des inondations et l'hydrologie des bassins versants. Ces mesures consistent essentiellement en des courbes de tarage, qui décrivent une relation entre la hauteur d'eau et le débit. L'amélioration de la surveillance du débit pour diverses conditions d'écoulement sur les sites fluviaux peut être réalisée en tirant parti des nouvelles technologies développées pour les mesures au sol et la télédétection, rendant ainsi service à la compréhension des processus hydrologiques de surface à l'échelle du bassin versant.

Plusieurs méthodes ont été élaborées pour estimer les débits à partir de différents types de données (vitesses, cotes d'eau, etc.). Parmi celles-ci, on peut citer les méthodes inverses qui se basent sur des modèles hydrauliques en utilisant des données altimétriques. L'estimation du débit dépend de plusieurs paramètres inconnus et difficiles à mesurer (bathymétrie du lit mineur, frottements, conditions aux limites). Pour résoudre ce problème, différentes simplifications de la bathymétrie ont été proposées et peuvent être soit explicites, soit implicites. La majorité de ces approches utilisent des données *a priori* dans un modèle 1D en négligeant les interactions entre le lit mineur et la plaine inondable. Ainsi, une approche simple basée sur une géométrie explicite bidimensionnelle trouve un intérêt dans le contexte des tronçons non-jaugés.

6.1 Modèle périodique 2D comme approche

Pour réduire la dimensionnalité du problème inverse dans le cas non-jaugé, nous proposons une approche explicite non-uniforme bidimensionnelle de la bathymétrie (cf. Chapitre 5). Ce modèle géométrique consiste à représenter la variabilité morphologique du tronçon par des fonctions strictement périodiques.

Dans le Chapitre 2 et dans l'article *Mahdade et al.* (2020), nous avons analysé, à l'aide des ondelettes, la variabilité morphologique de quelques tronçons de petites rivières. Cette étude justifie le choix de l'approche périodique et la corrélation entre les variables géométriques (transversales et de la forme en plan) et hydrauliques. La simplification périodique de la morphologie présente un degré de complexité supérieure à une simplification uniforme, mais elle reste soumise à plusieurs hypothèses :

- elle n'a de sens qu'en régime stationnaire,
- elle suppose l'absence d'apports intermédiaires.

Dans le Chapitre 5, nous avons décrit le modèle périodique 2D. Celui-ci est composé d'un modèle géométrique équivalent et d'un solveur de Saint-Venant (Basilisk). La géométrie équivalente est représentée par un petit nombre de paramètres dont les inconnues sont les deux coefficients de frottement, la profondeur maximale des radiers ainsi que l'approfondissement maximal des mouilles relativement aux radiers.

La construction du modèle périodique équivalent au tronçon réel suppose la connaissance de quelques paramètres morphologiques et topographiques, notamment le niveau de plein bord au tronçon, à partir des observables télédétections. Ce niveau correspond au maximum du rayon hydraulique équivalent au tronçon et permet de définir la largeur de plein bord équivalente. La paramétrisation suppose également une connaissance de la distribution des altitudes en lit majeur. Cette géométrie simplifiée permet de reproduire les signatures topographiques du tronçon réel présentées dans le Chapitre 4. Plusieurs signatures relatives à la surface libre d'un écoulement présent dans le tronçon, telles que l'aire inondée par unité de longueur de lit mineur \tilde{A} en fonction de la cote moyenne $\bar{\eta}'$ de la surface sans tendance, ainsi que la variance $\sigma_{\eta'}$ de cette cote, ont ensuite été analysées, et liées à la courbe de tarage au tronçon constituée par la courbe signature $Q(\bar{\eta}')$. Dans les tests d'inversion présentés dans le Chapitre 5, l'optimisation multi-objective à l'aide de l'algorithme CaRaMEL a montré que les signatures d'aire inondée et de variance de η' contiennent bien de l'information relative à la morphologie du tronçon, et que cette information peut être utilisée pour inférer certains paramètres du modèle périodique équivalent du tronçon. Cette inversion a été possible à la fois quand la géométrie et les états hydrauliques étaient eux-mêmes issus d'une référence périodique, mais aussi pour la géométrie et les états issus des simulations stationnaires sur la Garonne. Si les paramètres de profondeur maximale et minimale au talweg sont assez bien identifiables, il n'en est pas de même pour les coefficients de frottement. Ce résultat semble rejoindre la conclusion d'autres études (*Oubanas et al.*, 2018a; *Durand et al.*, 2016; *Garambois et Monnier*, 2015) montrant que, du fait de problèmes d'équifinalité, l'inversion de tous les paramètres inconnus n'est pas possible simultanément. Dans le cas d'une approche périodique, un *a priori* sur les coefficients de frottements pourrait permettre de mieux encadrer la courbe de tarage au tronçon résultante. Serait-il plus judicieux de combiner plusieurs sources de données télédétections pour avoir plus d'informations sur le tronçon ? Les données SWOT sont-elles la bonne application de cette approche ou bien une autre application (par exemple, LSPIV) ?

Ce premier test préliminaire du modèle est une démonstration de faisabilité qui ouvre sur plusieurs perspectives :

- il serait intéressant et nécessaire de tester l'approche sur tout le tronçon de 40 km et sur les états issus de la simulation transitoire, pour évaluer l'impact de l'hypothèse de stationnarité nécessaire dans l'approche périodique ;
- nous n'avons pas ajouté de simulation du bruit instrumental dans ces tests. C'est un aspect très important pour les applications à des observables télédétections, il sera donc nécessaire d'aborder cette question dans les développements à court terme, par exemple en bruitant la simulation de référence avec le simulateur SWOT ;
- au-delà de l'approche périodique, nous pourrions nous baser sur l'utilisation de la teinte multi-spectrale \underline{U} définie au Chapitre 3.3 pour générer de la variabilité bathymétrique à l'intérieur même de la procédure d'inversion : il s'agirait alors de considérer tout ou partie des paramètres décrivant la relation $z_b = f(\underline{U})$ comme faisant partie des paramètres à inverser (avec les coefficients de rugosité), plutôt que de les caler indépendamment sur des données *in situ* de profils en travers;

La poursuite de ces travaux sur l'approche périodique n'est pas limitée uniquement aux études liées à l'estimation des débits. En effet, le modèle périodique permet également de revenir sur des questions de géomorphologie des rivières en analysant la variabilité des courbes de tarage sur différentes sections pour voir si elles suivent une relation de type AMHG. De plus, la description non-uniforme de la bathymétrie (variabilité transversale corrélée avec une variabilité de la forme en plan) pourrait être intéressante pour tester certaines hypothèses sur les processus de géomorphologie fluviale, comme les changements de zones

préférentielles d'érosion en fonction du caractère débordant ou non de l'écoulement (hypothèse de *flow convergence* de *MacWilliams Jr et al.*, 2006).

6.2 Basilisk comme outil de modélisation hydraulique

Au début de la thèse, le choix de l'outil approprié pour mettre en oeuvre le modèle périodique a été l'objet de longs questionnements. C'est finalement le modèle Basilisk qui a été choisi, choix motivé par son avantage de raffinement adaptatif des mailles pour réduire le temps de calcul ainsi que sa gestion très paramétrisable des conditions aux limites. Basilisk ne possède pas d'interface qui faciliterait la visualisation des entrées et sorties (à la différence des autres modèles hydrauliques tels HEC-RAS, TELEMAC, etc.) ; il est programmé en langage C et ceci m'a demandé un peu de temps pour me familiariser avec ce nouvel outil et pour élaborer l'architecture du modèle périodique. Mais en contrepartie, cet outil de programmation permet d'accéder à des fonctions d'assez bas niveau qui permettent de créer des configurations particulières, de gérer comme on le souhaite les termes sources (exemple de la topographie sans tendance, etc.). Nous disposons désormais d'une bibliothèque d'utilitaires programmés sur Basilisk et qui sont généralement liés aux développements que nous avons présentés dans le Chapitre 3 et le Chapitre 5. Ces codes sont fournis dans l'Annexe C et comprennent :

- la gestion des fichiers d'entrée / sortie du modèle compatible avec la parallélisation (la bibliothèque `esri-binary-grids.h`) ;
- la gestion des injections des débits également compatible avec la parallélisation (la bibliothèque `inflow-tilt.h`) ;
- l'implémentation du terme source de Manning (la bibliothèque `manning-tilt.h`) ;
- la construction de la topographie périodique (les bibliothèques : `kinoshita.h` et `live_terrain.h`) pour générer la topographie "à la volée" ;
- une implémentation de l'approximation de l'onde diffusante à l'aide du solveur de Poisson (exemple `diffwave.c`).

Etant donné les gains en temps de calcul, il serait intéressant de tester l'approximation de l'onde pour produire une version sans inertie de la simulation de référence sur la Garonne. Cela pourrait ouvrir de nouvelles perspectives d'inversion, notamment avec la teinte multispectrale.

6.3 Traitements et jeux de données produits pendant la thèse

En plus des développements algorithmiques dans Basilisk, cette thèse a également été l'occasion de proposer des méthodes d'analyse spatiale qui pourront être appliquées à d'autres secteurs d'étude et aider à la caractérisation régionale des cours d'eau. Plus qu'un jeu de données définitif, la construction de la simulation de référence sur la Garonne donne un cadre méthodologique qui pourra évoluer, notamment :

- par l'enrichissement du jeu de données *in situ* de profils en travers, avec des levés plus récents,
- par l'utilisation d'autres jeux de données multispectraux, comme celles des satellites Landsat ou Sentinel,
- par une poursuite des tests de valeurs de coefficient de rugosité en lit mineur et majeur, avec en particulier l'idée de les spatialiser le long du tronçon.
- par la superposition d'un bruit simulant le bruit instrumental de l'altimètre SWOT, à l'aide du simulateur dédié.

Ce jeu de données est également susceptible d'être utilisé pour tester d'autres algorithmes d'inversion que ceux proposés dans cette thèse, et réciproquement des jeux de données similaires à haute résolution pourraient être produits sur de grands fleuves avec cette méthodologie. De tels jeux de données sont encore relativement peu nombreux à l'heure actuelle dans les travaux autour de la mission SWOT.

Bibliographie

- Abramian, A. (2015), Self organisation of sediment transport in alluvial rivers, Thèse de doctorat, Sorbonne Paris Cité.
- Afshari, S., A. A. Tavakoly, M. A. Rajib, X. Zheng, M. L. Follum, E. Omranian, et B. M. Fekete (2018), Comparison of new generation low-complexity flood inundation mapping tools with a hydrodynamic model, *Journal of Hydrology*, 556, 539–556.
- Alsdorf, D. E., et D. P. Lettenmaier (2003), Tracking fresh water from space, *Science*, 301(5639), 1491–1494.
- Altenau, E. H., T. M. Pavelsky, D. Moller, C. Lion, L. H. Pitcher, G. H. Allen, P. D. Bates, S. Calmant, M. Durand, et L. C. Smith (2017), Airswot measurements of river water surface elevation and slope: Tanana river, ak, *Geophysical Research Letters*, 44(1), 181–189.
- Audusse, E., F. Bouchut, M.-O. Bristeau, R. Klein, et B. t. Perthame (2004), A fast and stable well-balanced scheme with hydrostatic reconstruction for shallow water flows, *SIAM Journal on Scientific Computing*, 25(6), 2050–2065.
- Bartsch, A., W. Wagner, K. Scipal, C. Pathe, D. Sabel, et P. Wolski (2009), Global monitoring of wetlands—the value of envisat asar global mode, *Journal of environmental management*, 90(7), 2226–2233.
- Bates, P. D., M. S. Horritt, et T. J. Fewtrell (2010), A simple inertial formulation of the shallow water equations for efficient two-dimensional flood inundation modelling, *Journal of Hydrology*, 387(1-2), 33–45.
- Beven, K. (2006), A manifesto for the equifinality thesis, *Journal of Hydrology*, 320(1), 18–36, doi:10.1016/j.jhydrol.2005.07.007.
- Biancamaria, S., M. Durand, K. Andreadis, P. Bates, A. Boone, N. Mognard, E. Rodriguez, D. Alsdorf, D. Lettenmaier, et E. Clark (2011), Assimilation of virtual wide swath altimetry to improve arctic river modeling, *Remote Sensing of Environment*, 115(2), 373–381.
- Biancamaria, S., D. P. Lettenmaier, et T. M. Pavelsky (2016), The swot mission and its capabilities for land hydrology, *Surveys in Geophysics*, 37(2), 307–337.
- Birkett, C. (1995), The contribution of topex/poseidon to the global monitoring of climatically sensitive lakes, *Journal of Geophysical Research: Oceans*, 100(C12), 25,179–25,204.
- Birkett, C. M. (1998), Contribution of the topex nasa radar altimeter to the global monitoring of large rivers and wetlands, *Water Resources Research*, 34(5), 1223–1239.
- Blöschl, G., et M. Sivapalan (1995), Scale issues in hydrological modelling: a review, *Hydrological processes*, 9(3-4), 251–290.
- Bonnema, M. G., S. Sikder, F. Hossain, M. Durand, C. J. Gleason, et D. M. Bjerklie (2016), Benchmarking wide swath altimetry-based river discharge estimation algorithms for the ganges river system, *Water Resources Research*, 52(4), 2439–2461.

- Buchanan, T. J., et W. P. Somers (1969), *Discharge measurements at gaging stations*, US Government Printing Office Washington, DC.
- Calmant, S., et F. Seyler (2006), Continental surface waters from satellite altimetry, *Comptes Rendus Geoscience*, 338(14-15), 1113–1122.
- Chandelier, L., G. Martinoty, et al. (2009), Radiometric aerial triangulation for the equalization of digital aerial images and orthoimages, *Photogramm. Eng. Remote Sens*, 75(2), 193–200.
- Church, M. (2006), Bed material transport and the morphology of alluvial river channels, *Annu. Rev. Earth Planet. Sci.*, 34, 325–354.
- Crétaux, J.-F., et C. Birkett (2006), Lake studies from satellite radar altimetry, *Comptes Rendus Geoscience*, 338(14-15), 1098–1112.
- Crétaux, J.-F., M. Bergé-Nguyen, M. Leblanc, R. Abarca Del Rio, F. Delclaux, N. Mognard, C. Lion, R. K. Pandey, S. Tweed, S. Calmant, et al. (2011), Flood mapping inferred from remote sensing data, *Int. Water Technol. J.*, 1, 48–62.
- Creutin, J., M. Muste, A. Bradley, S. Kim, et A. Kruger (2003), River gauging using piv techniques: a proof of concept experiment on the iowa river, *Journal of Hydrology*, 277(3-4), 182–194.
- David, M. (2016), Dynamique fluviale de la garonne à l’anthropocène: trajectoire d’évolution du tronçon fluvial compris entre les confluences de l’ariège et du tarn (garonne toulousaine, 90 km), Thèse de doctorat.
- de Lavenne, A., V. Andréassian, G. Thirel, M.-H. Ramos, et C. Perrin (2019), A regularization approach to improve the sequential calibration of a semidistributed hydrological model, *Water Resources Research*, 55(11), 8821–8839, doi:<https://doi.org/10.1029/2018WR024266>.
- de Oliveira Campos, I., F. Mercier, C. Maheu, G. Cochonneau, P. Kosuth, D. Blitzkow, et A. Cazenave (2001), Temporal variations of river basin waters from topex/poseidon satellite altimetry. application to the amazon basin, *Comptes Rendus de l’Académie des Sciences-Series IIA-Earth and Planetary Science*, 333(10), 633–643.
- Deutsch, C. V., et L. Wang (1996), Hierarchical Object-Based Stochastic Modeling of Fluvial Reservoirs, *Mathematical Geology*, 28(7).
- Domeneghetti, A., G.-P. Schumann, R. Frasson, R. Wei, T. Pavelsky, A. Castellarin, A. Brath, et M. Durand (2018), Characterizing water surface elevation under different flow conditions for the upcoming swot mission, *Journal of Hydrology*, 561, 848–861.
- Doocy, S., A. Daniels, S. Murray, et T. D. Kirsch (2013), The human impact of floods: a historical review of events 1980-2009 and systematic literature review, *PLoS currents*, 5.
- Durand, M., L.-L. Fu, D. P. Lettenmaier, D. E. Alsdorf, E. Rodriguez, et D. Esteban-Fernandez (2010), The surface water and ocean topography mission: Observing terrestrial surface water and oceanic submesoscale eddies, *Proceedings of the IEEE*, 98(5), 766–779.
- Durand, M., J. Neal, E. Rodríguez, K. M. Andreadis, L. C. Smith, et Y. Yoon (2014), Estimating reach-averaged discharge for the river severn from measurements of river water surface elevation and slope, *Journal of Hydrology*, 511, 92–104.
- Durand, M., C. Gleason, P.-A. Garambois, D. Bjerklie, L. Smith, H. Roux, E. Rodriguez, P. D. Bates, T. M. Pavelsky, J. Monnier, et al. (2016), An intercomparison of remote sensing river discharge estimation algorithms from measurements of river height, width, and slope, *Water Resources Research*, 52(6), 4527–4549.

- Esteban-Fernandez, D. (2013), Swot project: Mission performance and error budget. revision a, nasa, *Rapp. tech.*, JPL Tech. Rep. JPL D-79084, 83 pp.[Available online at <http://swot.jpl.nasa.gov>
- Fjørtoft, R., J.-M. Gaudin, N. Pourthié, J.-C. Lalaurie, A. Mallet, J.-F. Nouvel, J. Martinot-Lagarde, H. Oriot, P. Borderies, C. Ruiz, et al. (2013), Karin on swot: Characteristics of near-nadir ka-band interferometric sar imagery, *IEEE Transactions on Geoscience and Remote Sensing*, *52*(4), 2172–2185.
- Frappart, F., S. Calmant, M. Cauhopé, F. Seyler, et A. Cazenave (2006), Preliminary results of envisat ra-2-derived water levels validation over the amazon basin, *Remote sensing of Environment*, *100*(2), 252–264.
- Frasson, R. P. d. M., R. Wei, M. Durand, J. T. Minear, A. Domeneghetti, G. Schumann, B. A. Williams, E. Rodriguez, C. Picamilh, C. Lion, et al. (2017), Automated river reach definition strategies: Applications for the surface water and ocean topography mission, *Water Resources Research*, *53*(10), 8164–8186.
- Fujita, I., M. Muste, et A. Kruger (1998), Large-scale particle image velocimetry for flow analysis in hydraulic engineering applications, *Journal of hydraulic Research*, *36*(3), 397–414.
- Fulton, J., et J. Ostrowski (2008), Measuring real-time streamflow using emerging technologies: Radar, hydroacoustics, and the probability concept, *Journal of Hydrology*, *357*(1-2), 1–10.
- Garambois, P.-A., et J. Monnier (2015), Inference of effective river properties from remotely sensed observations of water surface, *Advances in Water Resources*, *79*, 103–120.
- Gejadze, I., et P.-O. Malaterre (2017), Discharge estimation under uncertainty using variational methods with application to the full saint-venant hydraulic network model, *International Journal for Numerical Methods in Fluids*, *83*(5), 405–430.
- Gejadze, I. Y., et P.-O. Malaterre (2016), Design of the control set in the framework of variational data assimilation, *Journal of Computational Physics*, *325*, 358–379.
- Gerlak, A. K., J. Lautze, et M. Giordano (2011), Water resources data and information exchange in trans-boundary water treaties, *International Environmental Agreements: Politics, Law and Economics*, *11*(2), 179–199.
- Giustarini, L., P. Matgen, R. Hostache, M. Montanari, D. A. Plaza Guingla, V. Pauwels, G. De Lannoy, R. De Keyser, L. Pfister, L. Hoffmann, et al. (2011), Assimilating sar-derived water level data into a hydraulic model: a case study, *Hydrology and Earth System Sciences*, *15*(7), 2349–2365.
- Gleason, C. J., et M. T. Durand (2020), Remote sensing of river discharge: A review and a framing for the discipline, *Remote Sensing*, *12*(7), 1107.
- Gleason, C. J., et L. C. Smith (2014), Toward global mapping of river discharge using satellite images and at-many-stations hydraulic geometry, *Proceedings of the National Academy of Sciences*, *111*(13), 4788–4791.
- Gleason, C. J., et J. Wang (2015), Theoretical basis for at-many-stations hydraulic geometry, *Geophysical Research Letters*, *42*(17), 7107–7114.
- Gleason, C. J., L. C. Smith, et J. Lee (2014), Retrieval of river discharge solely from satellite imagery and at-many-stations hydraulic geometry: Sensitivity to river form and optimization parameters, *Water Resources Research*, *50*(12), 9604–9619.
- Gouldby, B., P. Sayers, J. Mulet-Marti, M. Hassan, et D. Benwell (2008), A methodology for regional-scale flood risk assessment, in *Proceedings of the Institution of Civil Engineers-Water Management*, vol. 161, pp. 169–182, Thomas Telford Ltd.
- Gupta, H. V., H. Kling, K. K. Yilmaz, et G. F. Martinez (2009), Decomposition of the mean squared error and NSE performance criteria: Implications for improving hydrological modelling, *Journal of Hydrology*, *377*(1), 80–91, doi:<https://doi.org/10.1016/j.jhydrol.2009.08.003>.

- Hadamard, J. (1902), Sur les problèmes aux dérivées partielles et leur signification physique, *Princeton university bulletin*, pp. 49–52.
- Haeni, F., M. L. Buursink, J. E. Costa, N. B. Melcher, R. T. Cheng, et W. J. Plant (2000), Ground penetrating radar methods used in surface-water discharge measurements, in *Eighth International Conference on Ground Penetrating Radar*, vol. 4084, pp. 494–500, SPIE.
- Häfliger, V., E. Martin, A. Boone, F. Habets, C. H. David, P.-A. Garambois, H. Roux, S. Ricci, L. Berthon, A. Thévenin, et al. (2015), Evaluation of regional-scale river depth simulations using various routing schemes within a hydrometeorological modeling framework for the preparation of the swot mission, *Journal of Hydrometeorology*, 16(4), 1821–1842.
- Hagemann, M., C. Gleason, et M. Durand (2017), Bam: Bayesian amhg-manning inference of discharge using remotely sensed stream width, slope, and height, *Water Resources Research*, 53(11), 9692–9707.
- Hallegatte, S., C. Green, R. J. Nicholls, et J. Corfee-Morlot (2013), Future flood losses in major coastal cities, *Nature climate change*, 3(9), 802–806.
- Hannah, D. M., S. Demuth, H. A. van Lanen, U. Looser, C. Prudhomme, G. Rees, K. Stahl, L. M. Tallaksen, et al. (2011), Large-scale river flow archives: importance, current status and future needs., *Hydrological Processes*, 25(7), 1191–1200.
- Hardesty, S., X. Shen, E. Nikolopoulos, et E. Anagnostou (2018), A numerical framework for evaluating flood inundation hazard under different dam operation scenarios—a case study in naugatuck river, *Water*, 10(12), 1798.
- Hauet, A. (2006), Estimation de débit et mesure de vitesse en rivière par large-scale particle image velocimetry, discharge estimates and velocity measurements using large-scale particle image velocimetry, *INP Grenoble, France (in French)*.
- Hauet, A. (2009), Estimation de débit et mesure de vitesse en rivière par large-scale particle image velocimetry, *La Houille Blanche*, 95(1), 80–85.
- Hauet, A., A. Kruger, W. F. Krajewski, A. Bradley, M. Muste, J.-D. Creutin, et M. Wilson (2008), Experimental system for real-time discharge estimation using an image-based method, *Journal of Hydrologic Engineering*, 13(2), 105–110.
- Hey, R. D., et C. R. Thorne (1986), Stable channels with mobile gravel beds, *Journal of Hydraulic engineering*, 112(8), 671–689.
- Hilton, J. E., S. Grimaldi, R. Cohen, N. Garg, Y. Li, S. Marvanek, V. Pauwels, et J. Walker (2019), River reconstruction using a conformal mapping method, *Environmental Modelling & Software*, 119, 197–213, doi:10.1016/j.envsoft.2019.06.006.
- Hoch, J. M., R. van Beek, H. C. Winsemius, et M. F. Bierkens (2018), Benchmarking flexible meshes and regular grids for large-scale fluvial inundation modelling, *Advances in Water Resources*, 121, 350–360.
- IGN, I. G. N. (2018), bd ortho version 2.0/ortho hrr version 1.0: descriptif de contenu, may 2013, updated july 2018, data retrieved from website of IGN, https://geoservices.ign.fr/ressources_documentaires/Espace_documentaire/ORTHO_IMAGES/BDORTHO_ORTHOHR/DC_BDORTHO_2-0_ORTHOHR_1-0.pdf.
- Jantzi, H., J.-M. Carozza, J.-L. Probst, et al. (2020), Les formes d'érosion en lit mineur rocheux: typologie, distribution spatiale et implications sur la dynamique du lit. exemple à partir des seuils rocheux molassiques de la moyenne garonne toulousaine (sud-ouest, france), *Géomorphologie: relief, processus, environnement*, 26(2).

- Kim, B., B. F. Sanders, J. E. Schubert, et J. S. Famiglietti (2014), Mesh type tradeoffs in 2d hydrodynamic modeling of flooding with a godunov-based flow solver, *Advances in Water Resources*, 68, 42–61.
- Kim, Y., M. Muste, A. Hauet, W. F. Krajewski, A. Kruger, et A. Bradley (2008), Stream discharge using mobile large-scale particle image velocimetry: A proof of concept, *Water Resources Research*, 44(9).
- Kinoshita, R. (1961), Investigation of channel deformation in ishikari river, *Report of Bureau of Resources*, 174.
- Kirstetter, G., F. Bourgin, P. Brigode, et O. Delestre (2020), Real-time inundation mapping with a 2d hydraulic modelling tool based on adaptive grid refinement: The case of the october 2015 french riviera flood, in *Advances in Hydroinformatics*, pp. 335–346, Springer.
- Kirstetter, G., O. Delestre, P.-Y. Lagrée, S. Popinet, et C. Josserand (2021), B-flood 1.0: an open-source saint-venant model for flash-flood simulation using adaptive refinement, *Geoscientific Model Development*, 14(11), 7117–7132.
- Knighton, A. (1981), Asymmetry of river channel cross-sections: Part i. quantitative indices, *Earth Surface Processes and Landforms*, 6(6), 581–588.
- Knighton, D. (2014), *Fluvial forms and processes: a new perspective*, Routledge.
- Kouraev, A. V., E. A. Zakharova, O. Samain, N. M. Mognard, et A. Cazenave (2004), Ob’river discharge from topex/poseidon satellite altimetry (1992–2002), *Remote sensing of environment*, 93(1-2), 238–245.
- Kousky, C. (2010), Learning from extreme events: Risk perceptions after the flood, *Land Economics*, 86(3), 395–422.
- Kurganov, A., et D. Levy (2002), Central-upwind schemes for the saint-venant system, *ESAIM: Mathematical Modelling and Numerical Analysis*, 36(3), 397–425.
- Lane, E. W. (1955), *Proceedings of the American Society of Civil Engineers*, 81(7), 1–17.
- Larnier, K., J. Monnier, P.-A. Garambois, et J. Verley (2021), River discharge and bathymetry estimation from swot altimetry measurements, *Inverse Problems in Science and Engineering*, 29(6), 759–789.
- Latapie, A., B. Terrier, A. Paquier, D. Chambon, et H. Doumenc (2012), Comparaison des résultats de l’approche hydro-géomorphologique avec la modélisation hydraulique : 1. cas d’étude sur la Garonne en aval de Toulouse, *Rapp. tech.*, ANR-08-RISK-03-01 EXTRAFL0 Report IV.1.
- Le Coarer, Y. (2007), Hydraulic signatures for ecological modelling at different scales, *Aquatic Ecology*, 41, 451–459, doi:10.1007/s10452-005-9005-3.
- Le Coz, J., C. Chaléon, L. Bonnifait, R. Le Boursicaud, B. Renard, F. Branger, J. Diribarne, et M. Valente (2013), Analyse bayésienne des courbes de tarage et de leurs incertitudes: la méthode baratin, *La Houille Blanche*, (6), 31–41.
- Le Coz, J., B. Renard, L. Bonnifait, F. Branger, et R. Le Boursicaud (2014), Combining hydraulic knowledge and uncertain gaugings in the estimation of hydrometric rating curves: A bayesian approach, *Journal of Hydrology*, 509, 573–587.
- Le Moine, N., et M. Mahdade (2021), A preliminary assessment of a newly-defined multispectral hue space for retrieving river depth with optical imagery and in situ calibration data, *Remote Sensing*, 13(21), 4435.
- Legleiter, C. J., D. A. Roberts, et R. L. Lawrence (2009), Spectrally based remote sensing of river bathymetry, *Earth Surface Processes and Landforms*, 34(8), 1039–1059.

- Legleiter, C. J., P. J. Kinzel, et J. M. Nelson (2017), Remote measurement of river discharge using thermal particle image velocimetry (piv) and various sources of bathymetric information, *Journal of Hydrology*, 554, 490–506.
- Leopold, L. B., et W. B. Langbein (1966), River meanders, *Scientific American*, 214(6), 60–73.
- Leopold, L. B., et T. Maddock (1953), *The hydraulic geometry of stream channels and some physiographic implications*, vol. 252, US Government Printing Office.
- Lhomme, J., P. Sayers, B. Gouldby, P. Samuels, M. Wills, et J. Mulet-Marti (2008), Recent development and application of a rapid flood spreading method.
- Lin, P., M. Pan, H. E. Beck, Y. Yang, D. Yamazaki, R. Frasson, C. H. David, M. Durand, T. M. Pavelsky, G. H. Allen, et al. (2019), Global reconstruction of naturalized river flows at 2.94 million reaches, *Water resources research*, 55(8), 6499–6516.
- MacWilliams Jr, M. L., J. M. Wheaton, G. B. Pasternack, R. L. Street, et P. K. Kitanidis (2006), Flow convergence routing hypothesis for pool-riffle maintenance in alluvial rivers, *Water Resources Research*, 42(10).
- Mahdade, M., N. Le Moine, R. Moussa, O. Navratil, et P. Ribstein (2020), Automatic identification of alternating morphological units in river channels using wavelet analysis and ridge extraction, *Hydrology and Earth System Sciences*, 24(7), 3513–3537.
- Medina, C. E., J. Gomez-Enri, J. J. Alonso, et P. Villares (2008), Water level fluctuations derived from envisat radar altimeter (ra-2) and in-situ measurements in a subtropical waterbody: Lake izabal (guatemala), *Remote Sensing of Environment*, 112(9), 3604–3617.
- Melcher, N., J. Costa, F. Haeni, R. Cheng, E. Thurman, M. Buursink, K. Spicer, E. Hayes, W. Plant, W. Keller, et al. (2002), River discharge measurements by using helicopter-mounted radar, *Geophysical Research Letters*, 29(22), 41–1.
- Monteil, C., F. Zaoui, N. Le Moine, et F. Hendrickx (2020), Multi-objective calibration by combination of stochastic and gradient-like parameter generation rules – the caramel algorithm, *Hydrology and Earth System Sciences*, 24(6), 3189–3209, doi:10.5194/hess-24-3189-2020.
- Moussa, R., et C. Bocquillon (2000), Approximation zones of the Saint-Venant equations for flood routing with overbank flow, *Hydrology and Earth System Sciences Discussions*, 4(2), 251–260.
- Muste, M., K. Lee, D. Kim, C. Bacotiu, M. Rojas Oliveros, Z. Cheng, et F. Quintero (2020), Revisiting hysteresis of flow variables in monitoring unsteady streamflows, *Journal of Hydraulic Research*, 58(6), 867–887, doi:10.1080/00221686.2020.1786742.
- Navratil, O. (2005), Débit de pleins bords et géométrie hydraulique: une description synthétique de la morphologie des cours d'eau pour relier le bassin versant et les habitats aquatiques, Thèse de doctorat, Thèse de doctorat en mécanique des milieux géophysiques et Environnement.
- Neal, J., L. Hawker, J. Savage, M. Durand, P. Bates, et C. Sampson (2021), Estimating river channel bathymetry in large scale flood inundation models, *Water Resources Research*, 57(5), e2020WR028,301.
- Néelz, S., et G. Pender (2013), Benchmarking the latest generation of 2d hydraulic modelling packages, *Environment Agency: Bristol, UK*.
- Néelz, S., G. Pender, et al. (2010), Benchmarking of 2d hydraulic modelling packages.
- Nicollet, G., et M. Uan (1979), Ecoulements permanents à surface libre en lits composés, *La Houille Blanche*, (1), 21–30.

- Nobre, A. D., L. A. Cuartas, M. Hodnett, C. D. Rennó, G. Rodrigues, A. Silveira, et S. Saleska (2011), Height above the nearest drainage—a hydrologically relevant new terrain model, *Journal of Hydrology*, 404(1-2), 13–29.
- O’Connor, J. E., et J. E. Costa (2004), *The world’s largest floods, past and present: their causes and magnitudes*, 1254, US Geological Survey.
- Oki, T., et S. Kanae (2006), Global hydrological cycles and world water resources, *science*, 313(5790), 1068–1072.
- Oubanas, H., I. Gejadze, P.-O. Malaterre, M. Durand, R. Wei, R. d. M. Frasson, et A. Domeneghetti (2018a), Discharge estimation in ungauged basins through variational data assimilation: The potential of the swot mission, *Water Resources Research*, 54(3), 2405–2423.
- Oubanas, H., I. Gejadze, P.-O. Malaterre, et F. Mercier (2018b), River discharge estimation from synthetic swot-type observations using variational data assimilation and the full saint-venant hydraulic model, *Journal of Hydrology*, 559, 638–647.
- Oudin, L., F. Hervieu, C. Michel, C. Perrin, V. Andréassian, F. Anctil, et C. Loumagne (2005), Which potential evapotranspiration input for a lumped rainfall–runoff model?: Part 2 – Towards a simple and efficient potential evapotranspiration model for rainfall–runoff modelling, *Journal of Hydrology*, 303(1), 290–306, doi:<https://doi.org/10.1016/j.jhydrol.2004.08.026>.
- Oudin, L., A. Kay, V. Andréassian, et C. Perrin (2010), Are seemingly physically similar catchments truly hydrologically similar?, *Water Resources Research*, 46(11).
- Papa, F., B. Legrésy, et F. Rémy (2003), Use of the topex–poseidon dual-frequency radar altimeter over land surfaces, *Remote sensing of Environment*, 87(2-3), 136–147.
- Papa, F., C. Prigent, F. Durand, et W. B. Rossow (2006), Wetland dynamics using a suite of satellite observations: A case study of application and evaluation for the indian subcontinent, *Geophysical Research Letters*, 33(8).
- Papa, F., S. K. Bala, R. K. Pandey, F. Durand, V. Gopalakrishna, A. Rahman, et W. B. Rossow (2012), Ganga-brahmaputra river discharge from jason-2 radar altimetry: An update to the long-term satellite-derived estimates of continental freshwater forcing flux into the bay of bengal, *Journal of Geophysical Research: Oceans*, 117(C11).
- Park, C. C. (1977), World-wide variations in hydraulic geometry exponents of stream channels: an analysis and some observations, *Journal of Hydrology*, 33(1-2), 133–146.
- Parker, G., P. Diplas, et J. Akiyama (1983), Meander bends of high amplitude, *Journal of Hydraulic Engineering*, 109(10), 1323–1337.
- Pavelsky, T. M., M. T. Durand, K. M. Andreadis, R. E. Beighley, R. C. Paiva, G. H. Allen, et Z. F. Miller (2014), Assessing the potential global extent of swot river discharge observations, *Journal of Hydrology*, 519, 1516–1525.
- Plessix, R.-E. (2006), A review of the adjoint-state method for computing the gradient of a functional with geophysical applications, *Geophysical Journal International*, 167(2), 495–503, doi:10.1111/j.1365-246X.2006.02978.x.
- Popinet, S. (2003), Gerris: a tree-based adaptive solver for the incompressible euler equations in complex geometries, *Journal of computational physics*, 190(2), 572–600.
- Popinet, S. (2011), Quadtree-adaptive tsunami modelling, *Ocean Dynamics*, 61(9), 1261–1285.

- Popinet, S. (2015), A quadtree-adaptive multigrid solver for the serre–green–naghdi equations, *Journal of Computational Physics*, 302, 336–358.
- Rantz, S. E. (1982), *Measurement and computation of streamflow*, vol. 2175, US Department of the Interior, Geological Survey.
- Rebolho, C., V. Andréassian, et N. Le Moine (2018), Inundation mapping based on reach-scale effective geometry, 22(11), 5967–5985, doi:10.5194/hess-22-5967-2018.
- Ridenour, G. S., et J. R. Giardino (1994), Methodology: discriminant function analysis of compositional data: an example from hydraulic geometry, *Physical Geography*, 15(5), 481–492.
- Rodriguez, E. (2015), Surface water and ocean topography mission (swot), science requirements document. jpl document d-61923. jpl d-61923, feb. 12, 2015.
- Rouhier, L., M. Le Lay, F. Garavaglia, N. Le Moine, F. Hendrickx, C. Monteil, et P. Ribstein (2017), Impact of mesoscale spatial variability of climatic inputs and parameters on the hydrological response, *Journal of Hydrology*, 553, 13–25, doi:https://doi.org/10.1016/j.jhydrol.2017.07.037.
- Saint Venant, B. (1871), Theorie du mouvement non-permanent des eaux avec application aux crues des rivières et à l’introduction des marées dans leur lit, *Academic de Sci. Comptes Rendus*, 73(99), 148–154.
- Sampson, C. C., A. M. Smith, P. D. Bates, J. C. Neal, L. Alfieri, et J. E. Freer (2015), A high-resolution global flood hazard model, *Water resources research*, 51(9), 7358–7381.
- Sanyal, J., et X. X. Lu (2004), Application of remote sensing in flood management with special reference to monsoon asia: a review, *Natural Hazards*, 33(2), 283–301.
- Sauvage, S., S. Teissier, P. Vervier, T. Ameziane, F. Garabetian, F. Delmas, et B. Caussade (2003), A numerical tool to integrate biophysical diversity of a large regulated river: hydrobiogeochemical bases. the case of the garonne river (france), *River Research and Applications*, 19(2), 181–198.
- Schaperow, J. R., D. Li, S. A. Margulis, et D. P. Lettenmaier (2019), A curve-fitting method for estimating bathymetry from water surface height and width, *Water Resources Research*, 55(5), 4288–4303.
- Shen, C., S. Wang, et X. Liu (2016), Geomorphological significance of at-many-stations hydraulic geometry, *Geophysical Research Letters*, 43(8), 3762–3770.
- Shen, X., E. N. Anagnostou, G. H. Allen, G. R. Brakenridge, et A. J. Kettner (2019a), Near-real-time non-obstructed flood inundation mapping using synthetic aperture radar, *Remote Sensing of Environment*, 221, 302–315.
- Shen, X., D. Wang, K. Mao, E. Anagnostou, et Y. Hong (2019b), Inundation extent mapping by synthetic aperture radar: A review, *Remote Sensing*, 11(7), 879.
- Sichangi, A. W., L. Wang, K. Yang, D. Chen, Z. Wang, X. Li, J. Zhou, W. Liu, et D. Kuria (2016), Estimating continental river basin discharges using multiple remote sensing data sets, *Remote Sensing of Environment*, 179, 36–53.
- Simpson, M. R. (2001), *Discharge measurements using a broad-band acoustic Doppler current profiler*, US Department of the Interior, US Geological Survey Reston, CA, USA.
- Smith, L. C., et T. M. Pavelsky (2008), Estimation of river discharge, propagation speed, and hydraulic geometry from space: Lena river, siberia, *Water Resources Research*, 44(3).
- Steiger, J., et D. Corenblit (2000), Dynamique fluviale de la garonne toulousaine, *Rapport Agence de l’eau Adour-Garonne*.

- Strack, O. D. (2017), *Analytical groundwater mechanics*, Cambridge University Press.
- Tabary, P., P. Dupuy, G. L'Henaff, C. Gueguen, L. Moulin, O. Laurantin, C. Merlier, et J.-M. Soubeyroux (2012), A 10-year (1997–2006) reanalysis of Quantitative Precipitation Estimation over France: methodology and first results, *IAHS-AISH Publication*, pp. 255–260.
- Teng, J., A. J. Jakeman, J. Vaze, B. F. Croke, D. Dutta, et S. Kim (2017), Flood inundation modelling: A review of methods, recent advances and uncertainty analysis, *Environmental modelling & software*, *90*, 201–216.
- Tuozzolo, S., G. Lind, B. Overstreet, J. Mangano, M. Fonstad, M. Hagemann, R. Frasson, K. Larnier, P.-A. Garambois, J. Monnier, et al. (2019), Estimating river discharge with swath altimetry: A proof of concept using airswot observations, *Geophysical Research Letters*, *46*(3), 1459–1466.
- Van Hooft, J. A., S. Popinet, C. C. Van Heerwaarden, S. J. Van der Linden, S. R. de Roode, et B. J. Van de Wiel (2018), Towards adaptive grids for atmospheric boundary-layer simulations, *Boundary-layer meteorology*, *167*(3), 421–443.
- Vandembroucq, D., et S. Roux (1997), Conformal Mapping on Rough Boundaries I: Applications to harmonic problems, *Phys. Rev. E*, *55*, 6171–6185, doi:10.1103/PhysRevE.55.6171.
- Vidal, J.-P., E. Martin, L. Franchistéguy, M. Baillon, et J.-M. Soubeyroux (2010), A 50-year high-resolution atmospheric reanalysis over France with the SAFRAN system, *International Journal of Climatology*, *30*(11), 1627–1644, doi:https://doi.org/10.1002/joc.2003.
- Wallingford, H. (2006), Rapid flood spreading methodology (rfsm), *Environment Agency, Thames Estuary*, *2100*.
- Yoon, Y., M. Durand, C. J. Merry, E. A. Clark, K. M. Andreadis, et D. E. Alsdorf (2012), Estimating river bathymetry from data assimilation of synthetic swot measurements, *Journal of Hydrology*, *464*, 363–375.
- Yoon, Y., P.-A. Garambois, R. C. Paiva, M. Durand, H. Roux, et E. Beighley (2016), Improved error estimates of a discharge algorithm for remotely sensed river measurements: Test cases on sacramento and g aronne r ivers, *Water Resources Research*, *52*(1), 278–294.
- Young, A. R. (2006), Stream flow simulation within uk ungauged catchments using a daily rainfall-runoff model, *Journal of Hydrology*, *320*(1-2), 155–172.
- Zheng, X., D. G. Tarboton, D. R. Maidment, Y. Y. Liu, et P. Passalacqua (2018), River Channel Geometry and Rating Curve Estimation Using Height above the Nearest Drainage, *Journal of the American Water Resources Association*, *54*(4), 785–806, doi:10.1111/1752-1688.12661.

Annexe A

Repères de crue de Juin 2000 de la Garonne

2D high resolution numerical simulation of historical flood events on the Garonne River, based on adaptive mesh refinement and a specific handling of tributary inlets for parallelization

Mounir Mahdade¹, Nicolas Le Moine¹, and Pierre Ribstein¹

¹*Sorbonne Université, CNRS, EPHE, Milieux environnementaux, transferts et interaction dans les hydrosystèmes et les sols, METIS, F-75005 Paris, France*

Supplementary: Flood marks of the June 2000 flood event

The national collaborative platform of flood sites and markers¹ provides marker sheets with a description of the site, its location accompanied by a few photos, as well as the estimated water level at this location. However, the topographic level is not provided although it is very important for the model calibration.

These sheets give the positions of the markers in a less exact way with inconsistencies between the photos taken, the description and the position written in the sheet. For this, we have corrected these locations by:

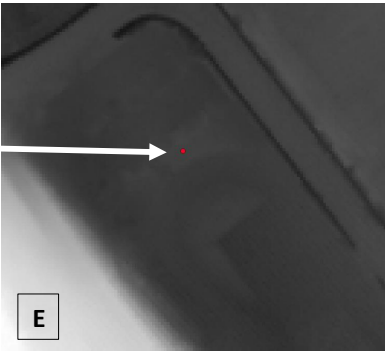
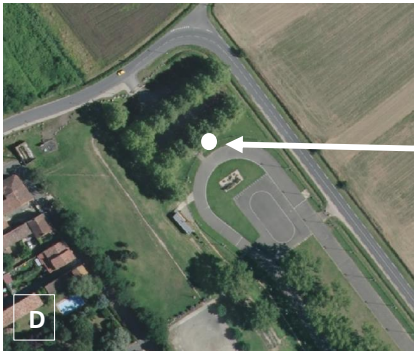
- Verifying that the position (x,y) is indeed the one in the photo by using Orthophotos from IGN with a resolution of 0.2 m.
- Using Google Maps (3D view with Google Street Map View), we compare the location with the photo taken by the agents during the flood (we can change the date of the photos on Google Maps for more precision on the location).

If there is a coherence between the position and the photos, we extract the topographic level in NGF meters using the 1m resolution DEM. Otherwise, we look for the right position using Google Maps and the description provided by the file.

¹ www.reperesdecrues.developpement-durable.gouv.fr

ID	Markers code	Location	City	X (old) in WGS84	X (new) in WGS84	Y (old) in WGS84	Y (new) in WGS84	Z in mNGF (old)	Z in mNGF (new)	Offset (m)	Comments
1	GTL_R_4807 :	Rest area on a truck detour	GRENADE	1.2949808	1.2949808	43.7752059	43.7752059	105.51	105.51	1.32	Location is exact
2	GTL_R_4805 :	Chemin du Gà / Chemin des Monges	ONDES	1.3107900	1.3108768	43.7798999	43.7797117	106.18	105.98	1.4	Location is modified
3	GTL_R_4804 :	access ramp to the old bridge	CASTELNAU-D'ESTRETEF ONDS	1.3324300	1.3316509	43.7768000	43.7756331	110.01	108.98	0 (traces)	Location is modified
4	GTL_R_4802 :	1 km downstream from Gagnac	GAGNAC-SUR-GARONNE	1.3681200	1.3675042	43.7028000	43.7035736	118.2	117.94	0 (traces)	Location is modified
5	WEB_R_2018 01164736 :	1-5 km downstream from Gagnac	GAGNAC-SUR-GARONNE	1.3649701	1.3649320	43.7107000	43.7101942	118.12	117.8	0 (traces)	Location is modified
6	GTL_R_4798 :	3 km downstream of the twin bridges	TOULOUSE	1.4063200	1.4066908	43.6337000	43.6341388	128.6	128.58	0.2	Location is modified and the offset is estimated from photos
7	GTL_R_4799 :	"Amicale du Chien d'utilité"	BLAGNAC	1.4004400	1.4014359	43.6401000	43.6387163	126.54	126.75	1.5	Location is modified and the offset is estimated from photos
8	GTL_R_4800 :	Building next to a playground	FENOUILLET	1.3915700	1.3933053	43.6776000	43.6757865	120.98	121.69	0.5	Location is modified
9	GTL_R_4801 :	Water treatment plant	FENOUILLET	1.3708201	1.3707427	43.6827000	43.6825321	119.94	120.22	0.42	Location is modified

GTL_R_4807 :



The location is exact. The link of the location on Google Maps:

https://www.google.fr/maps/@43.7752121,1.2955495,3a,75y,255.2h,86.11t/data=!3m6!1e1!3m4!1s2Zu mfP_bLqeiHIM8Wp86xw!2e0!7i13312!8i6656

Comment provided by the survey:

Photograph taken on 2000-06-20. The flood has left a clear trace on the building (dashes) (C)- at 1.32m from the ground.

GTL_R_4805 :

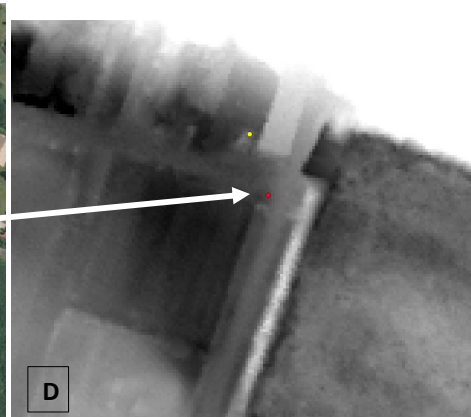


Image B shows the position of the flood marker height located on the electrical post that no longer exists on image A from Google Maps. This image shows the position of the marker. It helped us in addition to the Orthophoto of image C to locate the exact point of the marker (white point). The point with the yellow color is the old position of the marker. In D, a representation of the topography is taken from the DEM with a scale from black to white: min=105 m and max = 108 m. The location is exact. The link of the location on Google Maps:

<https://www.google.fr/maps/@43.779757,1.3109112,3a,75y,184.83h,82.4t/data=!3m7!1e1!3m5!1sEwhG-wwWp9OTCJlRgqpr5w!2e0!5s20140501T000000!7i13312!8i6656>

Comment provided by the survey:

Photograph taken on 2000-06-18. View towards the agglomeration. The level of the flood was marked on an electric post (arrow, B). This marker is at 1 – 1.40m from the ground.

GTL_R_4804 :



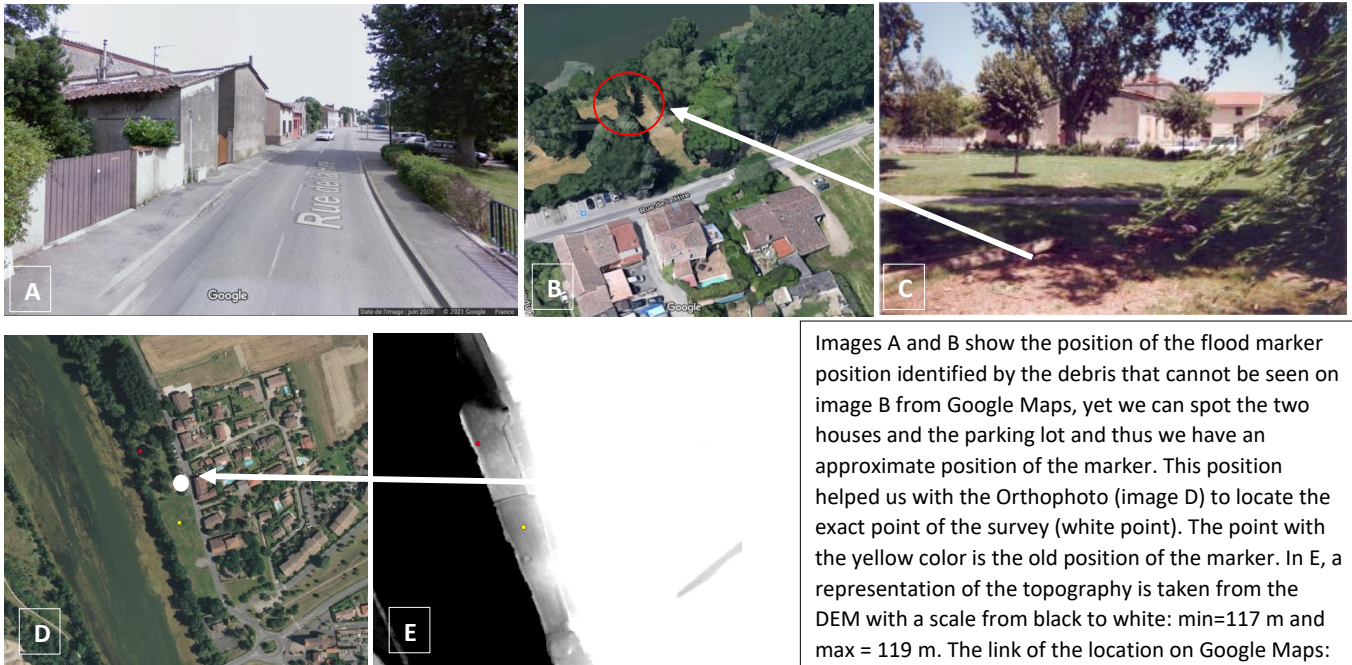
Image C shows the position of the flood marker position located on the trail. Images A and B from Google Maps show the position of the electrical post and the house that helped us, in addition to the comment provided by the survey, to detect the real position of the marker. So, we located exactly from the Orthophoto of the image D the exact point of the survey (white point). The point with the yellow color is the old position of the marker. In E, a representation of the topography is taken from the DEM with a scale from black to white: min=107 m and max = 111 m. The link of the location on Google Maps:

<https://www.google.fr/maps/@43.7757731,1.3316622,3a,60y,183.63h,80.29t/data=!3m6!1e1!3m4!1scA6ZVuiw3FCcjH91-Q5Cw!2e0!7i13312!8i6656>

Comment provided by the survey:

Photograph taken on 2000-06-18. View towards the NEE on Galanti. In the foreground - the flood has left a clear trace of mud (arrow) on the access ramp to the old bridge.

GTL_R_4802 :



Images A and B show the position of the flood marker position identified by the debris that cannot be seen on image B from Google Maps, yet we can spot the two houses and the parking lot and thus we have an approximate position of the marker. This position helped us with the Orthophoto (image D) to locate the exact point of the survey (white point). The point with the yellow color is the old position of the marker. In E, a representation of the topography is taken from the DEM with a scale from black to white: min=117 m and max = 119 m. The link of the location on Google Maps:

<https://www.google.fr/maps/place/43%C2%B042'10.1%22N+1%C2%B022'05.2%22E/@43.7037302,1.3678047,111m/data=!3m1!1e3!4m5!3m4!1s0x0:0x0!8m2!3d43.7028!4d1.36812>

Comment provided by the survey:

Photograph taken on 2000-06-18. Opposite view of the Garonne towards La Hire. In the foreground - you can see debris deposited by the flood. The houses were not flooded.

WEB_R_201801164736 :



Comment provided by the survey:

Photograph taken on 2000-06-18. In the foreground we can see a clear flood line. The flood did not reach the house which carries a marker of the flood of 1875-06-23 (arrows) - at 2.00m from the ground.

Image B shows the position of flood marker position located next to the building shown also in image A from Orthophoto. It shows the exact position of the marker. This position in the image B with the comment provided by the survey helped us to locate this position (red point). The point with the yellow color is the old position of the marker. The link of the location on Google Maps:

<https://www.google.fr/maps/@43.7102017,1.3650153,3a,87.1y,335.48h,74.82t/data=!3m6!1e1!3m4!1sDD0hY8cgjY8kaUm b2ri-kQ!2e0!7i13312!8i6656>

GTL_R_4798 :



Image B shows the position of the flood marker position located on the blue portal that no longer exists on Google Maps (D). The gate was located from the Orthophoto in image A and image D by positioning the exact point of the survey (white point). The point with the yellow color is the old position of the marker. In D, a representation of the topography is taken from the DEM with a scale from black to white: min=127 m and max = 130 m. The link of the location on Google Maps:

<https://www.google.fr/maps/@43.633682,1.4066733,3a,75y,349.06h,100.97t/data=!3m6!1e1!3m4!1sVFJLitAinKYu1k9PN0680Q!2e0!7i13312!8i6656>

Comment provided by the survey:

Photograph taken on 2000-06-18. View on the entrance gate and towards the Garonne. The flood has left a clear trace on the gate (arrows).

GTL_R_4799 :

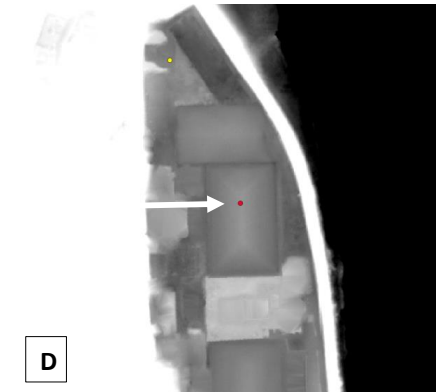


Image B shows the position of the flood marker position located on a grass that is not well defined. We were able to locate the entrance (image A) on Google Maps. This position helped us from the Orthophoto of image C to position the approximate point of the survey (white point). The point with the yellow color being the old position of the marker which is far from what we see on the image of the sheet. In D, a representation of the topography is taken from the DEM with a scale from black to white: min=125 m and max = 128 m. The link of the location on Google Maps:

<https://www.google.fr/maps/place/43%C2%B038'24.4%22N+1%C2%B024'01.6%22E/@43.6386892,1.4006949,162m/data=!3m1!1e3!4m5!3m4!1s0x0:0x0!8m2!3d43.640114d1.40044>

Comment provided by the survey:

Photograph taken on 2000-06-18. View downstream on the Amicale's land. In the center - we can see a trace of mud in the hedge (dashes).

GTL_R_4800 :



Image A shows the position of the flood marker position located on a building that cannot be located easily on Google Maps. The position of the same building is found in the image C and B (white point). The link of the location on Google Maps:

<https://www.google.fr/maps/place/43%C2%B040'32.9%22N+1%C2%B023'36.0%22E/@43.6753527,1.3931272,58a,35y,39.49t/data=!3m1!1e3!4m6!3m5!1s0x0:0x43336e1ab6abfd90e!7e2!8m2!3d43.6757914!4d1.3933273>

Comment provided by the survey:

Photograph taken on 2000-06-18. The flood has left a clear trace on the building at 0 - 50cm from the ground (dotted line).

GTL_R_4801 :



Image B shows the position of the flood marker located at the entrance of the treatment plant, which is located on image A from Google Maps. This position helped us in addition to the Orthophoto (image C) to locate the exact point of the marker (white point). The point with the yellow color is the old position of the marker. In D, a representation of the topography is taken from the DEM with a scale from black to white: min=119 m and max = 121 m. The link of the location on Google Maps:

<https://www.google.fr/maps/@43.682452,1.3706536,3a,75y,53.4h,84.76t/data=!3m6!1e1!3m4!1sR-3tidiQiXJ7m6hhYwYe4Q!2e0!7i13312!8i6656>

Comment provided by the survey:

Photograph taken on 2000-06-18. The flood level was marked on the entrance gate at 0 – 0.42m from the ground (arrow).

Annexe B

Implémentation de l'approximation de l'onde diffusante 2D sous Basilisk

B.1 Dérivation de l'approximation de l'onde diffusante

Pour établir l'équation de l'onde diffusante, nous repartons du système de Saint-Venant en 2D (une équation de continuité + 2 projections de l'équation dynamique) :

$$\frac{\partial}{\partial t} \begin{bmatrix} h \\ h u_x \\ h u_y \end{bmatrix} = -\nabla \cdot \begin{bmatrix} h u_x & h u_y \\ h u_x^2 + \frac{1}{2}gh^2 & h u_x u_y \\ h u_x u_y & h u_y^2 + \frac{1}{2}gh^2 \end{bmatrix} - gh \begin{bmatrix} 0 \\ \partial_x z_b \\ \partial_y z_b \end{bmatrix} - \frac{1}{\rho} \begin{bmatrix} 0 \\ \tau_x \\ \tau_y \end{bmatrix}$$

L'approximation de l'onde diffusante consiste à négliger les termes d'accélération inertielle dans le membre de gauche (projections de l'équation dynamique), ainsi que les termes d'accélération convective dans le membre de droite :

$$\begin{bmatrix} \frac{\partial h}{\partial t} \\ \frac{\partial(h u_x)}{\partial t} \\ \frac{\partial(h u_y)}{\partial t} \end{bmatrix} = -\nabla \cdot \begin{bmatrix} h u_x & h u_y \\ \frac{1}{2}gh^2 & 0 \\ 0 & \frac{1}{2}gh^2 \end{bmatrix} - \nabla \cdot \begin{bmatrix} 0 & 0 \\ h u_x^2 & h u_x u_y \\ h u_x u_y & h u_y^2 \end{bmatrix} - gh \begin{bmatrix} 0 \\ \partial_x z_b \\ \partial_y z_b \end{bmatrix} - \frac{1}{\rho} \begin{bmatrix} 0 \\ \tau_x \\ \tau_y \end{bmatrix}$$

En notant respectivement $q_x = h u_x$ et $q_y = h u_y$ les composantes du vecteur densité de flux, le système se résume alors à :

$$\begin{cases} \frac{\partial h}{\partial t} = -\nabla \cdot \mathbf{q} \\ 0 = -gh \frac{\partial h}{\partial x} - gh \frac{\partial z_b}{\partial x} - \frac{1}{\rho} \tau_x \\ 0 = -gh \frac{\partial h}{\partial y} - gh \frac{\partial z_b}{\partial y} - \frac{1}{\rho} \tau_y \end{cases}$$

L'équation dynamique (deux dernières équations scalaires) se résume donc à un quasi-équilibre entre frottement et forces résultant du gradient de pression : en effet, en notant $\eta = z_b + h$ la cote de la surface libre, on a :

$$\begin{cases} \frac{\partial h}{\partial t} = -\nabla \cdot \mathbf{q} \\ 0 = -gh \frac{\partial \eta}{\partial x} - \frac{1}{\rho} \tau_x \\ 0 = -gh \frac{\partial \eta}{\partial y} - \frac{1}{\rho} \tau_y \end{cases}$$

En prenant comme modèle de frottement le modèle de Manning, nous avons $\boldsymbol{\tau} = \rho g n^2 h^{-\frac{7}{3}} \|\mathbf{q}\| \mathbf{q}$ et donc :

$$\begin{cases} q_x = -\frac{1}{n^2 \|\mathbf{q}\|} h^{\frac{10}{3}} \frac{\partial \eta}{\partial x} \\ q_y = -\frac{1}{n^2 \|\mathbf{q}\|} h^{\frac{10}{3}} \frac{\partial \eta}{\partial y} \end{cases}$$

D'où

$$\|\mathbf{q}\| = \frac{1}{n^2 \|\mathbf{q}\|} h^{\frac{10}{3}} \|\nabla \eta\|$$

Nous obtenons donc une expression explicite pour la norme du vecteur densité de flux, puis pour le vecteur lui-même :

$$\begin{aligned} \|\mathbf{q}\| &= \frac{1}{n} h^{\frac{5}{3}} \|\nabla \eta\|^{\frac{1}{2}} \\ \mathbf{q} &= -\frac{1}{n} (\eta - z_b)^{\frac{5}{3}} \|\nabla \eta\|^{-\frac{1}{2}} \nabla \eta \end{aligned}$$

Il ne reste donc finalement que l'équation de continuité, qui s'écrit sous la forme d'une équation de diffusion :

$$\frac{\partial \eta}{\partial t} = +\nabla \cdot \left(\underbrace{\frac{1}{n} (\eta - z_b)^{\frac{5}{3}} \|\nabla \eta\|^{-\frac{1}{2}} \nabla \eta}_{D(\eta, \|\nabla \eta\|)} \right)$$

Le problème est donc très non-linéaire, la diffusivité D dépendant à la fois de η et de la norme du gradient de η . Cependant nous pouvons facilement l'implémenter sous Basilisk grâce à solveur de Poisson (<http://basilisk.fr/src/diffusion.h>), qui utilise un schéma *staggered grid* implicite (Figure B.1).

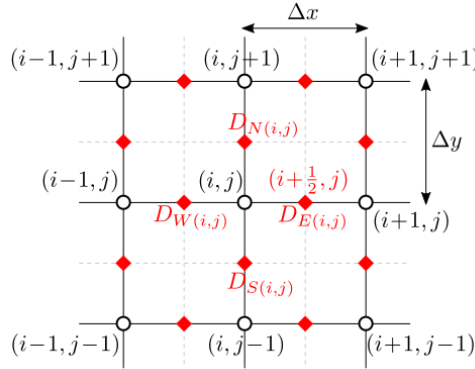


Figure B.1:

Il est nécessaire de désingulariser la norme du gradient de η , qui intervient à la puissance $-\frac{1}{2}$ dans la diffusivité. Nous utilisons :

$$D(\eta, \|\nabla \eta\|) = \frac{1}{n} (\eta - z_b)^{\frac{5}{3}} \left(\|\nabla \eta\| + \epsilon \right)^{-\frac{1}{2}}$$

B.2 Cas périodique en topographie sans tendance

Comme pour le système de Saint-Venant complet, la résolution en topographie périodique dépouillée de sa tendance nécessite un petite réécriture du problème. En notant $\mathbf{I}_{\text{reg}} = (I_x, I_y)^T$ la tendance régionale qui est

le moteur de l'écoulement, nous avons :

$$\begin{cases} \nabla \eta = \nabla \eta' - \mathbf{I}_{\text{reg}} \\ h = (\eta - z_b) = (\eta' - z'_b) \end{cases}$$

Nous obtenons :

$$\frac{\partial \eta'}{\partial t} = +\nabla \cdot \left(\underbrace{\frac{1}{n} (\eta' - z'_b)^{\frac{5}{3}} \|\nabla \eta' - \mathbf{I}_{\text{reg}}\|^{-\frac{1}{2}} (\nabla \eta' - \mathbf{I}_{\text{reg}})}_{D(\eta', \|\nabla \eta'\|)} \right) = \nabla \cdot (D \nabla \eta') - \mathbf{I}_{\text{reg}} \cdot \nabla D$$

Nous voyons donc qu'il n'y a pas d'obstacle à résoudre le problème avec le solveur de Poisson en variable η' sur la topographie z'_b (cote sans tendance de la surface libre et topographie sans tendance, respectant toutes deux la condition aux limites périodique), sous réserve d'ajouter un terme source fictif à l'équation, donné par $-\mathbf{I}_{\text{reg}} \cdot \nabla D$. La désingularisation de la norme du gradient reste bien sûr nécessaire :

$$D(\eta', \|\nabla \eta'\|) = \frac{1}{n} (\eta' - z'_b)^{\frac{5}{3}} \left(\|\nabla \eta' - \mathbf{I}_{\text{reg}}\| + \epsilon \right)^{-\frac{1}{2}}$$

avec $\epsilon = 0.001 |\mathbf{I}_{\text{reg}}|$.

B.3 Conservation de la masse

Une des spécificités de la formulation onde diffusante est que, là où la surface est sèche, on spécifie malgré tout une cote de la surface libre égale à la cote topographique : $\eta = z_b$. Il se peut donc que sur un pas de temps, on passe sur une cellule donnée de $\eta = z_b$ à $\eta < z_b$ (ou, de façon synonyme, de $\eta' = z'_b$ à $\eta' < z'_b$), avec typiquement un flux non-nul entre une cellule mouillée et une cellule sèche si la diffusivité à l'interface n'est pas nulle. A la fin de l'itération, on rétablit $\eta \leftarrow \max\{\eta, z_b\}$ mais cela revient à créer artificiellement un volume (par unité de surface) $\Delta \eta^{(i)} = (z_b - \eta^{(i)})$ sur la cellule en question. Pour assurer la conservation du volume, on stocke en mémoire la quantité $\Delta \eta^{(i)}$ (dans le champ scalaire `excess[]`), que l'on compensera dans l'équation de diffusion au pas de temps suivant sous la forme d'un terme puits $-\frac{\Delta \eta^{(i)}}{\Delta t}$. Nous pouvons même tirer parti du caractère implicite du solveur de Poisson pour améliorer l'estimation du terme puits nécessaire ; en effet, si un rétablissement de $\eta = z_b$ a été nécessaire à un pas de temps donné sur une cellule donnée, il y a fort à parier qu'à l'itération suivante, ce rétablissement sera à nouveau nécessaire. Le terme puits compensatoire associé aurait pour valeur $-\left(\frac{z_b - \eta^{(i+1)}}{\Delta t}\right)$. Nous pouvons donc construire une estimation semi-implicite du terme puits compensatoire 'au milieu du pas de temps', à l'aide d'un paramètre α et du booléen $\mathcal{K}_{(\Delta \eta^{(i)} > 0)}$ (qui vaut 1 si $\Delta \eta^{(i)} > 0$ sur la cellule à l'itération précédente, et 0 sinon) :

$$r = - \left[(1 - \alpha) \underbrace{\frac{\Delta \eta^{(i)}}{\Delta t}}_{\text{explicite}} + \alpha \underbrace{\mathcal{K}_{(\Delta \eta^{(i)} > 0)} \left(\frac{z_b - \eta^{(i+1)}}{\Delta t} \right)}_{\text{implicite}} \right]$$

$$r = - \left[(1 - \alpha) \frac{\Delta \eta^{(i)}}{\Delta t} + \alpha \mathcal{K}_{(\Delta \eta^{(i)} > 0)} \frac{z_b}{\Delta t} \right] + \left[\alpha \mathcal{K}_{(\Delta \eta^{(i)} > 0)} \frac{1}{\Delta t} \right] \eta^{(i+1)}$$

Nous obtenons donc une terme réactif sous forme affine implicite, directement géré par le solveur de Poisson sous Basilisk :

$$r = r_0 + \beta \eta^{(i+1)}$$

Cette correction est stable à la condition $\alpha < 1$, une étude plus poussée de ce schéma numérique est nécessaire pour démontrer cette propriété. La condition $\eta < z_b$ peut être produite par la fonction `diffusion()`, mais également par l'adaptivité (il se peut qu'après l'opération d'adaptation de la grille, la surface `eta[]` soit localement inférieure à la surface `zb[]`) : dans ce dernier cas de figure, si la surface est localement sèche, il faut ignorer cette condition et ne pas mémoriser d'excédent à compenser.

B.4 Implémentation Basilisk C

```

1 #include "grid/quadtrees.h"
2 #include "run.h"
3 #include "diffusion.h"
4 #include "kinoshita.h"
5 #include "terrain.h"
6 #include "live_terrain.h"
7 #include "esri-binary-grids.h"
8
9 #pragma autolink -lfftw3
10
11 #define LEVEL 7
12 #define MINLEVEL 5
13 #define ETAE 1.0e-3 // error on free surface elevation (1 mm)
14 #define HE 1.0e-3 // error on water depth (1 mm)
15 #define SE 1.0e-6 // error on slope (m/m)
16 #define ZE 1.0e-2 // error on bed elevation (1 cm)
17 #define UE 1.0e-2 // 0.01 m/s
18 #define tmax 3600.
19 #define sec_per_day 86400.
20
21 double G = 9.81, dry = 2.*HE;
22 double hrun = 3., hpool = 6., Kmin = 25., Kmaj = 17.;
23 double etaini = -1.0, prec = 0.01;
24 scalar zb[], eta[], h[], nmanning[], l[], r[], beta[], excess[];
25 face vector D[];
26
27 double dt;
28 mgstats mgd;
29
30 coord tilt;
31
32 // Garonne:
33 struct Kinoshita K = { .kcurv = 2.*M_PI/3400.,
34                       .theta0 = 1.15,
35                       .alphas = 1./10.,
36                       .alphaf = 1./96.,
37                       .sinuosity = -1. } ;
38
39 double ChannelWidth(double s, struct Kinoshita * _K)
40 {
41     return(200.);
42 }
43
44 double TalwegDepth(double s, struct Kinoshita * _K, double kmax)
45 {
46     double phase = 2.*_K->kcurv*s ; // factor 2 for pi-periodicity
47     double res = 0.5*(hpool-hrun)*cos(phase) + 0.5*(hpool+hrun);
48     return(res);
49 }
50
51 double TopoFloodplain(double phi)
52 {
53     return 0.012*140.*sqrt(M_PI)*erf(fabs(phi)/140.)/2. + 0.004*fabs(phi);
54 }

```

```

55
56 /**
57 ## Diffusivity function */
58
59 int update_diffusivity(scalar zb, scalar eta, face vector D)
60 {
61     double eps_slope = sqrt(sq(tilt.x)+sq(tilt.y))/1000.;
62
63     //    /\ reminder : D.x and D.y are not co-located
64     foreach_face()
65     {
66         double hm = (eta[+eta[-1,0]-zb[]-zb[-1,0])/2.; // depth interpolated at face center
67         hm = max(hm,0.);
68         double sn = (eta[]-eta[-1,0])/Delta; // slope component along face normal
69         double st = (eta[0,-1]+eta[-1,-1]-eta[0,1]-eta[-1,1])/4./Delta; // slope component
70         // transverse to face normal
71         sn -= tilt.x ;
72         st -= tilt.y ;
73         double S = sqrt(sq(sn)+sq(st)) + eps_slope ; // desingularize slope
74         D.x[] = 2.*pow(hm,5./3.)/sqrt(S)/(nmanning[]+nmanning[-1,0]) ;
75     }
76     boundary((scalar *){D});
77     return(0);
78 }
79
80 int update_sink(double Dt)
81 {
82     double a = 0.99;
83     foreach() // cell-centered source term compensating for detrending and mass unbalance
84     {
85         beta[] = excess[]>0. ? a/Dt : 0.;
86         r[] = -tilt.x * (D.x[1,0]-D.x[])/Delta - tilt.y * (D.y[0,1]-D.y[])/Delta
87             - (1.-a)*excess[]/Dt;
88         r[] += (excess[]>0.) ? -a*zb[]/Dt : 0.;
89     }
90     boundary({r});
91     return(0);
92 }
93
94 /**
95 # Main
96 */
97
98 int main (int argc, char * argv[])
99 {
100     (void) init_harmo_kinoshita (& K);
101     L0 = npatterns*3400./K.sinuosity;
102     size (L0);
103     origin (0.,-L0/2.);
104     init_grid (1 << LEVEL);
105
106     periodic(right);
107
108     tilt.x = 1.1e-3;
109     tilt.y = 0.;
110
111     run();
112
113     return(0);
114 }
115
116 event init (i=0)
117 {
118     (void) live_terrain ("topo_garonne.smp",& K,"topo");

```

```

119 terrain (zb,"topo", NULL);
120
121 scalar zmin = zb.dmin;
122
123 foreach()
124 {
125     eta[] = zb[]>etaini ? zb[] : etaini;
126     h[] = eta[]-zb[];
127     double zzmin = zmin[]<nodata ? zmin[] : zb[];
128     nmanning[] = zzmin<0. ? 1./Kmin : 1./Kmaj;
129     excess[] = 0.;
130 }
131 boundary({zb,eta,excess});
132 (void) update_diffusivity(zb,eta,D);
133 (void) update_sink(1.0);
134 }
135
136 event integration (i++;t<tmax)
137 {
138     (void) update_diffusivity(zb,eta,D);
139
140     stats sx = statsf (D.x);
141     stats sy = statsf (D.y);
142     double dtExplicit = 0.25*L0*L0/N/N/fmax(sx.max,sy.max);
143     // Set time step to a multiple of explicit time step
144     dt = dtnext(40.*dtExplicit);
145
146     // recompute sink term
147     (void) update_sink(dt);
148
149     mgd = diffusion(eta,dt,D,r,beta);
150
151     // Adapt grid
152     (void) adapt();
153
154     foreach()
155     {
156         excess[] = (eta[]<zb[]) ? zb[]-eta[] : 0.; // extra water needed to maintain eta >= zb
157         eta[] = fmax(eta[],zb[]);
158     }
159
160     boundary({zb,eta,excess});
161 }
162
163 // Adaptivity
164
165 scalar absu[];
166
167 int adapt() {
168
169 #if TREE
170     scalar etamask[], slope[];
171
172     foreach()
173     {
174         h[] = eta[]-zb[] > dry ? eta[]-zb[] : 0.;
175         etamask[] = h[] > dry ? eta[] : 0.;
176         slope[] = h[] > dry ? sqrt( sq(eta[1,0]-eta[-1,0]-2.*tilt.x*Delta) + sq(eta[0,1]-eta
177         [0,-1]-2.*tilt.y*Delta) )/2./Delta : 0.;
178     }
179     boundary ({h,etamask,slope});
180
181     astats s = adapt_wavelet ({h,etamask,slope}, (double []){HE,ETA,SE},
182         LEVEL, MINLEVEL);
183     boundary({h});

```

```

183
184     scalar zmin = zb.dmin;
185
186     foreach(){
187         double zzmin = zmin[]<nodata ? zmin[] : zb[];
188         nmanning[] = zzmin<0. ? 1./Kmin : 1./Kmaj;
189         l[] = level;
190         bool wet_stencil = false;
191         foreach_neighbor(1)
192             wet_stencil = wet_stencil | (h[]>dry);
193         // If 3x3 neighborhood is water-free according to newly adapted h[], just ignore adapted
194         // eta[], set it to newly adapted zb[].
195         // It will disable sink term.
196         eta[] = wet_stencil ? eta[] : zb[];
197     }
198     boundary(all);
199
200 //     fprintf (ferr, "# refined %d cells, coarsened %d cells\n", s.nf, s.nc);
201     return s.nf;
202 #else // Cartesian
203     return 0;
204 #endif
205 }

```


Annexe C

Codes sur Basilisk

C.1 Injection des débits pour le calage du coefficient de Manning en lit mineur

```
1 #include "saint-venant.h"
2 #include "input.h"
3 #include "output.h"
4 #include "esri-binary-grids.h"
5 #include "terrain.h"
6 #include "sourceterm/manning-tilt.h"
7 #include "inflow-tilt.h"
8
9 #define MINLEVEL 6
10 #define ETAE      1e-2 // error on free surface elevation (1 cm)
11 #define HE        1e-2 // error on water depth (1 cm)
12 #define ZE        1e-2 // error on bed elevation (1 cm)
13 #define UE        1e-2 // 0.01 m/s
14 #define tmax      86400.
15 #define sec_per_day 86400.
16 #define NbInlets 6
17 #define Qverdun 90.
18 #define etaini 7.5
19
20 int LEVEL;
21
22 struct Inlet INLETS[NbInlets];
23 double Kmin = 24., Kmaj = 15.;
24 double rotation = -0.43798, xc0 = 565e3, yc0 = 6294e3;
25 FILE * fpcontrol;
26 FILE * results ;
27 double Qlast[NbInlets];
28
29 scalar zte[];
30
31 coord LambertToLocal(coord P)
32 {
33     return (coord) {cos(rotation)*(P.x-xc0)-sin(rotation)*(P.y-yc0) , sin(rotation)*(P.x-xc0)
34                   +cos(rotation)*(P.y-yc0) };
35 }
36 coord LocalToLambert(coord P)
37 {
38     return (coord) {xc0+cos(-rotation)*P.x-sin(-rotation)*P.y, yc0+sin(-rotation)*P.x+cos(-
39                   rotation)*P.y };
40 }
```

```

40 int output_flt_lambert (struct OutputFLT p); // prototype, see definition at the end of the
    file
41
42 int main (int argc, char * argv[])
43 {
44     G = 9.81;
45
46     // Boundary conditions
47     u.n[top] = neumann(0.);
48     u.t[top] = dirichlet(0.);
49     h[top] = neumann(0.);
50     zb[top] = neumann(0.);
51     eta[top] = neumann(0.);
52     nmanning[top] = neumann(0.);
53     zte[top] = neumann(0.);
54
55     run();
56
57     for(int in=0;in<NbInlets;in++)
58         (void) Deallocate_Inlet(INLETS+in);
59
60     if(pid()==0.)
61         fclose(fpcontrol);
62
63     return(0);
64 }
65
66 // Initial conditions
67
68 event init (i = 0)
69 {
70
71     LEVEL = 12;
72     L0 = 32768.;
73
74     tilt.x = 0.;
75     tilt.y = 1.103e-3;
76
77     // Init grid !\ local coordinates, not Lambert 93
78
79     size (L0);
80     X0 = -L0/2.;
81     Y0 = -17068.;
82     origin (X0, Y0);
83     N = 1 << LEVEL;
84     init_grid (1 << LEVEL);
85
86     terrain (zb, "./garonne_2m_local", NULL);
87     boundary({zb});
88
89     terrain (zte, "./enveloppe_4m_local", NULL);
90     boundary({zte});
91
92     char fltfile[200];
93     sprintf(fltfile, "./topo_ini.flt");
94     output_flt(zb,fltfile);
95
96     coord LeftBank,RightBank;
97
98     /** Discharge inlets, numbered from upstream to downstream from 0 to (NbInlets-1)
99         The coordinates of the cross-sections boundaries are defined in Lambert 93
100         hen transformed into local coordinates */
101
102     // Inlet #0 : Garonne in Pont des Catalans
103     int inj=0;

```

```

104 LeftBank = LambertToLocal((coord) {573025.,6279345.});
105 RightBank = LambertToLocal((coord) {573085.,6279655.});
106 (void) Init_Inlet(&INLETS[inj],LeftBank,RightBank);
107 INLETS[inj].ID = inj;
108 INLETS[inj].with_velocity = true;
109 printf("Left bank : (%g,%g) ; Right bank : (%g,%g)\n",INLETS[inj].Segment[0].x,INLETS[inj]
    ].Segment[0].y,INLETS[inj].Segment[1].x,INLETS[inj].Segment[1].y);
110
111 // Inlet #1 : Touch (528 km2)
112 inj++;
113 LeftBank = LambertToLocal((coord){570317.,6281695.});
114 RightBank = LambertToLocal((coord){570352.,6281296.});
115 (void) Init_Inlet(&INLETS[inj],LeftBank,RightBank);
116 INLETS[inj].ID = inj;
117 INLETS[inj].with_velocity = false;
118 printf("Left bank : (%g,%g) ; Right bank : (%g,%g)\n",INLETS[inj].Segment[0].x,INLETS[inj]
    ].Segment[0].y,INLETS[inj].Segment[1].x,INLETS[inj].Segment[1].y);
119
120 // Inlet #2 : Aussonnelle (190 km2)
121 inj++;
122 LeftBank = LambertToLocal((coord){567088.,6289477.});
123 RightBank = LambertToLocal((coord){567355.,6289178.});
124 (void) Init_Inlet(&INLETS[inj],LeftBank,RightBank);
125 INLETS[inj].ID = inj;
126 INLETS[inj].with_velocity = false;
127 printf("Left bank : (%g,%g) ; Right bank : (%g,%g)\n",INLETS[inj].Segment[0].x,INLETS[inj]
    ].Segment[0].y,INLETS[inj].Segment[1].x,INLETS[inj].Segment[1].y);
128
129 // Inlet #3 : Hers-Mort (1524 km2)
130 inj++;
131 LeftBank = LambertToLocal((coord){567779.,6297539.});
132 RightBank = LambertToLocal((coord){567937.,6297906.});
133 (void) Init_Inlet(&INLETS[inj],LeftBank,RightBank);
134 INLETS[inj].ID = inj;
135 INLETS[inj].with_velocity = false;
136 printf("Left bank : (%g,%g) ; Right bank : (%g,%g)\n",INLETS[inj].Segment[0].x,INLETS[inj]
    ].Segment[0].y,INLETS[inj].Segment[1].x,INLETS[inj].Segment[1].y);
137
138 // Inlet #4 : Save (1146 km2)
139 inj++;
140 LeftBank = LambertToLocal((coord){562061.,6298072.});
141 RightBank = LambertToLocal((coord){562447.,6298177.});
142 (void) Init_Inlet(&INLETS[inj],LeftBank,RightBank);
143 INLETS[inj].ID = inj;
144 INLETS[inj].with_velocity = false;
145 printf("Left bank : (%g,%g) ; Right bank : (%g,%g)\n",INLETS[inj].Segment[0].x,INLETS[inj]
    ].Segment[0].y,INLETS[inj].Segment[1].x,INLETS[inj].Segment[1].y);
146
147 // Inlet #5 : Saint-Pierre (135 km2)
148 inj++;
149 LeftBank = LambertToLocal((coord){559482.,6302451.});
150 RightBank = LambertToLocal((coord){559759.,6302163.});
151 (void) Init_Inlet(&INLETS[inj],LeftBank,RightBank);
152 INLETS[inj].ID = inj;
153 INLETS[inj].with_velocity = false;
154 printf("Left bank : (%g,%g) ; Right bank : (%g,%g)\n",INLETS[inj].Segment[0].x,INLETS[inj]
    ].Segment[0].y,INLETS[inj].Segment[1].x,INLETS[inj].Segment[1].y);
155
156 DT = 1.0;
157
158 scalar zmin = zb.dmin;
159 foreach(){
160     nmanning[] = zmin[] < (zte[] + 3.) ? 1./Kmin : 1./Kmaj;
161     eta[] = fmax(etaini,zte[] + 0.4);
162     h[] = eta[] > zb[] ? eta[] - zb[] : 0.;

```

```

163     eta[] = zb[] + h[];
164     u.x[] = 0.;
165     u.y[] = 0.;
166 }
167
168 boundary ((scalar *){u});
169 boundary({h,eta,nmanning});
170
171 for(int inj=0;inj<NbInlets;inj++)
172     Qlast[inj] = 0.;
173
174 if(pid()==0)
175     fpcontrol = fopen("./inflow_control_6inj.txt","w");
176     results = fopen("./outflow_results.txt","w");
177 }
178
179 double etainj;
180 double Qinj[NbInlets] = {0.925*Qverdun,0.15*0.075*Qverdun,0.05*0.075*Qverdun,0.43*0.075*
    Qverdun,0.33*0.075*Qverdun,0.04*0.075*Qverdun};
181 double Tinj[NbInlets] = {0.,3600.,3600.,3600.,3600.,3600.};
182
183 event inflows (i++)
184 {
185     for(int inj=0;inj<NbInlets;inj++){
186         if(t>=Tinj[inj]){
187             double Qtarg = Qlast[inj]+(1.-exp((Tinj[inj]-t)/180.))*(Qinj[inj]-Qlast[inj]);
188             etainj = eta_b ( Qtarg*INLETS[inj].Controller.fQ, &INLETS[inj]);
189             if(etainj>18.0)etainj=18.0;
190
191             (void) set_inlet_fields(&INLETS[inj],etainj);
192         }
193     }
194
195     foreach()
196         eta[] = zb[]+h[];
197
198     boundary ((scalar *){u});
199     boundary({h,eta});
200
201 }
202
203 double etas;
204
205 event inflow_control(i++)
206 {
207     coord Section[2];
208     double Qmes[NbInlets];
209
210     for(int j=0;j<NbInlets;j++){
211
212         // The discharge is controlled slightly downstream of the inlet sections (distance dmes
213         )
214         double dmes = sqrt(2.)*(L0/N);
215         Section[0] = (coord) { INLETS[j].Segment[0].x+dmes*INLETS[j].vec_n.x , INLETS[j].
    Segment[0].y+dmes*INLETS[j].vec_n.y };
216         Section[1] = (coord) { INLETS[j].Segment[1].x+dmes*INLETS[j].vec_n.x , INLETS[j].
    Segment[1].y+dmes*INLETS[j].vec_n.y };
217         // "Measure" discharge
218         Qmes[j] = segment_flux (Section,h,u);
219         if(t<=Tinj[j])Qlast[j] = Qmes[j];
220
221         // Set PID controllers parameters
222         // https://en.wikipedia.org/wiki/PID_controller
223         double Ku = 0.5, Tu = dmes*1.0;
224         INLETS[j].Controller.Kp = 0.2*Ku;

```

```

224     INLETS[j].Controller.Ti = 30.;
225     INLETS[j].Controller.Td = 5.;
226     double Qtarg = t>Tinj[j] ? Qlast[j]+(1.-exp((Tinj[j]-t)/180.))* (Qinj[j]-Qlast[j]) : 0.;
227
228     if(Qmes[j]>0. & Qtarg>0. & fabs(log(INLETS[j].Controller.fQ))<log(10.))
229         (void) update_PID_error(Qmes[j],Qtarg, & (INLETS[j].Controller) );
230     else
231         (void) Reset_PID_Controller( & (INLETS[j].Controller) );
232     }
233
234     double Vol;
235     Vol = 0.;
236
237     foreach(reduction(+:Vol))
238         Vol+=sq(Delta)*h[];
239
240     if(pid()==0){
241         fprintf(fpcontrol, "%g %g %g %g %g %g %g %g %g %g %g %g %g %g %g %g %g %g %g %g %g %g\n", t, dt, Vol,
242             Qmes[0], INLETS[0].Controller.e, INLETS[0].Controller.E, INLETS[0].
243             Controller.fQ ,
244             Qmes[1], INLETS[1].Controller.e, INLETS[1].Controller.E, INLETS[1].
245             Controller.fQ ,
246             Qmes[2], INLETS[2].Controller.e, INLETS[2].Controller.E, INLETS[2].
247             Controller.fQ ,
248             Qmes[3], INLETS[3].Controller.e, INLETS[3].Controller.E, INLETS[3].
249             Controller.fQ ,
250             Qmes[4], INLETS[4].Controller.e, INLETS[4].Controller.E, INLETS[4].
251             Controller.fQ );
252         fflush(fpcontrol);
253     }
254 }
255
256 event outflow(i++)
257 {
258
259     coord Avalcoord[2];
260     double Qaval, etaaval;
261
262     Avalcoord[0] = LambertToLocal((coord) {558575. ,6308295.});
263     Avalcoord[1] = LambertToLocal((coord) {558733. ,6308374.});
264
265     Qaval = segment_flux (Avalcoord,h,u);
266     etaaval = segment_etamean (Avalcoord,h,zb);
267
268     if(pid()==0){
269         fprintf(results, "%g %g %g %g\n", t, dt, Qaval, etaaval);
270         fflush(results);
271     }}
272
273
274 scalar l[];
275
276 event snapshot (t+=600. ; t<=tmax) {
277
278     char fltfile[200];
279     sprintf(fltfile, "./_h_1%d_Q_90_Kmin_%d_Kmaj_%d_%5.5d.flr", LEVEL, (int)Kmin, (int)Kmaj, (int)(t
280         /60.));
281     output_flt(h,fltfile);

```

```

281  sprintf(fltfile, "./ux_l%d_Q_90_Kmin_%d_Kmaj_%d_%5.5d.flt", LEVEL, (int)Kmin, (int)Kmaj, (int)(
282  t/60.));
283  output_flt(u.x, fltfile);
284
285  sprintf(fltfile, "./uy_l%d_Q_90_Kmin_%d_Kmaj_%d_%5.5d.flt", LEVEL, (int)Kmin, (int)Kmaj, (int)(
286  t/60.));
287  output_flt(u.y, fltfile);
288
289  foreach()
290      l[] = level;
291
292  sprintf(fltfile, "./level_l%d_Q_90_Kmin_%d_Kmaj_%d_%5.5d.flt", LEVEL, (int)Kmin, (int)Kmaj, (
293  int)(t/60.));
294  output_flt(l, fltfile);
295  }
296
297  // Adaptivity
298
299  scalar absu[];
300
301  int adapt() {
302  #if TREE
303      scalar eta[];
304      foreach()
305          eta[] = h[] > dry ? h[] + zb[] : 0;
306      boundary ({eta});
307
308      foreach()
309          absu[] = h[] > dry ? sqrt(u.y[]*u.y[]+u.x[]*u.x[]) : 0.;
310
311  /**
312  We can now use wavelet adaptation on the list of scalars *{ ,hmax}*
313  with thresholds *{ETA,EMAXE}*. The compiler is not clever enough yet
314  and needs to be told explicitly that this is a list of *double*s,
315  hence the *(double[])*
316  [type casting](http://en.wikipedia.org/wiki/Type_conversion).
317
318  The function then returns the number of cells refined.*/
319
320  astats s = adapt_wavelet ({h,absu}, (double[]){HE,UE},
321  LEVEL, MINLEVEL);
322
323  // Refine inlets
324
325  for(int in=0; in<NbInlets; in++){
326
327      double xA = INLETS[in].Segment[0].x , xB = INLETS[in].Segment[1].x, yA = INLETS[in].
328      Segment[0].y , yB = INLETS[in].Segment[1].y;
329      coord vec_n = {yA - yB, xB - xA};
330      coord vec_t = {xB - xA, yB - yA};
331      double norm = sqrt(sq(vec_n.x) + sq(vec_n.y));
332      assert (norm > 0.);
333      vec_n.x = vec_n.x/norm + 1e-6, vec_n.y = vec_n.y/norm - 1.5e-6;
334      vec_t.x = vec_t.x/norm, vec_t.y = vec_t.y/norm;
335
336  //   printf(" Refining inlet %d, A = (%g,%g), vec_n = (%g,%g)\n", in, xA, yA, vec_n.x, vec_n.y)
337  ;
338
339      refine (level < LEVEL
340          && fabs((x-xA)*vec_n.x+(y-yA)*vec_n.y) <= 2.83*(L0/(1 << LEVEL))
341          && ((x-xA)*vec_t.x+(y-yA)*vec_t.y) >= 0.
342          && ((x-xA)*vec_t.x+(y-yA)*vec_t.y) <= norm
343          );
344  }

```

```

341 scalar zmin = zb.dmin;
342 foreach(){
343     nmanning[] = zmin[] < (ztc[] + 3.) ? 1./Kmin : 1./Kmaj;
344     l[] = level;
345 }
346
347 boundary(all);
348
349 // fprintf (ferr, "# refined %d cells, coarsened %d cells\n", s.nf, s.nc);
350 return s.nf;
351
352 #else // Cartesian
353     return 0;
354 #endif
355 }
356
357 // And finally we set the event that will apply our *adapt()* function at every timestep.
358 event do_adapt (i++) adapt();
359
360 //////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
361
362 int output_flt_lambert (struct OutputFLT p)
363 {
364     FILE * flt , * hdr ;
365     scalar input = p.f;
366     char * pch;
367     char buffer[100];
368     double * data_double;
369     coord current_point;
370     char hdrfile[200];
371
372     // default values
373     if (p.box[0][0] == 0. && p.box[0][1] == 0. &&
374         p.box[1][0] == 0. && p.box[1][1] == 0.) {
375         p.box[0][0] = X0;      p.box[0][1] = Y0;
376         p.box[1][0] = X0 + L0; p.box[1][1] = Y0 + L0;
377         if (p.Delta == 0) p.Delta = L0/N;
378     }
379
380     double Delta = p.Delta;
381     int nx = (p.box[1][0] - p.box[0][0])/Delta;
382     int ny = (p.box[1][1] - p.box[0][1])/Delta;
383
384     data_double = (double *)malloc(nx*ny*sizeof(double));
385     for(int i=0; i<(nx*ny); i++)
386         data_double[i] = nodata;
387
388     // data
389     for (int j = ny-1; j >= 0; j--) {
390         double yp93 = Delta*j + p.box[0][1] + Delta/2.; // center of pixel
391         for (int i = 0; i < nx; i++) {
392             double xp93 = Delta*i + p.box[0][0] + Delta/2., v;
393             float vf;
394
395             current_point.x = xp93;
396             current_point.y = yp93;
397             current_point = LambertToLocal(current_point);
398             double xp = current_point.x; // local x
399             double yp = current_point.y; // local y
400
401             if (p.mask.i) { // masking
402                 if (p.linear) {
403                     double m = interpolate (p.mask, xp, yp);

```



```

406     if (m < 0.)
407         v = nodata;
408     else
409         v = interpolate (p.f, xp, yp);
410     }
411     else {
412         Point point = locate (xp, yp);
413         if (point.level < 0 || val(p.mask) < 0.)
414             v = nodata;
415         else
416             v = val(p.f);
417     }
418     }
419     else if (p.linear){
420     Point point = locate (xp, yp);
421     v = point.level >= 0 ? interpolate (p.f, xp, yp) : nodata;
422     }
423     else {
424     Point point = locate (xp, yp);
425     v = point.level >= 0 ? val(p.f) : nodata;
426     }
427     if (v == nodata)
428     data_double[i+(ny-1-j)*nx] = -9999.;
429     else
430     data_double[i+(ny-1-j)*nx] = v;
431
432     }
433 }
434
435 if (pid() == 0) { // master writes .FLT and .HDR files
436 @if _MPI
437     MPI_Reduce (MPI_IN_PLACE, data_double, nx*ny, MPI_DOUBLE, MPI_MAX, 0,
438     MPI_COMM_WORLD);
439 @endif
440
441 // Set path to header file
442 // hdrfile = (char *)malloc(sizeof(char)*strlen(p.file));
443 strcpy(hdrfile,p.file);
444
445 pch = hdrfile + strlen(hdrfile)-3;
446 // strncpy (pch,"hdr",3);
447 memcpy (pch,"hdr",3);
448
449 if(!(hdr = fopen (hdrfile, "w")))
450 {
451     printf("Failed to open header file %s\n",hdrfile);
452     return -1;
453 }
454
455 // header
456 fprintf (hdr, "ncols           %d\n", nx);
457 fprintf (hdr, "nrows           %d\n", ny);
458 fprintf (hdr, "xllcorner         %.8g\n", p.box[0][0]);
459 fprintf (hdr, "yllcorner         %.8g\n", p.box[0][1]);
460 fprintf (hdr, "cellsize          %.8g\n", Delta);
461 fprintf (hdr, "nodata_value      -9999\n");
462 fprintf (hdr, "byteorder         LSBFIRST\n");
463
464 fflush(hdr);
465 fclose(hdr);
466 // free(hdrfile);
467
468 // open flt file
469
470 bool opened = false;

```

```

471 if( !(flt = fopen (p.file, "wb")) ) {
472     perror (p.file);
473     exit(1);
474 }
475 else
476     opened = true;
477
478 for(int i=0;i<(nx*ny);i++)
479 {
480     float vf = (float)data_double[i];
481     fwrite(&vf,1,sizeof(float),flt);
482 }
483 fclose(flt);
484 }
485 #if _MPI
486     else // slave does not write anything
487         MPI_Reduce (data_double, NULL, nx*ny, MPI_DOUBLE, MPI_MAX, 0,
488             MPI_COMM_WORLD);
489 #endif
490
491 free(data_double);
492 return(0);
493
494 }

```

C.2 Injection des hydrogrammes pour le calage du coefficient de Manning en lit majeur

```

1 #include "saint-venant.h"
2 #include "input.h"
3 #include "output.h"
4 #include "esri-binary-grids.h"
5 #include "terrain.h"
6 #include "sourceterm/manning-tilt.h"
7 #include "inflow-tilt.h"
8 #include "gauges.h"
9
10 #define MINLEVEL 6
11 #define ETAE 1e-2 // error on free surface elevation (1 cm)
12 #define HE 1e-2 // error on water depth (1 cm)
13 #define ZE 1e-2 // error on bed elevation (1 cm)
14 #define UE 1e-2 // 0.01 m/s
15 #define datenum_ini 730645.0 // Matlab/Scilab datenum() for 2000/06/08 00:00
16 #define tmax 777600. // 9 days
17 #define sec_per_day 86400.
18 #define NbInlets 6
19 #define NbReperes 9
20 #define etaini 8.4
21
22 int LEVEL;
23
24 struct Inlet INLETS[NbInlets];
25 double Kmin = 25., Kmaj = 20.;
26 double rotation = -0.43798, xc0 = 565e3, yc0 = 6294e3;
27 FILE * fpcontrol;
28 FILE * results ;
29 double Qlast[NbInlets];
30 coord ReperesLocal;
31 coord Reperes[NbReperes];
32 scalar zte[], eta_PHE[], h_PHE[];
33
34 coord LambertToLocal(coord P)
35 {

```

```

36     return (coord) {cos(rotation)*(P.x-xc0)-sin(rotation)*(P.y-yc0) , sin(rotation)*(P.x-xc0)
37     +cos(rotation)*(P.y-yc0) };
38 }
39 coord LocalToLambert(coord P)
40 {
41     return (coord) {xc0+cos(-rotation)*P.x-sin(-rotation)*P.y, yc0+sin(-rotation)*P.x+cos(-
42     rotation)*P.y };
43 }
44 int output_flt_lambert (struct OutputFLT p); // prototype, see definition at the end of the
45     file
46 int main (int argc, char * argv[])
47 {
48     G = 9.81;
49 // Boundary conditions
50 u.n[top] = neumann(0.);
51 u.t[top] = dirichlet(0.);
52 h[top] = neumann(0.);
53 zb[top] = neumann(0.);
54 eta[top] = neumann(0.);
55 nmanning[top] = neumann(0.);
56 zte[top] = neumann(0.);
57
58 run();
59
60 for(int in=0;in<NbInlets;in++)
61     (void) Deallocate_Inlet(INLETS+in);
62
63 if(pid()==0.)
64     fclose(fpcontrol);
65
66 return(0);
67 }
68
69 // Initial conditions
70
71 event init (i = 0)
72 {
73
74     LEVEL = 12;
75     L0 = 32768.;
76
77     tilt.x = 0.;
78     tilt.y = 1.103e-3;
79
80 // Init grid /\ local coordinates, not Lambert 93
81
82     size (L0);
83     X0 = -L0/2.;
84     Y0 = -17068.;
85     origin (X0, Y0);
86     N = 1 << LEVEL;
87     init_grid (1 << LEVEL);
88
89     terrain (zb, "./garonne_2m_local", NULL);
90     boundary({zb});
91
92     terrain (zte, "./enveloppe_4m_local", NULL);
93     boundary({zte});
94
95     foreach()
96     {
97         eta_PHE[] = zb[];

```

```

98     h_PHE[] = 0.0;
99 }
100 char fltfile[200];
101 sprintf(fltfile, "./topo_ini.flt");
102 output_flt(zb,fltfile);
103
104     coord LeftBank,RightBank;
105
106 /** Discharge inlets, numbered from upstream to downstream from 0 to (NbInlets-1)
107     The coordinates of the cross-sections boundaries are defined in Lambert 93
108     hen transformed into local coordinates */
109
110 // Inlet #0 : Garonne au Pont des Catalans
111 int inj=0;
112 LeftBank = LambertToLocal((coord) {573025.,6279345.});
113 RightBank = LambertToLocal((coord) {573085.,6279655.});
114 (void) Init_Inlet(&INLETS[inj],LeftBank,RightBank);
115 INLETS[inj].ID = inj;
116 INLETS[inj].with_velocity = true;
117 printf("Left bank : (%g,%g) ; Right bank : (%g,%g)\n",INLETS[inj].Segment[0].x,INLETS[inj]
118 ].Segment[0].y,INLETS[inj].Segment[1].x,INLETS[inj].Segment[1].y);
119 (void) load_hydrograph(&INLETS[inj], ". /input/hydrogrammes/Q_Bazacle.txt",datenum_ini);
120
121 // Inlet #1 : Touch (528 km2)
122 inj++;
123 LeftBank = LambertToLocal((coord){570317.,6281695.});
124 RightBank = LambertToLocal((coord){570352.,6281296.});
125 (void) Init_Inlet(&INLETS[inj],LeftBank,RightBank);
126 INLETS[inj].ID = inj;
127 INLETS[inj].with_velocity = false;
128 printf("Left bank : (%g,%g) ; Right bank : (%g,%g)\n",INLETS[inj].Segment[0].x,INLETS[inj]
129 ].Segment[0].y,INLETS[inj].Segment[1].x,INLETS[inj].Segment[1].y);
130 (void) load_hydrograph(&INLETS[inj], ". /input/hydrogrammes/Q_Touch.txt",datenum_ini);
131
132 // Inlet #2 : Aussonnelle (190 km2)
133 inj++;
134 LeftBank = LambertToLocal((coord){567088.,6289477.});
135 RightBank = LambertToLocal((coord){567355.,6289178.});
136 (void) Init_Inlet(&INLETS[inj],LeftBank,RightBank);
137 INLETS[inj].ID = inj;
138 INLETS[inj].with_velocity = false;
139 printf("Left bank : (%g,%g) ; Right bank : (%g,%g)\n",INLETS[inj].Segment[0].x,INLETS[inj]
140 ].Segment[0].y,INLETS[inj].Segment[1].x,INLETS[inj].Segment[1].y);
141 (void) load_hydrograph(&INLETS[inj], ". /input/hydrogrammes/Q_Aussonnelle.txt",datenum_ini
142 );
143
144 // Inlet #3 : Hers-Mort (1524 km2)
145 inj++;
146 LeftBank = LambertToLocal((coord){567779.,6297539.});
147 RightBank = LambertToLocal((coord){567937.,6297906.});
148 (void) Init_Inlet(&INLETS[inj],LeftBank,RightBank);
149 INLETS[inj].ID = inj;
150 INLETS[inj].with_velocity = false;
151 printf("Left bank : (%g,%g) ; Right bank : (%g,%g)\n",INLETS[inj].Segment[0].x,INLETS[inj]
152 ].Segment[0].y,INLETS[inj].Segment[1].x,INLETS[inj].Segment[1].y);
153 (void) load_hydrograph(&INLETS[inj], ". /input/hydrogrammes/Q_HersMort.txt",datenum_ini);
154
155 // Inlet #4 : Save (1146 km2)
156 inj++;
157 LeftBank = LambertToLocal((coord){562061.,6298072.});
158 RightBank = LambertToLocal((coord){562447.,6298177.});
159 (void) Init_Inlet(&INLETS[inj],LeftBank,RightBank);
160 INLETS[inj].ID = inj;
161 INLETS[inj].with_velocity = false;
162 printf("Left bank : (%g,%g) ; Right bank : (%g,%g)\n",INLETS[inj].Segment[0].x,INLETS[inj]

```

```

158   ].Segment[0].y, INLETS[inj].Segment[1].x, INLETS[inj].Segment[1].y);
159   (void) load_hydrograph(&INLETS[inj], "./input/hydrogrammes/Q_Save.txt", datenum_ini);
160
161   // Inlet #5 : Saint-Pierre (135 km2)
162   inj++;
163   LeftBank = LambertToLocal((coord){559482.,6302451.});
164   RightBank = LambertToLocal((coord){559759.,6302163.});
165   (void) Init_Inlet(&INLETS[inj], LeftBank, RightBank);
166   INLETS[inj].ID = inj;
167   INLETS[inj].with_velocity = false;
168   printf("Left bank : (%g,%g) ; Right bank : (%g,%g)\n", INLETS[inj].Segment[0].x, INLETS[inj].Segment[0].y, INLETS[inj].Segment[1].x, INLETS[inj].Segment[1].y);
169   (void) load_hydrograph(&INLETS[inj], "./input/hydrogrammes/Q_StPierre.txt", datenum_ini);
170
171   // Coordinates of the 2000 flood markers
172
173   // GTL_R_4807 : Aire de repos sur une d viation de camions
174   Reperes[0]= LambertToLocal((coord){562718.,6298836.});
175   // GTL_R_4805 Chemin du G / Chemin des Monges
176   Reperes[1]= LambertToLocal((coord){564008.,6299309.});
177
178   // GTL_R_4804 rampe d accs l ancien pont
179   Reperes[2]= LambertToLocal((coord){565671.,6298820.});
180
181   // GTL_R_4802 1 km en aval de Gagnac
182   Reperes[3]= LambertToLocal((coord){568392.,6290754.});
183
184   // WEB_R_201801164736 1-5 km en aval de Gagnac
185   Reperes[4]= LambertToLocal((coord){568199.,6291494.});
186
187   // GTL_R_4798 3 km en aval des ponts jumeaux
188   Reperes[5]= LambertToLocal((coord){571395.,6282974.});
189
190   // GTL_R_4799 "Amicale du Chien d'utilit "
191   Reperes[6]= LambertToLocal((coord){570981.,6283491.});
192
193   // GTL_R_4800 B timent c t d'un terrain de jeu
194   Reperes[7]= LambertToLocal((coord){570409.,6287624.});
195
196   // GTL_R_4801 Station d puration
197   Reperes[8]= LambertToLocal((coord){568604.,6288410.});
198
199
200   DT = 1.0;
201
202   scalar zmin = zb.dmin;
203   foreach(){
204     nmanning[] = zmin[] < (zte[] + 3.) ? 1./Kmin : 1./Kmaj;
205     eta[] = fmax(etaini, zte[] + 0.2);
206     h[] = eta[] > zb[] ? eta[] - zb[] : 0.;
207     eta[] = zb[] + h[];
208     u.x[] = 0.;
209     u.y[] = 0.;
210   }
211
212   boundary ((scalar *){u});
213   boundary ({h, eta, nmanning});
214
215   for(int inj=0; inj<NbInlets; inj++)
216     Qlast[inj] = 0.;
217
218   if(pid()==0)
219     fpcontrol = fopen("./inflow_control_6inj.txt", "w");
220     results = fopen("./outflow_results.txt", "w");

```



```

282     Qmes[0], INLETS[0].Controller.e, INLETS[0].Controller.E, INLETS[0].
Controller.fQ ,
283     Qmes[1], INLETS[1].Controller.e, INLETS[1].Controller.E, INLETS[1].
Controller.fQ ,
284     Qmes[2], INLETS[2].Controller.e, INLETS[2].Controller.E, INLETS[2].
Controller.fQ ,
285     Qmes[3], INLETS[3].Controller.e, INLETS[3].Controller.E, INLETS[3].
Controller.fQ ,
286     Qmes[4], INLETS[4].Controller.e, INLETS[4].Controller.E, INLETS[4].
Controller.fQ ,
287     Qmes[5], INLETS[5].Controller.e, INLETS[5].Controller.E, INLETS[5].
Controller.fQ );
288     fflush(fpcontrol);
289 }
290 }
291
292 event update_PHE (i++)
293 {
294     foreach()
295     {
296         eta_PHE[] = ( eta[] > eta_PHE[] ) & ( h[]>dry ) ? eta[] : eta_PHE[]; // eta_PHE[]
is updated only if eta[] increases and the cell is not dry at the current iteration
297         h_PHE[] = ( h[] > h_PHE[] ) ? h[] : h_PHE[];
298     }
299 }
300
301 // Adaptivity
302
303 scalar absu[];
304
305 int adapt() {
306 #if TREE
307     scalar eta[];
308     foreach()
309         eta[] = h[] > dry ? h[] + zb[] : 0;
310     boundary ({eta});
311
312     foreach()
313         absu[] = h[] > dry ? sqrt(u.y[]*u.y[]+u.x[]*u.x[]) : 0.;
314
315     /**
316     We can now use wavelet adaptation on the list of scalars *{ ,hmax}*
317     with thresholds *{ETAe,HMAXE}*. The compiler is not clever enough yet
318     and needs to be told explicitly that this is a list of *double*s,
319     hence the *(double[])*
320     [type casting](http://en.wikipedia.org/wiki/Type_conversion).
321
322     The function then returns the number of cells refined.*/
323
324     astats s = adapt_wavelet ({h,eta}, (double[]){HE,ETAe}, LEVEL, MINLEVEL);
325
326     // Refine inlets
327
328     for(int in=0;in<NbInlets;in++){
329
330         double xA = INLETS[in].Segment[0].x , xB = INLETS[in].Segment[1].x, yA = INLETS[in].
Segment[0].y , yB = INLETS[in].Segment[1].y;
331         coord vec_n = {yA - yB, xB - xA};
332         coord vec_t = {xB - xA, yB - yA};
333         double norm = sqrt(sq(vec_n.x) + sq(vec_n.y));
334         assert (norm > 0.);
335         vec_n.x = vec_n.x/norm + 1e-6, vec_n.y = vec_n.y/norm - 1.5e-6;
336         vec_t.x = vec_t.x/norm, vec_t.y = vec_t.y/norm;
337
338     //     printf(" Refining inlet #%d, A = (%g,%g), vec_n = (%g,%g)\n",in,xA,yA,vec_n.x,vec_n.y)

```

```

;
339
340     refine (level < LEVEL
341             && fabs((x-xA)*vec_n.x+(y-yA)*vec_n.y) <= 2.83*(L0/(1 << LEVEL))
342             && ((x-xA)*vec_t.x+(y-yA)*vec_t.y) >= 0.
343             && ((x-xA)*vec_t.x+(y-yA)*vec_t.y) <= norm
344             );
345     }
346
347     for(int rep=0;rep<NbReperes;rep++)
348     {
349         RepereLocal = Reperes[rep];
350         refine (level < LEVEL
351                 && sqrt( sq(x-RepereLocal.x)+sq(y-RepereLocal.y) ) <= 2.*( L0/(1 << LEVEL) )
352                 );
353     }
354
355     scalar zmin = zb.dmin;
356     foreach(){
357         nmanning[] = zmin[] < (zte[]+3.) ? 1./Kmin : 1./Kmaj;
358         l[] = level;
359     }
360
361     boundary(all);
362
363     // fprintf (ferr, "# refined %d cells, coarsened %d cells\n", s.nf, s.nc);
364     return s.nf;
365
366 #else // Cartesian
367     return 0;
368 #endif
369 }
370
371 // And finally we set the event that will apply our *adapt()* function at every timestep.
372
373 event do_adapt (i++) adapt();
374
375 ////////////////////////////////////////////////////////////////////////////////////////////
376
377 int output_flt_lambert (struct OutputFLT p)
378 {
379     FILE * flt , * hdr ;
380     scalar input = p.f;
381     char * pch;
382     char buffer[100];
383     double * data_double;
384     coord current_point;
385     char hdrfile[200];
386
387     // default values
388     if (p.box[0][0] == 0. && p.box[0][1] == 0. &&
389         p.box[1][0] == 0. && p.box[1][1] == 0.) {
390         p.box[0][0] = X0;     p.box[0][1] = Y0;
391         p.box[1][0] = X0 + L0; p.box[1][1] = Y0 + L0;
392         if (p.Delta == 0) p.Delta = L0/N;
393     }
394
395     double Delta = p.Delta;
396     int nx = (p.box[1][0] - p.box[0][0])/Delta;
397     int ny = (p.box[1][1] - p.box[0][1])/Delta;
398
399     data_double = (double *)malloc(nx*ny*sizeof(double));
400     for(int i=0;i<(nx*ny);i++)
401         data_double[i] = nodata;
402

```



```

403 // data
404 for (int j = ny-1; j >= 0; j--) {
405     double yp93 = Delta*j + p.box[0][1] + Delta/2.; // center of pixel
406     for (int i = 0; i < nx; i++) {
407         double xp93 = Delta*i + p.box[0][0] + Delta/2., v;
408         float vf;
409
410         current_point.x = xp93;
411         current_point.y = yp93;
412         current_point = LambertToLocal(current_point);
413         double xp = current_point.x; // local x
414         double yp = current_point.y; // local y
415
416         if (p.mask.i) { // masking
417             if (p.linear) {
418                 double m = interpolate (p.mask, xp, yp);
419                 if (m < 0.)
420                     v = nodata;
421                 else
422                     v = interpolate (p.f, xp, yp);
423             }
424             else {
425                 Point point = locate (xp, yp);
426                 if (point.level < 0 || val(p.mask) < 0.)
427                     v = nodata;
428                 else
429                     v = val(p.f);
430             }
431         }
432         else if (p.linear){
433             Point point = locate (xp, yp);
434             v = point.level >= 0 ? interpolate (p.f, xp, yp) : nodata;
435         }
436         else {
437             Point point = locate (xp, yp);
438             v = point.level >= 0 ? val(p.f) : nodata;
439         }
440         if (v == nodata)
441             data_double[i+(ny-1-j)*nx] = -9999.;
442         else
443             data_double[i+(ny-1-j)*nx] = v;
444     }
445 }
446 }
447
448 if (pid() == 0) { // master writes .FLT and .HDR files
449 #if _MPI
450     MPI_Reduce (MPI_IN_PLACE, data_double, nx*ny, MPI_DOUBLE, MPI_MAX, 0,
451               MPI_COMM_WORLD);
452 #endif
453
454 // Set path to header file
455 // hdrfile = (char *)malloc(sizeof(char)*strlen(p.file));
456 strcpy(hdrfile,p.file);
457
458 pch = hdrfile + strlen(hdrfile)-3;
459 // strncpy (pch,"hdr",3);
460 memcpy (pch,"hdr",3);
461
462 if(!(hdr = fopen (hdrfile, "w")))
463 {
464     printf("Failed to open header file %s\n",hdrfile);
465     return -1;
466 }
467

```

```

468 // header
469 fprintf (hdr, "ncols          %d\n", nx);
470 fprintf (hdr, "nrows          %d\n", ny);
471 fprintf (hdr, "xllcorner      %.8g\n", p.box[0][0]);
472 fprintf (hdr, "yllcorner      %.8g\n", p.box[0][1]);
473 fprintf (hdr, "cellsize       %.8g\n", Delta);
474 fprintf (hdr, "nodata_value   -9999\n");
475 fprintf (hdr, "byteorder     LSBFIRST\n");
476
477 fflush(hdr);
478 fclose(hdr);
479 // free(hdrfile);
480
481 // open flt file
482
483 bool opened = false;
484 if( !(flt = fopen (p.file, "wb")) ) {
485     perror (p.file);
486     exit(1);
487 }
488 else
489     opened = true;
490
491 for(int i=0;i<(nx*ny);i++)
492 {
493     float vf = (float)data_double[i];
494     fwrite(&vf,1,sizeof(float),flt);
495 }
496 fclose(flt);
497 }
498 #if _MPI
499 else // slave does not write anything
500     MPI_Reduce (data_double, NULL, nx*ny, MPI_DOUBLE, MPI_MAX, 0,
501     MPI_COMM_WORLD);
502 #endif
503
504 free(data_double);
505 return(0);
506
507 }

```

C.3 Inversion avec Caramel

```

1 #include "grid/quadtree.h"
2 #include "run.h"
3 #include "diffusion.h"
4 #include "kinoshita.h"
5 #include "terrain.h"
6 #include "live_terrain.h"
7 #include "esri-binary-grids.h"
8
9 #pragma autolink -lfft3
10
11 #define LEVEL 8
12 #define MINLEVEL 6
13 #define ETAE 1.0e-3 // error on free surface elevation (1 cm)
14 #define HE 1.0e-3 // error on water depth (1 cm)
15 #define SE 1.0e-6
16 #define ZE 1.0e-2 // error on bed elevation (1 cm)
17 #define UE 1.0e-2 // 0.01 m/s
18 #define tmax 3600.
19 #define sec_per_day 86400.
20
21 double G = 9.81, dry = 2.*HE;

```

```

22 double hrun,hpool,Kmin,Kmaj;
23 double etaini,prec,faceExplicit;
24 scalar zb[],eta[],h[],nmanning[],l[],r[],beta[],excess[];
25 face vector D[];
26 double * TIME = NULL, * AREA = NULL, * VOLUME = NULL, * MEAN = NULL, * STDEV = NULL, *
    DISCHARGE = NULL;
27 int nsteps,tag;
28
29 double dt, tini;
30 mgstats mgd;
31
32 coord tilt;
33 FILE * flog, * f_out;
34
35 // Garonne:
36 struct Kinoshita K = { .kcurv = 2.*M_PI/3700.,
37                       .theta0 = 0.96,
38                       .alphas = 1./10.,
39                       .alphaf = 1./96.,
40                       .sinuosity = -1. } ;
41
42 double ChannelWidth(double s, struct Kinoshita * _K)
43 {
44     return(200.);
45 }
46
47 double TalwegDepth(double s, struct Kinoshita * _K, double kmax)
48 {
49     double phase = 2.*_K->kcurv*s ; // factor 2 for pi-periodicity
50     double res = 0.5*(hpool-hrun)*cos(phase) + 0.5*(hpool+hrun);
51 // double res = hrun + (hpool-hrun)*fabs(signed_curvature(s,_K))/kmax;
52     return(res);
53 }
54
55 double TopoFloodplain(double phi)
56 {
57     return (0.018*73.*sqrt(M_PI)/2.)*erf((fabs(phi)-36.)/73.) + 0.0052*fabs(phi);
58 }
59
60
61 /**
62  ## Diffusivity function */
63
64 int update_diffusivity(scalar zb, scalar eta, face vector D, bool write)
65 {
66     double eps_slope = sqrt(sq(tilt.x)+sq(tilt.y))/1000.;
67
68 //     /\ reminder : D.x and D.y are not co-located
69     foreach_face()
70     {
71         double hl = eta[-1,0]-zb[-1,0] ;
72         double hm = (eta[]+eta[-1,0]-zb[]-zb[-1,0])/2.; // depth interpolated at face center
73         hm = max(hm,0.);
74         double sn = (eta[]-eta[-1,0])/Delta; // slope component along face normal
75         double st = (eta[0,-1]+eta[-1,-1]-eta[0,1]-eta[-1,1])/4./Delta; // slope component
            transverse to face normal
76         sn -= tilt.x ;
77         st -= tilt.y ;
78         double S = sqrt(sq(sn)+sq(st)) + eps_slope ; // desingularize slope
79         D.x[] = 2.*pow(hm,5./3.)/sqrt(S)/(nmanning[]+nmanning[-1,0]) ;
80     }
81
82     boundary((scalar *){D});
83     return(0);
84 }

```

```

85
86 int update_sink(double Dt)
87 {
88     double a = 0.99;
89     foreach() // cell-centered source term compensating for detrending and mass unbalance
90     {
91         beta[] = excess[]>0. ? a/Dt : 0.;
92         r[] = -tilt.x * (D.x[1,0]-D.x[])/Delta - tilt.y * (D.y[0,1]-D.y[])/Delta - (1.-a)*excess
           []/Dt;
93         r[] += (excess[]>0.) ? -a*zb[]/Dt : 0.;
94     }
95     boundary({r});
96     return(0);
97 }
98
99 bool test_convergence( double * tt, double * val, int n, double prec, double timefrac)
100 {
101     if(n<50) return(false);
102
103     double lastval = val[n-1];
104     int j = (n-2);
105     while( fabs(val[j]-lastval)<prec*lastval )
106         j--;
107
108     return (tt[n-1]-tt[j+1])>((tt[n-1]-tt[0])*timefrac) ;
109 }
110
111 /**
112 # Main
113 */
114
115 int main (int argc, char * argv[])
116 {
117     (void) init_harmo_kinoshita (& K);
118     LO = npatterns*3700./K.sinuosity;
119     size (LO);
120     origin (0.,-LO/2.);
121     init_grid (1 << LEVEL);
122
123     // mask (x > (LO/2.) ? right : none);
124     periodic(right);
125
126     tilt.x = 1.1e-3;
127     tilt.y = 0.;
128
129     // Read parameters
130
131     char inputfile[300], outputfile[300];
132     char buffer[200];
133     strcpy (inputfile, argv[1]);
134     strcpy (outputfile, argv[2]);
135
136
137     FILE * f_in = fopen(inputfile,"r");
138     fgets(buffer, 200, f_in);
139     fclose(f_in);
140     sscanf(buffer, "%d %lf %lf %lf %lf %lf %lf %lf", &tag, &hrun, &hpool, &Kmin, &Kmaj, &etaini, &prec
           , &facExplicit);
141
142     f_out = fopen(outputfile, "w");
143
144     /* char ratingfile[200];
145     sprintf(ratingfile, "rating_%5.5d.dat", tag);
146     flog = fopen(ratingfile, "w");
147 */

```

```

148
149 // Run
150
151 run();
152
153 // fclose(flog);
154 fclose(f_out);
155
156 free(TIME);
157 free(AREA);
158 free(VOLUME);
159 free(MEAN);
160 free(STDEV);
161
162 return(0);
163 }
164
165 event init (i=0)
166 {
167     char kdtroot[200];
168     sprintf(kdtroot, "./topo_%5.5d", tag);
169
170     (void) live_terrain ("./topo_garonne_caramel_v2.smp",
171                        & K, kdtroot);
172
173     terrain (zb, kdtroot, NULL);
174 //     output_flt(zb, "topo_ini.flt");
175
176     scalar zmin = zb.dmin;
177
178     foreach()
179     {
180         eta[] = zb[] > etaini ? zb[] : etaini;
181         h[] = eta[] - zb[];
182         double zzmin = zmin[] < nodata ? zmin[] : zb[];
183         nmanning[] = zzmin < 0. ? 1./Kmin : 1./Kmaj;
184         excess[] = 0.;
185     }
186     boundary({zb, eta, excess});
187     (void) update_diffusivity(zb, eta, D, true);
188     (void) update_sink(1.0);
189 }
190
191 event write_log(i++)
192 {
193     // Compute flooded area, volume,
194     // mean and variance of detrended water surface elevation
195
196     double V = 0., A = 0.;
197     double mean_eta = 0., var_eta = 0.;
198
199     foreach()
200     {
201         h[] = eta[] - zb[] > dry ? eta[] - zb[] : 0.;
202         if(h[] > 0.)
203         {
204             double dA = sq(Delta);
205             A += dA;
206             V += dA * h[];
207             mean_eta += dA * eta[];
208             var_eta += dA * sq(eta[]);
209         }
210     }
211
212     mean_eta /= A;

```

```

213 var_eta = fmax(var_eta/A - mean_eta*mean_eta,0.) ;
214
215 // Compute discharge
216
217 double Q = 0., Lcheck = 0.;
218 double xlim = L0/2./((double)npatterns)-L0/(3.*(double)N);
219
220 foreach()
221 {
222     if(x>=xlim & (x-Delta)<xlim)
223     {
224         Q += Delta * D.x[] * ( tilt.x + ( eta[-1,0]-eta[] )/Delta ); // diffusive flux
225         Lcheck += Delta;
226     }
227 }
228
229 TIME = (double *) realloc(TIME , (i+1)*sizeof(double));
230 AREA = (double *) realloc(AREA , (i+1)*sizeof(double));
231 VOLUME = (double *) realloc(VOLUME , (i+1)*sizeof(double));
232 MEAN = (double *) realloc(MEAN , (i+1)*sizeof(double));
233 STDEV = (double *) realloc(STDEV , (i+1)*sizeof(double));
234 DISCHARGE = (double *) realloc(DISCHARGE , (i+1)*sizeof(double));
235
236 TIME[i] = t;
237 AREA[i] = A;
238 VOLUME[i] = V;
239 MEAN[i] = mean_eta;
240 STDEV[i] = sqrt(var_eta);
241 DISCHARGE[i] = Q;
242 nsteps = i+1;
243
244 // fprintf(flog,"%d %.5e %.5e %.5e %.5e %.5e %.5e\n",i,t,A,V,mean_eta,var_eta,Q,Lcheck
-L0);
245 // fflush(flog);
246 }
247
248 event integration (i++;t<tmax)
249 {
250     (void) update_diffusivity(zb,eta,D,false);
251
252     stats sx = statsf (D.x);
253     stats sy = statsf (D.y);
254     double dtExplicit = 0.25*L0*L0/N/N/fmax(sx.max,sy.max);
255     dt = dtnext(facExplicit*dtExplicit);
256
257     // recompute sink term at each time step (but not diffusivity)
258     (void) update_sink(dt);
259
260     mgd = diffusion(eta,dt,D,r,beta);
261
262     (void) adapt();
263
264     foreach()
265     {
266         excess[] = (eta[]<z b[]) ? z b[]-eta[] : 0.; // extra water needed to maintain eta >= z b
267         eta[] = fmax(eta[],z b[]);
268     }
269
270     boundary({z b,eta,excess});
271 }
272
273 // Adaptivity
274
275 scalar absu[];
276

```

```

277 int adapt() {
278
279 #if TREE
280     scalar etamask[], slope[];
281
282     foreach()
283     {
284         h[] = eta[]-zb[] > dry ? eta[]-zb[] : 0.;
285         etamask[] = h[] > dry ? eta[] : 0.;
286         slope[] = h[] > dry ? sqrt( sq(eta[1,0]-eta[-1,0]-2.*tilt.x*Delta) + sq(eta[0,1]-eta
[0,-1]-2.*tilt.y*Delta) )/2./Delta : 0.;
287     }
288     boundary ({h,etamask,slope});
289
290
291     astats s = adapt_wavelet ({h,etamask,slope}, (double []){HE,ETAE,SE},
292         LEVEL, MINLEVEL);
293     boundary({h});
294
295     scalar zmin = zb.dmin;
296
297     foreach(){
298         double zzmin = zmin[]<nodata ? zmin[] : zb[];
299         nmanning[] = zzmin<0. ? 1./Kmin : 1./Kmaj;
300         l[] = level;
301         bool wet_stencil = false;
302         foreach_neighbor(1)
303             wet_stencil = wet_stencil | (h[]>dry);
304         // If 3x3 neighborhood is water-free according to newly adapted h[], just ignore adapted
305         // eta[], set it to newly adapted zb[].
306         // It will disable sink term.
307         eta[] = wet_stencil ? eta[] : zb[];
308     }
309     boundary(all);
310
311     // fprintf (ferr, "# refined %d cells, coarsened %d cells\n", s.nf, s.nc);
312     return s.nf;
313 #else // Cartesian
314     return 0;
315 #endif
316 }
317
318 // And finally we set the event that will apply our *adapt()* function at every timestep.
319
320 //event do_adapt (i++) adapt();
321
322 event stop (t+=60.) {
323     if ( (test_convergence( TIME, DISCHARGE, nsteps, prec, 0.33) & test_convergence( TIME,
STDEV, nsteps, prec, 0.33)) | (t>=tmax) )
324         return 1;
325     // the default is to return zero
326 }
327
328 event finalize (t = end) {
329     // this will happen after the simulation has completed due to
330     // the "stop" event above
331     fprintf(f_out,"%d %.4f %.6e %.6e %.6e %.6e %.6e\n",(int)(t>=tmax),t,*(AREA + (nsteps-1))
,*(VOLUME + (nsteps-1)),*(MEAN + (nsteps-1)),*(STDEV + (nsteps-1)),*(DISCHARGE + (nsteps
-1)));
332 }

```

C.4 Fonctions développées sur Basilisk

C.4.1 Fonction *kinoshita.h*

```
1 /**
2 # Toolbox for computing the Kinoshita curve using FFT
3 */
4
5 #include <fftw3.h>
6 #include <complex.h>
7
8 #define M_PI acos(-1.0)
9 #define npatterns 1
10 #define nharmo 8 // nharmo < N/(2*npatterns)
11
12 struct Kinoshita {
13     double kcurv;
14     double theta0;
15     double alphas;
16     double alphaf;
17     double sinuosity;
18 };
19
20 double complex KinoFFT[2*nharmo+1];
21
22 double complex tangent_unit_vector (double s, struct Kinoshita * _K )
23 {
24     double th3 = pow(_K->theta0,3.0);
25     double theta = _K->theta0*sin(_K->kcurv*s)
26                 + _K->alphas*th3*cos(3.*_K->kcurv*s)
27                 - _K->alphaf*th3*sin(3.*_K->kcurv*s);
28
29     return cexp(I*theta);
30 }
31
32 double signed_curvature (double s, struct Kinoshita * _K )
33 {
34     double k_th3 = pow(_K->theta0,3.0)*(_K->kcurv);
35     double phase = (_K->kcurv)*s;
36     double sc = (_K->theta0)*(_K->kcurv)*cos(phase)
37                - 3.*(_K->alphas)*k_th3*sin(3.*phase)
38                - 3.*(_K->alphaf)*k_th3*cos(3.*phase);
39     return sc;
40 }
41
42 double complex get_kinoshita_position(double s, struct Kinoshita * _K)
43 {
44     double complex res = s/(_K->sinuosity);
45     double phase = (_K->kcurv)*s;
46
47     for(int m=-nharmo;m<(nharmo+1);m++)
48         res += KinoFFT[m+nharmo]*(cos(m*phase)+I*sin(m*phase));
49
50     return res;
51 }
52
53 int init_harmo_kinoshita ( struct Kinoshita * _K )
54 {
55     fftw_complex *in, *out;
56     fftw_plan p;
57     in = (fftw_complex*) fftw_malloc(sizeof(fftw_complex) * N);
58     out = (fftw_complex*) fftw_malloc(sizeof(fftw_complex) * N);
59
60     // initialize input array
```



```

61 double Lcurv = 2.*M_PI/(_K->kcurv);
62
63
64 for(int m=0;m<N;m++)
65 {
66     double sm = Lcurv*m/((double)N);
67     double complex Tm = tangent_unit_vector(sm,_K);
68     in[m][0] = creal(Tm);
69     in[m][1] = cimag(Tm);
70 }
71
72 // compute forward Fourier transform of T = dK_ds
73
74 p = fftw_plan_dft_1d(N, in, out, FFTW_FORWARD, FFTW_ESTIMATE);
75 fftw_execute(p);
76 fftw_destroy_plan(p);
77
78 // retrieve sinuosity (inverse of real part of 1st element out[0] */
79
80 double sinuosity = N/out[0][0];
81 _K->sinuosity = sinuosity;
82 printf("sinuosity = %g\n",sinuosity);
83 in[0][0] = 0.;
84 in[0][1] = 0.;
85
86 // integrate in Fourier space
87
88 for(int m=1;m<N;m++)
89 {
90     double wavenum = m<(N/2) ? (_K->kcurv)*m : (_K->kcurv)*(m-N);
91     in[m][0] = (1./wavenum)*out[m][1];
92     in[m][1] = -(1./wavenum)*out[m][0];
93 }
94
95 // compute inverse Fourier transform to get detrended talweg curve
96
97 p = fftw_plan_dft_1d(N, in, out, FFTW_BACKWARD, FFTW_ESTIMATE);
98 fftw_execute(p);
99 fftw_destroy_plan(p);
100
101 // Decomposition on the basis of exp(i kcurv m s)
102
103 for(int m=-nharmo;m<(nharmo+1);m++)
104     KinoFFT[m+nharmo] = (m>=0) ? in[m][0]/N + I*in[m][1]/N : in[N+m][0]/N + I*in[N+m][1]/N
105     ;
106
107 return(0);
108 }
109
110 /**
111 Function to get s(x,y): project point (x,y) on curve and get corresponding curvilinear
112     abscissa
113 */
114 coord get_curvilinear_coord ( coord P, struct Kinoshita * _K )
115 {
116     double d2min = 1.e30;
117     double sproj = 0.;
118
119     // get first estimate of projection
120
121     for(int m=0;m<N;m++)
122     {
123         double srel = m/((double)N);

```

```

124     double sm = (L0*_K->sinuosity)*(-1+(npatterns+2)*srel);
125     double complex Km = get_kinoshita_position(sm,_K);
126     double d2 = sq(P.x-creal(Km))+sq(P.y-cimag(Km));
127     if(d2<d2min){
128         d2min = d2;
129         sproj = sm;
130     }
131 }
132
133 // refine position
134
135 double pscal;
136 double complex Kproj,Tproj,dTproj,KprojP;
137
138 do {
139     double complex Kproj = get_kinoshita_position(sproj,_K);
140     Tproj = tangent_unit_vector(sproj,_K);
141     dTproj = signed_curvature(sproj,_K)*I*Tproj;
142     KprojP = P.x + I*P.y - Kproj;
143     pscal = creal(KprojP)*creal(Tproj) + cimag(KprojP)*cimag(Tproj);
144     double aa = creal(KprojP)*creal(dTproj) + cimag(KprojP)*cimag(dTproj) - 1.0;
145     sproj -= 0.5*(pscal/aa);
146 } while(fabs(pscal)>(1e-6)*cabs(KprojP));
147
148 // return curvilinear coordinates
149
150 pscal = creal(KprojP)*creal(I*Tproj) + cimag(KprojP)*cimag(I*Tproj);
151 double dist = pscal>=0. ? cabs(KprojP) : -cabs(KprojP);
152
153 // return sproj;
154 return (coord){sproj,dist};
155 }

```

C.4.2 Fonction *live_terrain.h*

```

1 #include "terrain.h"
2 #include "kinoshita.h"
3
4 double ChannelWidth(double, struct Kinoshita *);
5 double TalwegDepth(double, struct Kinoshita *,double);
6 double TopoFloodplain(double);
7
8 int batch_kdt (double * X, double * Y, double * Z, long n, const char * name)
9 {
10     int pagesize = 4096;
11
12     // FILE * fp = fopen("dump.smp","w");
13
14     KdtHeap h;
15     kdt_heap_create (&h, kdt_tmpfile (), 0, -1, 1000000);
16     KdtPoint p;
17     for(long i = 0;i<n;i++){
18         p.x = X[i];
19         p.y = Y[i];
20         p.z = Z[i];
21         kdt_heap_put (&h, &p);
22     //     fprintf(fp,"% .2f % .2f % .3f\n",p.x,p.y,p.z);
23     }
24     kdt_heap_flush (&h);
25
26     Kdt * kdt = kdt_new ();
27     kdt_create (kdt,name,pagesize,&h,NULL, NULL);
28     kdt_destroy (kdt);
29
30     // fclose(fp);

```

```

31     return 0;
32 }
33
34 double complex lookup_table_C(double x, int n, double * X, double complex * Y, double
    complex * V, double complex * _vn)
35 {
36     // find upper bound
37
38     int i=0;
39     while( X[i]<x & i<(n-1))
40         i++;
41
42     // interpolate between i and i+1
43
44     double complex res = Y[i-1] + (x-X[i-1])*(Y[i]-Y[i-1])/(X[i]-X[i-1]);
45
46     double complex vn = V[i-1] + (x-X[i-1])*(V[i]-V[i-1])/(X[i]-X[i-1]);
47     *_vn = vn/cabs(vn);
48
49     return(res);
50 }
51
52 double lookup_table_R(double x, int n, double * X, double * V)
53 {
54     // find upper bound
55
56     int i=0;
57     while( X[i]<x & i<(n-1))
58         i++;
59
60     // interpolate between i and i+1
61
62     double res = V[i-1] + (x-X[i-1])*(V[i]-V[i-1])/(X[i]-X[i-1]);
63     return(res);
64 }
65
66 static double falsepos_phi (double Width, double complex * _tvec, double * PHI_TAB, double
    complex * Z_TAB, double complex * V_TAB, int ntab,
67     double binf, double winf,
68     double bsup, double wsup)
69 {
70     int n = 0;
71     double prec = 1e-6;
72     double newb, neww, angle;
73     double complex dZ, vn1, vn2;
74     winf -= Width; // width deficit at binf compared to target
75     wsup -= Width; // width excess at bsup compared to target
76     do {
77         newb = (binf*wsup - bsup*winf)/(wsup - winf); // linear interpolation
78         dZ = lookup_table_C(+1.0*newb, ntab, PHI_TAB, Z_TAB, V_TAB, &vn1)
79             - lookup_table_C(-1.0*newb, ntab, PHI_TAB, Z_TAB, V_TAB, &vn2);
80         // * _tvec = vn1*csqrt(vn2/vn1);
81         angle = cimag(clog(dZ/(*_tvec)));
82         neww = cabs(dZ)*cos(angle) - Width;
83         if (neww > 0.)
84             bsup = newb, wsup = neww;
85         else
86             binf = newb, winf = neww;
87         n++;
88     } while (fabs(neww/Width) > prec && n < 100);
89     if (n >= 100)
90         fprintf (stderr, "WARNING: falsepos_phi(): convergence not reached\n");
91
92     return newb;
93 }

```

```

94
95 int live_terrain(char * smpfile, struct Kinoshita * _K, const char * name)
96 {
97     // Get geometry of channel tubular neighborhood
98
99     char buffer[200];
100     double * s_grid = NULL, * t_grid = NULL;
101     double ** x_grid = NULL, ** y_grid = NULL, ** phi_grid = NULL;
102     double complex ** v_grid = NULL;
103     int ns = 0, nt = 0, tcurr;
104     double sp, tp, xp, yp, PHIp, VXp, VYp;
105
106     FILE * smpdata = fopen(smpfile, "r");
107     double sp_prev = -1.0e30;
108
109     while( fgets(buffer, 200, smpdata) != NULL ) // parse until end-of-file
110     {
111         sscanf(buffer, "%lf %lf %lf %lf %lf %lf %lf", &sp, &tp, &xp, &yp, &PHIp, &VXp, &VYp);
112
113         if(sp > sp_prev) // new section
114         {
115             ns++;
116             tcurr = 1;
117             s_grid = (double *) realloc(s_grid, ns * sizeof(double));
118             s_grid[ns-1] = sp;
119             sp_prev = sp;
120             x_grid = (double **) realloc(x_grid, ns * sizeof(double *));
121             y_grid = (double **) realloc(y_grid, ns * sizeof(double *));
122             phi_grid = (double **) realloc(phi_grid, ns * sizeof(double *));
123             v_grid = (double complex **) realloc(v_grid, ns * sizeof(double complex *));
124             x_grid[ns-1] = NULL;
125             y_grid[ns-1] = NULL;
126             phi_grid[ns-1] = NULL;
127             v_grid[ns-1] = NULL;
128
129             if(ns > 1) // not the 1st section => nt is known
130             {
131                 x_grid[ns-1] = (double *) malloc(nt * sizeof(double));
132                 y_grid[ns-1] = (double *) malloc(nt * sizeof(double));
133                 phi_grid[ns-1] = (double *) malloc(nt * sizeof(double));
134                 v_grid[ns-1] = (double complex *) malloc(nt * sizeof(double complex));
135             }
136         }
137         else
138             tcurr++;
139
140         if(ns == 1)
141         {
142             nt = tcurr;
143             t_grid = (double *) realloc(t_grid, tcurr * sizeof(double));
144             t_grid[tcurr-1] = tp;
145             x_grid[0] = (double *) realloc(x_grid[0], tcurr * sizeof(double));
146             y_grid[0] = (double *) realloc(y_grid[0], tcurr * sizeof(double));
147             phi_grid[0] = (double *) realloc(phi_grid[0], tcurr * sizeof(double));
148             v_grid[0] = (double complex *) realloc(v_grid[0], tcurr * sizeof(double complex));
149         }
150
151         x_grid[ns-1][tcurr-1] = xp;
152         y_grid[ns-1][tcurr-1] = yp;
153         phi_grid[ns-1][tcurr-1] = PHIp;
154         v_grid[ns-1][tcurr-1] = -VXp - I * VYp;
155     }
156     fclose(smpdata);
157     fprintf(stderr, "%d,%d grid in tubular coordinates\n", ns, nt);
158

```

```

159 double * ksigned = (double *) malloc(ns*sizeof(double));
160 double kmax = -1.0e30;
161 int indmax = -1;
162
163 // Locate section of maximum curvature
164
165 for(int sec=0;sec<ns;sec++)
166 {
167     ksigned[sec] = signed_curvature (s_grid[sec],_K);
168     if( fabs(ksigned[sec])>kmax )
169     {
170         kmax = fabs(ksigned[sec]);
171         indmax = sec;
172     }
173 }
174
175 // Compute maximum value of half-width to full width ratio, amax, at this section
176
177 double Width = ChannelWidth(s_grid[indmax],_K); // get channel width at current section
178 double complex tvec = I * tangent_unit_vector (s_grid[indmax],_K);
179
180 double complex * Zsec = (double complex *) malloc(nt*sizeof(double complex));
181 double complex * Vsec = (double complex *) malloc(nt*sizeof(double complex));
182
183 for(int j=0;j<nt;j++)
184 {
185     Zsec[j] = x_grid[indmax][j] + I * y_grid[indmax][j];
186     Vsec[j] = v_grid[indmax][j];
187 }
188
189 double bsup = 0., wsup = -1.0e30;
190 double complex dZ,vn1,vn2;
191 double angle;
192
193 while(wsup<1.1*Width)
194 {
195     bsup+=0.1;
196     dZ = lookup_table_C(+1.0*bsup,nt,phi_grid[indmax],Zsec,Vsec,&vn1)
197         - lookup_table_C(-1.0*bsup,nt,phi_grid[indmax],Zsec,Vsec,&vn2);
198 //     tvec = vn1*csqrt(vn2/vn1);
199     angle = cimag(clog(dZ/tvec));
200     wsup = cabs(dZ)*cos(angle);
201 }
202 double phi = falsepos_phi(Width,&tvec,phi_grid[indmax], Zsec,Vsec,nt,0.,0.,bsup,wsup);
203
204 double tl = lookup_table_R(+1.0*phi,nt,phi_grid[indmax],t_grid);
205 double tr = lookup_table_R(-1.0*phi,nt,phi_grid[indmax],t_grid);
206 double amax = (tl>-tr) ? tl/(tl-tr) : -tr/(tl-tr);
207 fprintf(stderr,"phi = %.3f, tr = %.2f, tl = %.2f, amax = %.3f\n",phi,tr,tl,amax);
208
209 // Build channel topography
210
211 double * Xkdt = (double *) malloc(ns*nt*sizeof(double));
212 double * Ykdt = (double *) malloc(ns*nt*sizeof(double));
213 double * Zkdt = (double *) malloc(ns*nt*sizeof(double));
214
215 int nsamp = 0;
216
217 for(int sec=0;sec<ns;sec++)
218 {
219     double Width = ChannelWidth(s_grid[sec],_K); // get channel width at current section
220     double Depth = TalwegDepth(s_grid[sec],_K,kmax); // get max depth at current section
221     double complex tvec = I * tangent_unit_vector (s_grid[sec],_K);
222
223     double Wleft = Width * (0.5 + (ksigned[sec]/kmax) * (amax-0.5) );

```

```

224     double Wright = Width-Wleft;
225
226     for(int j=0;j<nt;j++)
227     {
228         Xkdt[nsamp] = x_grid[sec][j];
229         Ykdt[nsamp] = y_grid[sec][j];
230         double wrel = t_grid[j]>0. ? t_grid[j]/Wleft : fabs(t_grid[j]/Wright);
231         Zkdt[nsamp] = wrel <= 1.0 ? -1.0*Depth*cos(0.5*M_PI*wrel) : TopoFloodplain(phi_grid[
sec][j]);
232         nsamp++;
233     }
234 }
235
236 // Write KDT files
237
238 (void) batch_kdt (Xkdt,Ykdt,Zkdt,(long) nsamp,name);
239
240 // Free arrays
241
242 free(Xkdt);
243 free(Ykdt);
244 free(Zkdt);
245 free(s_grid);
246 free(t_grid);
247
248 for(int sec=0;sec<ns;sec++)
249 {
250     free(x_grid[sec]);
251     free(y_grid[sec]);
252     free(phi_grid[sec]);
253     free(v_grid[sec]);
254 }
255 free(x_grid);
256 free(y_grid);
257 free(phi_grid);
258 free(v_grid);
259
260 return(0);
261 }

```

C.4.3 Fonction *esri-binary-grids.h*

```

1 /**
2 ## *output_flt()*: ESRI Binary Float Grid (.FLT,.HDR) format
3
4 The Esri Binary Float Grid format consists of 2 files: a binary .FLT image file and ASCII .
5 HDR header file with the same file name but different file extension. For example, Test.
6 FLT and Test.HDR.
7 http://surferhelp.goldensoftware.com/subsys/subsys\_ESRI\_binary\_float\_grid\_file\_descr.htm
8
9 The arguments are:
10
11 *f*
12 : a scalar field (compulsory).
13
14 *file*
15 : an array of char, path to the output file (with .FLT extension)
16
17 $\Delta$
18 : size of a grid element. Default is 1/N.
19
20 *linear*
21 : whether to use bilinear or first-order interpolation. Default is
22 first-order.

```

```

22 *box*
23 : the lower-left and upper-right coordinates of the domain to consider.
24 Default is the entire domain.
25
26 *mask*
27 : if set, this field will be used to mask out, the regions
28 of the domain for which *mask* is negative. */
29
30 struct OutputFLT {
31     scalar f;
32     char * file;
33     double Delta;
34     bool linear;
35     double box[2][2];
36     scalar mask;
37 };
38
39 trace
40 int output_flt (struct OutputFLT p)
41 {
42     FILE * flt , * hdr ;
43     scalar input = p.f;
44     char * pch;
45     char buffer[100];
46     double * data_double;
47     char hdrfile[200];
48
49     // default values
50     if (p.box[0][0] == 0. && p.box[0][1] == 0. &&
51         p.box[1][0] == 0. && p.box[1][1] == 0.) {
52         p.box[0][0] = X0;      p.box[0][1] = Y0;
53         p.box[1][0] = X0 + L0; p.box[1][1] = Y0 + L0;
54         if (p.Delta == 0) p.Delta = L0/N;
55     }
56
57     double Delta = p.Delta;
58     int nx = (p.box[1][0] - p.box[0][0])/Delta;
59     int ny = (p.box[1][1] - p.box[0][1])/Delta;
60
61     data_double = (double *)malloc(nx*ny*sizeof(double));
62     for(int i=0;i<(nx*ny);i++)
63         data_double[i] = nodata;
64
65     // data
66     for (int j = ny-1; j >= 0; j--) {
67         double yp = Delta*j + p.box[0][1] + Delta/2.; // center of pixel
68         for (int i = 0; i < nx; i++) {
69             double xp = Delta*i + p.box[0][0] + Delta/2., v;
70             float vf;
71             if (p.mask.i) { // masking
72             if (p.linear) {
73                 double m = interpolate (p.mask, xp, yp);
74                 if (m < 0.)
75                     v = nodata;
76                 else
77                     v = interpolate (p.f, xp, yp);
78             }
79             else {
80                 Point point = locate (xp, yp);
81                 if (point.level < 0 || val(p.mask) < 0.)
82                     v = nodata;
83                 else
84                     v = val(p.f);
85             }
86         }

```

```

87     else if (p.linear)
88     v = interpolate (p.f, xp, yp);
89     else {
90     Point point = locate (xp, yp);
91     v = point.level >= 0 ? val(p.f) : nodata;
92     }
93     if (v == nodata)
94     data_double[i+(ny-1-j)*nx] = -9999.;
95     else
96     data_double[i+(ny-1-j)*nx] = v;
97
98     }
99 }
100
101     if (pid() == 0) { // master writes .FLT and .HDR files
102 @if _MPI
103     MPI_Reduce (MPI_IN_PLACE, data_double, nx*ny, MPI_DOUBLE, MPI_MAX, 0,
104     MPI_COMM_WORLD);
105 @endif
106
107 // Set path to header file
108 //  hdrfile = (char *)malloc(sizeof(char)*strlen(p.file));
109     strcpy(hdrfile,p.file);
110
111     pch = hdrfile + strlen(hdrfile)-3;
112 //  strncpy (pch,"hdr",3);
113     memcpy (pch,"hdr",3);
114
115     if(!(hdr = fopen (hdrfile, "w")))
116     {
117         printf("Failed to open header file %s\n",hdrfile);
118         return -1;
119     }
120
121 // header
122     fprintf (hdr, "ncols          %d\n", nx);
123     fprintf (hdr, "nrows          %d\n", ny);
124     fprintf (hdr, "xllcorner      %.8g\n", p.box[0][0]);
125     fprintf (hdr, "yllcorner      %.8g\n", p.box[0][1]);
126     fprintf (hdr, "cellsize       %.8g\n", Delta);
127     fprintf (hdr, "nodata_value   -9999\n");
128     fprintf (hdr, "byteorder    LSBFIRST\n");
129
130     fflush(hdr);
131     fclose(hdr);
132 //  free(hdrfile);
133
134 //  open flt file
135
136     bool opened = false;
137     if( !(flt = fopen (p.file, "wb")) ) {
138         perror (p.file);
139         exit(1);
140     }
141     else
142         opened = true;
143
144     for(int i=0;i<(nx*ny);i++)
145     {
146         float vf = (float)data_double[i];
147         fwrite(&vf,1,sizeof(float),flt);
148     }
149     fclose(flt);
150 }
151 @if _MPI

```



```

152     else // slave does not write anything
153         MPI_Reduce (data_double, NULL, nx*ny, MPI_DOUBLE, MPI_MAX, 0,
154             MPI_COMM_WORLD);
155 @endif
156
157     free(data_double);
158     return(0);
159 }
160
161 int SYSTEM_IS_LITTLE_ENDIAN(void)
162 {
163     // First, we declare a 16-bit integer (short int), which has the value 0x0001
164     short int word = 0x0001;
165     // then we get a pointer and dereference it
166     char *b = (char *)&word;
167     // If the LSB is stored at lower address (e.g. the value that pointer points to), then
168     // system is LSBFIRST i.e. Little Endian
169     return (b[0] ? 1 : 0);
170 }
171
172 float ReverseFloat( float inFloat )
173 {
174     float retVal;
175     char *floatToConvert = ( char* ) & inFloat;
176     char *returnFloat = ( char* ) & retVal;
177
178     // swap the bytes into a temporary buffer
179     returnFloat[0] = floatToConvert[3];
180     returnFloat[1] = floatToConvert[2];
181     returnFloat[2] = floatToConvert[1];
182     returnFloat[3] = floatToConvert[0];
183
184     return retVal;
185 }
186
187 void lower_string(char s[]) {
188     int c = 0;
189
190     while (s[c] != '\0') {
191         if (s[c] >= 'A' && s[c] <= 'Z') {
192             s[c] = s[c] + 32;
193         }
194         c++;
195     }
196 }
197
198 /**
199 # *input_flt()*: ESRI Binary Float Grid (.FLT,.HDR) format
200
201 The Esri Binary Float Grid format consists of 2 files: a binary .FLT image file and ASCII .
202 HDR header file with the same file name but different file extension. For example, Test.
203 FLT and Test.HDR.
204 http://surferhelp.goldensoftware.com/subsys/subsys\_ESRI\_binary\_float\_grid\_file\_descr.htm
205 This is the reciprocal function of [*output_flt()*](esri-binary-grids.h#output_flt).
206
207 The arguments and their default values are:
208
209 *s*
210 : the scalar where the data will be stored. No default value. You
211 must specify this parameter
212
213 *file*
214 : the name of the file to read from.
215 The path to the header file is assumed to be the same as the binary file with the extension

```

```

214     ".hdr"
215 *nodatavalue*
216 : the value of the NoDataValue. Default is the same as that defined in
217 the raster file.
218
219 *linear*
220 : if true, the raster data is bilinearly interpolated. Default is false.
221 */
222
223 struct InputFLT {
224     scalar s;
225     char * file;
226     double nodatavalue;
227     bool linear;
228 };
229
230 int input_flt (struct InputFLT p)
231 {
232     FILE * flt , * hdr ;
233     scalar input = p.s;
234     char * hdrfile, * pch;
235     char buffer[100];
236
237     double DeltaGRD;
238     int nx, ny,i,j;
239     double XG0, YG0, ndv,center,dx,dy;
240     bool lsbfirst,reverse_bytes;
241     float f;
242
243 // Get path to header file
244     hdrfile = (char *)malloc(sizeof(char)*strlen(p.file));
245     strcpy(hdrfile,p.file);
246
247     pch = hdrfile + strlen(hdrfile)-3;
248     strncpy (pch,"hdr",3);
249
250     if(!(hdr = fopen (hdrfile, "r")))
251     {
252         printf("Failed to open header file %s\n",hdrfile);
253         return -1;
254     }
255
256 // Parse header file
257
258     while(fscanf(hdr,"%s",buffer)!=EOF)
259     {
260         lower_string(buffer);
261
262         if(!(strcmp(buffer,"ncols")))
263             fscanf(hdr,"%d",& nx);
264
265         if(!(strcmp(buffer,"nrows")))
266             fscanf(hdr,"%d",& ny);
267
268         if(!(strcmp(buffer,"cellsize")))
269             fscanf(hdr,"%lf",& DeltaGRD);
270
271         if(!(strcmp(buffer,"xllcenter")))
272         {
273             fscanf(hdr,"%lf",& XG0);
274             center=1.0;
275         }
276
277         if(!(strcmp(buffer,"yllcenter")))

```

```

278     {
279         fscanf(hdr,"%lf",& YGO);
280         center=1.0;
281     }
282
283     if(!(strcmp(buffer,"xllcorner")))
284     {
285         fscanf(hdr,"%lf",& XGO);
286         center=0.0;
287     }
288
289     if(!(strcmp(buffer,"yllcorner")))
290     {
291         fscanf(hdr,"%lf",& YGO);
292         center=0.0;
293     }
294
295     if(!(strcmp(buffer,"nodata_value")))
296         fscanf(hdr,"%lf",& ndv);
297
298     if(!(strcmp(buffer,"byteorder")))
299     {
300         fscanf(hdr,"%s",buffer);
301         lower_string(buffer);
302         lsbfirst = !strcmp(buffer,"lsbfirst");
303     }
304 }
305
306 XGO -= 0.5*DeltaGRD*center;
307 YGO -= 0.5*DeltaGRD*center;
308
309 fclose(hdr);
310
311 bool opened = false;
312 if( !(flt = fopen (p.file, "rb")) ) {
313     perror (p.file);
314     exit(1);
315 }
316 else
317     opened = true;
318
319 //default value of NoData value
320 if (!p.nodatavalue)
321     p.nodatavalue = ndv;
322
323 // Allocation of pointers to columns' start (size : nx)
324 double ** value ;
325 if( !(value = (double **) malloc(nx*sizeof(double *))) ) )
326     return(-1);
327
328 // Allocation of each column (each of size ny)
329 for(i=0;i<nx;i++)
330 {
331     if ( !(value[i] = (double *) malloc(ny*sizeof(double))) ) )
332         return(-1);
333 }
334
335 // read the data (float to double conversion and byte reverse if necessary)
336
337 if ( ( lsbfirst && !SYSTEM_IS_LITTLE_ENDIAN() ) |
338     ( !lsbfirst && SYSTEM_IS_LITTLE_ENDIAN() ) )
339     reverse_bytes = true;
340 else
341     reverse_bytes = false;
342

```

```

343 for(j = ny-1; j >= 0; j--){
344 for(i = 0 ; i < nx; i++){
345     fread(&f,sizeof(float),1,flt);
346     if (reverse_bytes)
347         f = ReverseFloat(f);
348     value[i][j] = f ;
349 }}
350
351 printf(" %d values read\n",nx*ny);
352
353 double LGx0 = nx*DeltaGRD;
354 double LGy0 = ny*DeltaGRD;
355 bool warning = false;
356 bool internal,incl;
357 double val;
358 double onehalf=1./2.;
359 double xllcenter = XGO+DeltaGRD/2., yllcenter = YGO+DeltaGRD/2.;
360
361 foreach() {
362
363     dx = (x - xllcenter)/DeltaGRD ; // relative offset in x w.r.t. center of LL pixel
364     i = (int)floor(dx);
365
366     dy = (y - yllcenter)/DeltaGRD ; // relative offset in y w.r.t. center of LL pixel
367     j = (int)floor(dy);
368
369     internal = (i >= 0 &&
370                i < nx-1 &&
371                j >= 0 &&
372                j < ny-1 );
373
374     if (p.linear && internal ) { // bi-linear interpolation
375
376         dx -= i;
377         dy -= j;
378
379         val = (1.-dx)*(1.-dy)*value[i][j]
380 + dx*(1.-dy)*value[i+1][j]
381 + (1.-dx)*dy*value[i][j+1]
382         + dx*dy*value[i+1][j+1];
383     }
384     else {
385         // Test if the point in the Basilisk area is included in the raster area
386         incl = (x >= XGO &&
387                x <= XGO + LGx0 &&
388                y >= YGO &&
389                y <= YGO + LGy0 );
390         if(incl) {
391             // snap relative offsets dx and dy to nearest integers
392             i = (int)(dx+onehalf);
393             j = (int)(dy+onehalf);
394             val = value[i][j];
395         }
396         else {
397             val = p.nodatavalue;
398             warning = true;
399         }
400     }
401 }
402
403 input[] = val ;
404 }
405
406 // Deallocation of each column
407 for(i=0;i<nx;i++)

```

```

408     free(value[i]);
409
410     // Deallocation of pointers to columns' start
411     free(value);
412     free(hdrfile);
413
414     if (warning)
415         fprintf (stderr,
416                 "input_flt(): Warning: Raster data is not covering all"
417                 " the simulation area\n");
418
419     fclose (flt);
420     return(0);
421 }

```

C.4.4 Fonction *inflow-tilt.h*

```

1 /**
2 # Toolbox for handling discharge inflows in a 2D domain without masks (MPI compatible)
3
4 ## Structures
5
6 The inflow procedure relies on the definition of an inlet segment (left bank to right bank),
7   which is then "rasterized" on grid faces. We first define a couple of useful structures
8   :
9
10 - the *BoundaryFace* structure, which is the basic element constituting the rasterized inlet
11   segment
12 */
13
14 typedef struct {
15     coord Center;
16     coord InwardNormal;
17     double Delta;
18 } BoundaryFace;
19
20 /**
21 - the *PID_controller* structure which implements a [proportional-integral-derivative](https
22   ://en.wikipedia.org/wiki/PID_controller) control loop. It will help regulate inflow
23   discharge according to the difference between the desired setpoint and the "measured"
24   discharge.*/
25 */
26
27 typedef struct {
28     double Kp;
29     double Ti;
30     double Td;
31     double e; // current error, e = -log(Q/Qsetpoint)
32     double E; // integral of error
33     double de_dt; // derivative
34     double fQ; // control variable, such that eta = eta_b(fQ*Q,inlet)
35 } PID_Controller;
36
37 /**
38 - the *Inlet* structure itself, basically composed of an array of *BoundaryFace* structures,
39   and fitted with a PID controller.
40 */
41
42 struct Inlet {
43     coord Segment[2]; // LEFT bank to RIGHT bank
44     coord vec_n; // inward normal
45     coord vec_t;
46     // Rasterization on grid faces ("staircase" version of the boundary):
47     int nf;
48     BoundaryFace * BFaces;
49     PID_Controller Controller;

```

```

42 // char Name[100];
43 int ID;
44 bool with_velocity;
45 int nQ;
46 double * Time;
47 double * Q;
48 };
49
50 /**
51 ## Initialization functions
52
53 ### Function to reset the PID controller
54 */
55
56 int Reset_PID_Controller(PID_Controller * _C)
57 {
58     _C->e = 0.;
59     _C->E = 0.;
60     _C->de_dt = 0.;
61     _C->fQ = 1.0;
62
63     return(0);
64 }
65
66 /**
67 ### Function to initialize the inlet structure
68
69 Most importantly, this function performs the rasterization of the inlet segment on the grid.
70 This rasterization is shared by all processes.
71 */
72 int Init_Inlet(struct Inlet * _inlet, coord LeftBank, coord RightBank)
73 {
74     scalar pscaln[], pscalt[];
75     double xA = LeftBank.x, yA = LeftBank.y, xB = RightBank.x, yB = RightBank.y;
76     (_inlet->Segment)[0] = (coord){xA,yA};
77     (_inlet->Segment)[1] = (coord){xB,yB};
78
79     double norm = sqrt(sq(xB-xA) + sq(yB-yA));
80     assert (norm > 0.);
81
82     // tangent and normal unit vector to the inflow section
83     _inlet->vec_t = (coord){ (xB-xA)/norm , (yB-yA)/norm };
84     _inlet->vec_n = (coord){ (yA-yB)/norm + 1.e-6 , (xB-xA)/norm - 1.5e-6};
85
86     (void) Reset_PID_Controller(&(_inlet->Controller));
87
88     /* Rasterize [AB] segment on grid faces */
89
90     // Compute scalar products for each cell center
91     foreach(){
92         pscaln[] = (x-xA)*(_inlet->vec_n.x) + (y-yA)*(_inlet->vec_n.y);
93         if(pscaln[]==0.) pscaln[] = -(1e-6); // "on the boundary" means "outside"
94         pscalt[] = (x-xA)*(_inlet->vec_t.x) + (y-yA)*(_inlet->vec_t.y);
95     }
96     boundary({pscaln,pscalt});
97
98     double xf,yf;
99     int nf = 0.;
100     int iig,jjg;
101     BoundaryFace * sendFaces = NULL;
102
103     foreach_face(x)
104     {
105         xf = x;
106         yf = y;

```

```

106     iig = pscaln[]>0. ? -ig : ig; // *inward* pointing normal
107     double facesize = Delta;
108
109     // check whether face-sharing cells are on either side of the segment
110     if( (pscaln[]*pscaln[-1,0]<0.) && (pscalt[]>=0.) && (pscalt[-1,0]>=0) && (pscalt[]<=norm
111     ) && (pscalt[-1,0]<=norm) )
112     {
113         // locate "outer" cell
114         Point point = locate(xf-iig*L0/N/2.,yf);
115
116         if(point.level>0) // keep face only if "outer", pseudo-ghost cell is in the sub-domain
117         (avoids duplicates when using MPI)
118         {
119             sendFaces = (BoundaryFace *) realloc(sendFaces,(nf+1)*sizeof(BoundaryFace));
120             sendFaces[nf].Center.x = xf;
121             sendFaces[nf].Center.y = yf;
122             sendFaces[nf].InwardNormal.x = (double) iig;
123             sendFaces[nf].InwardNormal.y = 0.;
124             sendFaces[nf].Delta = facesize;
125             nf++;
126         }
127     } // end if
128 } // end foreach_face(x)
129
130 foreach_face(y)
131 {
132     xf = x;
133     yf = y;
134     jjg = pscaln[]>0. ? -jg : jg; // *inward* pointing normal
135     double facesize = Delta;
136
137     // check whether face-sharing cells are on either side of the segment
138     if( (pscaln[]*pscaln[0,-1]<=0.) && (pscalt[]>=0.) && (pscalt[0,-1]>=0) && (pscalt[]<=
139     norm) && (pscalt[0,-1]<=norm) )
140     {
141         // locate "outer" cell
142         Point point = locate(xf,yf-jjg*L0/N/2.);
143
144         if(point.level>0) // keep face only if "outer", pseudo-ghost cell is in the sub-domain
145         (avoids duplicates when using MPI)
146         {
147             sendFaces = (BoundaryFace *) realloc(sendFaces,(nf+1)*sizeof(BoundaryFace));
148             sendFaces[nf].Center.x = xf;
149             sendFaces[nf].Center.y = yf;
150             sendFaces[nf].InwardNormal.x = 0.;
151             sendFaces[nf].InwardNormal.y = (double) jjg;
152             sendFaces[nf].Delta = facesize;
153             nf++;
154         }
155     } // end if
156 } // end foreach_face(y)
157
158 // We want all the processes to share the full rasterized boundary, so we gather the
159 results
160 #if _MPI
161     int nfa[npe()], nfat[npe()];
162     int nft;
163     nfat[0] = 0;
164     // First count how many faces where found by each process, put the results in array nfa
165     MPI_Allgather (&nf, 1, MPI_INT, &nfa[0], 1, MPI_INT, MPI_COMM_WORLD);
166
167     // Compute displacements
168     for(int j=1;j<npe();j++)
169         nfat[j] = nfat[j-1]+nfa[j-1];
170

```

```

166 // Compute total number of faces
167 nft = nfat[npe()-1]+nfa[npe()-1];
168 // Allocate receiver buffer in the structure
169 _inlet->BFaces = malloc(sizeof(BoundaryFace)*nft);
170 // Scale size of counts (nfa) & displacements (nfat) to have them in bytes
171 for(int j=0;j<npe();j++){
172     nfa[j] *= sizeof(BoundaryFace);
173     nfat[j] *= sizeof(BoundaryFace);
174 }
175 // Gather
176 MPI_Allgatherv (&sendFaces[0], nfa[pid()], MPI_BYTE,
177                &(_inlet->BFaces[0]),nfa,nfat,MPI_BYTE,
178                MPI_COMM_WORLD);
179 _inlet->nf = nft;
180 @else
181 _inlet->BFaces = malloc(sizeof(BoundaryFace)*nf);
182 for(int j=0;j<nf;j++){
183     _inlet->BFaces[j].Center.x = sendFaces[j].Center.x;
184     _inlet->BFaces[j].Center.y = sendFaces[j].Center.y;
185     _inlet->BFaces[j].InwardNormal.x = sendFaces[j].InwardNormal.x;
186     _inlet->BFaces[j].InwardNormal.y = sendFaces[j].InwardNormal.y;
187     _inlet->BFaces[j].Delta = sendFaces[j].Delta;
188 }
189 _inlet->nf = nf;
190 @endif
191
192 free(sendFaces);
193
194 _inlet->nQ = 0;
195 _inlet->Time = NULL;
196 _inlet->Q = NULL;
197
198 return(0);
199 }
200
201 int Deallocate_Inlet(struct Inlet * _inlet)
202 {
203     free(_inlet->BFaces);
204     _inlet->nf = 0.;
205     return(0);
206 }
207
208 /**
209 ## Discharge functions
210
211 The functions provided in this section are essentially a copy of the functions available in
212 [discharge.h](http://basilisk.fr/src/discharge.h): we seek the value  $\eta_b$  of the
213 water surface elevation upstream of the inlet section, which yields the desired inflow
214 discharge through the inlet section. The only difference is in the *bflux* function:
215 because we do not have a true boundary condition on  $z_b$  (typically Neumann), we have
216 to perform the hydrostatic reconstruction of interface elevation for each boundary face
217 composing the inlet section, just as in the Saint-Venant solver.
218
219 */
220
221 struct Eta_b {
222     // compulsory arguments
223     double Q_b;
224     struct Inlet * ptr_inlet;
225     // optional arguments
226     double prec; // precision (default 0.1%)
227 };
228
229 static double bflux (struct Eta_b p, double eta_b)
230 {
231     double Q = 0.;

```



```

225 char fluxfile[200];
226 bool write_fluxes = false ; // (p.ptr_inlet->ID)>0;
227 FILE * fp ;
228 if(write_fluxes){
229     sprintf(fluxfile,"out/bflux/inlet_%d_t_%.3f_pid_%d.csv",p.ptr_inlet->ID,t,pid());
230     fp = fopen(fluxfile,"w");
231     fprintf(fp,"%s\n","x,y,ig,jg,eta_b,zb[out],h[in],dQ");
232 }
233
234 // Center of inlet is taken as reference point for "lake-at-rest" correction with tilt
235 double xcent = (p.ptr_inlet->Segment[0].x + p.ptr_inlet->Segment[1].x)/2.;
236 double ycent = (p.ptr_inlet->Segment[0].y + p.ptr_inlet->Segment[1].y)/2.;
237
238 for(int k = 0;k<(p.ptr_inlet->nf);k++)
239 {
240     int iig = (int) p.ptr_inlet->BFaces[k].InwardNormal.x;
241     int jjg = (int) p.ptr_inlet->BFaces[k].InwardNormal.y;
242     // locate "outer", pseudo-ghost cell (is in subdomain by construction)
243     double xout = (p.ptr_inlet->BFaces[k].Center.x) - iig*L0/N/2;
244     double yout = (p.ptr_inlet->BFaces[k].Center.y) - jjg*L0/N/2;
245     Point point = locate(xout,yout);
246
247     if(point.level>0){
248
249     /** As the function can be used with [topography detrending](manning-tilt.h), we also
250     correct  $\eta_b$  in the inlet area as a function of pseudo-ghost cell coordinates  $(x_{\text{out}}, y_{\text{out}})$  so that the water surface elevation truly corresponds to a
251     lake-at-rest condition:
252     */
253     double corr_eta = tilt.x*(xout-xcent)+tilt.y*(yout-ycent);
254
255     /** Left / right states reconstruction (see [here](http://basilisk.fr/src/saint-venant.h#
256     computing-fluxes) for details): */
257
258     double dx = p.ptr_inlet->BFaces[k].Delta ;
259
260     double up = iig*u.x[iig,0]+jjg*u.y[0,jjg]; // projection of inner velocity on face
261     inward normal
262     double upp = iig*u.x[2*iig,0]+jjg*u.y[0,2*jjg];
263     double un = p.ptr_inlet->with_velocity ? iig*u.x[]+jjg*u.y[] : 0.;
264     double unn = p.ptr_inlet->with_velocity ? iig*u.x[-iig,0]+jjg*u.y[0,-jjg] : 0.;
265
266     double etap = eta[iig,jjg]; // inner cell
267     double etapp = eta[2*iig,2*jjg];
268     double etan = eta_b + corr_eta; // outer, pseudo-ghost cell
269     double etann = eta_b + corr_eta - iig*dx*tilt.x - jjg*dx*tilt.y;
270
271     double hp = h[iig,jjg];
272     double hpp = h[2*iig,2*jjg];
273     double hn = etan - zb[];
274     double hnn = etann - zb[-iig,-jjg];
275
276     // Estimate gradients
277
278     double gr_eta_p = eta.gradient(etan,etap,etapp)/dx;
279     double gr_eta_n = eta.gradient(etann,etan,etap)/dx;
280     double gr_h_p = h.gradient(hn,hp,hpp)/dx;
281     double gr_h_n = h.gradient(hnn,hn,hp)/dx;
282     double gr_u_p = u.x.gradient(un,up,upp)/dx;
283     double gr_u_n = u.x.gradient(unn,un,up)/dx;
284
285     // Left/Right state reconstruction
286
287     double zl = (etap-hp) - (dx/2.)*(gr_eta_p - gr_h_p);
288     double zr = (etan-hn) + (dx/2.)*(gr_eta_n - gr_h_n);

```

```

285     double zlr = max(zl, zr);
286
287     double hl = hp - (dx/2.)*gr_h_p;
288     double ul = up - (dx/2.)*gr_u_p;
289     hl = max(0., hl + zl - zlr);
290
291     double hr = hn + (dx/2.)*gr_h_n;
292     double ur = un + (dx/2.)*gr_u_n;
293     hr = max(0., hr + zr - zlr);
294
295     hp = hl;
296     up = ul;
297     hn = hr;
298     un = ur;
299
300     if (hp > dry || hn > dry) {
301         double fh, fu, dtmax;
302         kurganov (hn, hp, un, up, 0., &fh, &fu, &dtmax);
303         double dQ = fh*(p.ptr_inlet->BFaces[k].Delta);
304         Q += dQ;
305         if(write_fluxes)
306             fprintf(fp,"%g,%g,%d,%d,%g,%g,%g,%g\n", (p.ptr_inlet->BFaces[k].Center.x), (p.
ptr_inlet->BFaces[k].Center.y),iig,jjg,eta_b,zb[],hp,dQ);
307     }
308 } // end if point.level>0
309 } // end for
310
311 // Reduction
312
313 if(write_fluxes)
314     fclose(fp);
315
316 #if _MPI
317     MPI_Allreduce (MPI_IN_PLACE, &Q, 1, MPI_DOUBLE, MPI_SUM, MPI_COMM_WORLD);
318 #endif
319
320 // printf("Q(eta=%g)=%g\n",eta_b,Q);
321
322     return Q;
323
324 }
325
326
327 static double falsepos (struct Eta_b p,
328     double binf, double qinf,
329     double bsup, double qsup)
330 {
331     int n = 0;
332     double newb, newq;
333     qinf -= p.Q_b;
334     qsup -= p.Q_b;
335     do {
336         newb = (binf*qsup - bsup*qinf)/(qsup - qinf);
337         newq = bflux (p, newb) - p.Q_b;
338         if (newq > 0.)
339             bsup = newb, qsup = newq;
340         else
341             binf = newb, qinf = newq;
342         n++;
343     } while (fabs(newq/p.Q_b) > p.prec && n < 100);
344
345     if (n >= 100)
346         fprintf (stderr, "WARNING: eta_b(): convergence not reached\n");
347
348     return newb;

```

```

349 }
350
351 double eta_b (struct Eta_b p)
352 {
353     double zmin = HUGE, etas = 0., hs = 0.;
354
355     for(int k = 0;k<(p.ptr_inlet->nf);k++)
356     {
357         int iig = (int) p.ptr_inlet->BFaces[k].InwardNormal.x;
358         int jjg = (int) p.ptr_inlet->BFaces[k].InwardNormal.y;
359         // locate "outer" cell
360         double xint = (p.ptr_inlet->BFaces[k].Center.x) - iig*L0/N/2;
361         double yint = (p.ptr_inlet->BFaces[k].Center.y) - jjg*L0/N/2;
362         Point point = locate(xint,yint);
363         if(point.level>0){
364             if (zb[] < zmin)
365                 zmin = zb[];
366             etas += (p.ptr_inlet->BFaces[k].Delta)*h[]*eta[];
367             hs += (p.ptr_inlet->BFaces[k].Delta)*h[];
368         }
369     }
370
371     // Reduction
372
373     #if _MPI
374     MPI_Allreduce (MPI_IN_PLACE, &zmin, 1, MPI_DOUBLE, MPI_MIN, MPI_COMM_WORLD);
375     MPI_Allreduce (MPI_IN_PLACE, &etas, 1, MPI_DOUBLE, MPI_SUM, MPI_COMM_WORLD);
376     MPI_Allreduce (MPI_IN_PLACE, &hs, 1, MPI_DOUBLE, MPI_SUM, MPI_COMM_WORLD);
377     #endif
378
379     if (p.Q_b <= 0.)
380         return zmin - 1.;
381
382     // We try to find good bounds on the solution.
383
384     double etasup = hs > 0. ? etas/hs : zmin;
385     double Qsup = bflux (p, etasup), etainf = zmin, Qinf = 0.;
386     double h0 = etasup - zmin;
387     if (h0 < dry)
388         h0 = 1.;
389     int n = 0;
390     while (Qsup < p.Q_b && n++ < 100) {
391         etainf = etasup, Qinf = Qsup;
392         etasup += h0;
393         Qsup = bflux (p, etasup);
394     }
395     if (n >= 100)
396         fprintf (stderr, "WARNING: eta_b() not converged\n");
397
398     if (!p.prec) p.prec = 0.001; // 0.1% by default
399     return falsepos (p, etainf, Qinf, etasup, Qsup);
400 }
401
402 double segment_flux (coord segment[2], scalar h, vector u) // vient de "layered/hydro.h"
403 {
404     coord m = {segment[0].y - segment[1].y, segment[1].x - segment[0].x};
405     normalize (&m);
406     double tot = 0.;
407
408     foreach_segment (segment, p) {
409         double dl = 0.;
410         foreach_dimension() {
411             double dp = (p[1].x - p[0].x)*Delta/Delta_x*(fm.y[] + fm.y[0,1])/2.;
412             dl += sq(dp);
413         }

```

```

414     dl = sqrt (dl);
415     for (int i = 0; i < 2; i++) {
416         coord a = p[i];
417         tot += dl/2.*
418 interpolate_linear (point, (struct _interpolate)
419                     {h, a.x, a.y, 0.})*
420 (m.x*interpolate_linear (point, (struct _interpolate)
421                          {u.x, a.x, a.y, 0.}) +
422  m.y*interpolate_linear (point, (struct _interpolate)
423                          {u.y, a.x, a.y, 0.}));
424     }
425 }
426 // reduction
427 #if _MPI
428 MPI_Allreduce (MPI_IN_PLACE, &tot, 1, MPI_DOUBLE, MPI_SUM, MPI_COMM_WORLD);
429 #endif
430 return tot;
431 }
432
433 double segment_etamean (coord segment[2], scalar h, scalar zb)
434 {
435     char etafile[200];
436     bool write_eta = false ; // (p.ptr_inlet->ID)>0;
437     FILE * fpeta ;
438     if(write_eta){
439         sprintf(etafile,"out/eta/calculeta_pid_%d.txt",pid());
440         fpeta = fopen(etafile,"w");
441         fprintf(fpeta,"%s\n","x,y,dl,zb,h,dA_i,toteta,Atot");
442     }
443     double toteta = 0.;
444     double Atot = 0.;
445     double dA_i;
446     foreach_segment (segment, p) {
447         double dl = 0.;
448         foreach_dimension() {
449             double dp = (p[1].x - p[0].x)*Delta/Delta_x*(fm.y[] + fm.y[0,1])/2.;
450             dl += sq(dp);
451         }
452         dl = sqrt (dl);
453         for (int i = 0; i < 2; i++) {
454             coord a = p[i];
455             dA_i = dl/2.*interpolate_linear (point, (struct _interpolate){h, a.x, a.y, 0.});
456             toteta += dA_i*(interpolate_linear (point, (struct _interpolate){h, a.x, a.y, 0.})
457 + interpolate_linear (point, (struct _interpolate){zb, a.x, a.y, 0.}));
458             Atot += dA_i;
459             if(write_eta)
460                 fprintf(fpeta,"%g,%g,%g,%g,%g,%g,%g,%g\n", a.x,a.y,dl,interpolate_linear (point, (
461 struct _interpolate){zb, a.x, a.y, 0.}), interpolate_linear (point, (struct _interpolate
462 )){h, a.x, a.y, 0.}), dA_i,toteta,Atot);
463         }
464     }
465     if(write_eta)
466         fclose(fpeta);
467     // reduction
468     #if _MPI
469     MPI_Allreduce (MPI_IN_PLACE, &toteta, 1, MPI_DOUBLE, MPI_SUM, MPI_COMM_WORLD);
470     MPI_Allreduce (MPI_IN_PLACE, &Atot, 1, MPI_DOUBLE, MPI_SUM, MPI_COMM_WORLD);
471     #endif
472     if (Atot <= 0)
473         return -9999.;
474     else
475         return toteta/Atot;
476 }

```

```

477 int set_inlet_fields(struct Inlet * _inlet, double etain)
478 {
479     double xcent = (_inlet->Segment[0].x + _inlet->Segment[1].x)/2.;
480     double ycent = (_inlet->Segment[0].y + _inlet->Segment[1].y)/2.;
481
482     foreach(){ // set eta to etain for all points in alcove
483         double xrel = x-(_inlet->Segment)[0].x , yrel = y-(_inlet->Segment)[0].y;
484         double norm = sqrt(sq(_inlet->Segment[0].x-_inlet->Segment[1].x)+sq(_inlet->Segment
485 [0].y-_inlet->Segment[1].y));
486         double psn = xrel*(_inlet->vec_n.x) + yrel*(_inlet->vec_n.y);
487         double pst = xrel*(_inlet->vec_t.x) + yrel*(_inlet->vec_t.y);
488         if(psn<0. && (psn>=-500.) && pst>=0 && pst<=norm)
489             {
490                 double corr_eta = tilt.x*(x-xcent)+tilt.y*(y-ycent);
491                 h[] = fmax(0.,etain+corr_eta-zb[]);
492                 if( !(_inlet->with_velocity) | h[]<dry){
493                     u.x[] = 0.;
494                     u.y[] = 0.;
495                 }
496             }
497     } // end foreach()
498
499     // Set pseudo-Neumann velocity on segment outer cells
500
501     for(int k = 0;k<(_inlet->nf);k++)
502     {
503         int iig = (int) _inlet->BFaces[k].InwardNormal.x;
504         int jjg = (int) _inlet->BFaces[k].InwardNormal.y;
505         // locate "outer" cell (is in subdomain by construction)
506         double xint = (_inlet->BFaces[k].Center.x) - iig*L0/N/2;
507         double yint = (_inlet->BFaces[k].Center.y) - jjg*L0/N/2;
508         Point point = locate(xint,yint);
509         if(point.level>0){
510             double corr_eta = tilt.x*(xint-xcent)+tilt.y*(yint-ycent);
511             h[] = fmax(0.,etain+corr_eta-zb[]);
512             if( !(_inlet->with_velocity) | h[]<dry){
513                 u.x[] = 0.;
514                 u.y[] = 0.;
515             }
516         }
517     } // end for(int k...
518
519     // end
520
521     return(0);
522 }
523
524 int update_PID_error(double Qm, double Qsp, PID_Controller * _C)
525 {
526     double newe = -log(Qm/Qsp); // setpoint-measurement, in log
527     double de_dt = (newe-(_C->e))/dt;
528     double rho = exp(-dt/(_C->Ti));
529     // Integral error with exp(t-t0) weight function (~Gauss-Laguerre quadrature)
530     _C->E = rho*(_C->E) + de_dt*dt + ((_C->e)-de_dt*_C->Ti)*(1.-rho);
531
532     double newcontrol = (_C->Kp)*(newe
533                         + (_C->E)
534                         + de_dt*_C->Td)
535                       );
536     _C->e = newe;
537     (_C->fQ) *= exp(newcontrol);
538
539     return(0);
540 }

```

```

541 /**
542  ** Managing inflow hydrographs
543  */
544
545 /**
546  *** Read hydrograph from file
547  */
548
549 This function has 3 arguments: a pointer to an Inlet structure, the path to the text file
550 containing (time,Q) pairs, and an offset expressed in fractional days. In this
551 implementation all times are expressed in fractional days as computed by Matlab or
552 Scilab (e.g. 2022/01/01 14:30:00 is 738522.604167). As the file is parsed, times are
553 converted to seconds elapsed from the offset "datenum0".
554
555 */
556
557 int load_hydrograph(struct Inlet * _inlet, char * hydrofile, double datenum0)
558 {
559     char buffer[200];
560     int nQ;
561     double time,Q;
562     FILE * fp;
563
564     if(!(fp = fopen (hydrofile, "r")))
565     {
566         printf("Failed to open hydrograph file!\n");
567         return -1;
568     }
569
570     nQ = 0;
571
572     // Read (time,Q) pairs
573     while( fgets(buffer, 200, fp)!= NULL ) // parse until end-of-file
574     {
575         sscanf(buffer,"%lf %lf",&time, &Q);
576         double t_sec = (time-datenum0)*86400.;
577         if(t_sec>=0)
578         {
579             nQ++;
580             _inlet->Time = (double *) realloc( _inlet->Time, nQ*sizeof(double));
581             _inlet->Q = (double *) realloc( _inlet->Q , nQ*sizeof(double));
582             _inlet->Time[nQ-1] = t_sec;
583             _inlet->Q[nQ-1] = Q;
584         }
585     }
586
587     _inlet->nQ = nQ;
588     fclose(fp);
589     printf("%d records loaded from %s\n",nQ,hydrofile);
590
591     return(0);
592 }
593
594 /**
595  *** Linear interpolation in hydrograph
596  */
597
598 double interpolate_discharge(struct Inlet * _inlet, double time)
599 {
600     if(time <= _inlet->Time[0])
601         return(_inlet->Q[0]);
602
603     if(time >= _inlet->Time[( _inlet->nQ)-1])
604         return(_inlet->Q[( _inlet->nQ)-1]);

```

```

602 int pos = 0;
603 while(time > (_inlet->Time[pos]) )
604     pos++;
605
606 double t0 = _inlet->Time[pos-1];
607 double t1 = _inlet->Time[pos];
608 double Q0 = _inlet->Q[pos-1];
609 double Q1 = _inlet->Q[pos];
610
611 return (Q0 + (time-t0)*(Q1-Q0)/(t1-t0));
612 }
613

```

C.4.5 Fonction *manning-tilt.h*

```

1 // Manning coefficient
2 //double n = 0.025;
3 scalar nmanning[]; // n variable spatialement
4
5 // Tilt
6 //double tilt = 0.;
7 struct tilt {
8     double x;
9     double y;
10 };
11 struct tilt tilt = {.x = 0., .y = 0.};
12
13 void updatemanningtilt(scalar * evolving, scalar * sources, double dtmax, int numbersource )
14 {
15     // We first recover the evolving fields
16     scalar h = evolving[0];
17     vector u = { evolving[1], evolving[2] };
18     // double s, h2inv;
19     // double eps =
20
21     // Updates for evolving quantities
22     vector dshu = { sources[1], sources[2] };
23
24     foreach(){
25         if(h[] > dry){
26             // double s = dtmax*n*n*G*norm(u)/pow(h[],4./3.); // sans dimension
27             // FIXME : d singulariser h pour obtenir un ressuyage plus rapide des zones faiblement
28             // submerges ?
29             double s = dtmax*nmanning[]*nmanning[]*G*norm(u)/pow(h[],4./3.); // Equation (5) du
30             // m mo
31             h2inv =
32
33             foreach_dimension()
34                 // Translate it in an explicit form
35                 // dshu.x[] += G*h[]*tilt/(1+s) - h[]*u.x[]*s/(1+s)/dtmax;
36                 dshu.x[] += G*h[]*tilt.x/(1+s) - h[]*u.x[]*s/(1+s)/dtmax; // Equation (6) du m mo
37             }
38         }
39     boundary((scalar *){dshu});
40
41     // Calling of the next source term
42     numbersource++;
43     updatesource[numbersource](evolving, sources, dtmax, numbersource);
44 }
45
46 // Overloading
47 event initmanningtilt(i = 0){
48     updatesource[numbersource]=updatemanningtilt;
49     numbersource++;
50     updatesource[numbersource] = fnull;
51 }

```

