

THÈSE POUR OBTENIR LE GRADE DE DOCTEUR DE L'UNIVERSITÉ DE MONTPELLIER

En Sciences de la Terre et de l'Eau

École doctorale GAIA

Unité de recherche UMR 5243 - Géosciences

Couplage vagues-morphodynamique du littoral par principe de minimisation

Présentée par Ronan DUPONT
le 30 septembre 2024

Sous la direction de Frédéric BOUCHETTE
et Bijan MOHAMMADI

Devant le jury composé de

Marissa YATES
Emma Imen TURKI
Patrick MARCHESIELLO
Catherine CHOQUET
Mehmet ERSOY
Frédéric BOUCHETTE
Bijan MOHAMMADI

Chargée de Recherche, HDR
Maîtresse de conférences, HDR
Directeur de recherche
Professeure
Maître de conférences
Professeur
Professeur

École des Ponts, LHSV
Université de Rouen Normandie
IRD, LEGOS
Université de La Rochelle
Université de Toulon
Université de Montpellier
Université de Montpellier

Rapporteuse
Rapporteuse
Examinateur
Examinateur
Examinateur
Directeur de thèse
Directeur de thèse

Cette thèse a été financée par le CNRS.



RÉSUMÉ



UNIVERSITÉ
DE MONTPELLIER

Titre : Méthode de couplage vague-morphodynamisme du littoral par principe de minimisation.

Mot clés : Hydro-morphodynamisme, Optimisation, Validation de modèle, Littoral, Approche variationnelle, Minimisation d'énergie, Transport Optimal, Ondes.

Résumé : Les modèles morphodynamiques dans les eaux côtières peu profondes sont souvent très complexes, en particulier lorsqu'il s'agit de reproduire des phénomènes physiques tels que la création de barres sédimentaires. Les modèles classiques sont généralement hautement paramétrés; ils résolvent séparément les équations physiques de l'hydrodynamique et de la morphodynamique à une très petite échelle de l'ordre de la seconde dans le temps et du mètre dans l'espace. Durant cette thèse, nous avons développé un modèle numérique proposant une approche plus globale de la morphodynamique côtière, basée sur un principe d'optimisation.

La théorie de l'optimisation est l'étude de l'évolution d'un système en recherchant systématiquement le minimum d'une fonction dérivée de certaines de ses propriétés physiques. En utilisant la théorie de l'optimisation mathématique, nous avons conçu un modèle qui décrit l'évolution de l'élévation du fond marin en tenant compte du couplage entre les processus morphodynamiques et hydrodynamiques. Notre modèle est basé sur l'hypothèse que le fond marin s'adapte pour minimiser l'énergie des vagues. Le choix de cette fonction détermine la force motrice de l'évolution morphologique du fond marin.

Les modèles basés sur le principe de minimisation reposent sur le calcul de certaines dérivées. Ce calcul peut être effectué par des méthodes lourdes (différenciation automatique) ou plus légères (solutions analytiques), mais elles présentent toutes des inconvénients. En utilisant la dérivée à la manière d'Hadamard, nous avons élaboré une stratégie pour calculer le gradient de toute fonction de coût par rapport à la forme, ce qui nous permet de résoudre le problème d'optimisation au cœur du modèle. Cette stratégie nous a permis de créer un modèle morphodynamique générique qui peut être utilisé avec n'importe quel outil hydrodynamique. Ainsi, notre modèle a pu être validé numériquement (convergences, ...) mais également expérimentalement à travers des cas d'expériences en canal.

Grâce à ces développements, le code est opérationnel en 1D et en 2D et est disposé à résoudre des problèmes d'optimisation liés à l'ingénierie côtière, visant à optimiser les positions et les formes d'ouvrages de protection du littoral.

ABSTRACT



UNIVERSITÉ
DE MONTPELLIER

Title: Method of wave-morphodynamic coupling of the coastline by minimization principle.

Keywords: Hydro-morphodynamics, Optimization, Model validation, Coastal, Variational approach, Energy minimization, Optimal transport, Waves.

Abstract: Morphodynamical models in shallow coastal waters is a challenging topic, especially when trying to reproduce physical phenomena such as sandbar creation. Classic models are generally very complex and highly parameterized; they separately solve the physical equations of hydrodynamics and morphodynamics at a very small scale of the order of second in time and meter in space. During this thesis, we developed a numerical model proposing a more global approach to coastal morphodynamics, based on an optimization principle.

The optimization theory is the study of the evolution of a system while searching systematically for the minimum of a function derived from some of its physical properties. Using mathematical optimization theory, we have designed a model that describes the evolution of the sea bottom elevation while taking into account the coupling between morphodynamic and hydrodynamic processes. Our model is based on the assumption that the sea bottom adapts to minimize a wave energy. The choice

of this function determines the driving force behind the morphological evolution of the seabed.

Models based on the minimization principle rely on the calculation of some derivatives. This can be achieved by heavy methods (automatic differentiation) and lighter ones (analytical solutions); but they all have their drawbacks. Our strategy uses the Hadamard derivative to calculate the gradient of any cost function with respect to shape, allowing us to solve the optimization problem at the heart of the model. This strategy has enabled us to create a generic morphodynamic model that can be used with any hydrodynamic tool. Thus, our model has thus been validated both numerically (convergences, etc.) and experimentally, through flume canal experiments.

Thanks to these developments, the code is operational in 1D and 2D and is ready to solve optimization problems linked to coastal engineering, aimed at optimizing the positions and shapes of coastal protection structures.

TABLE OF CONTENTS

Current chapter contents

Résumé	4
Abstract	5
Table of Contents	7
Nomenclature	13
Introduction	17
1 Contexte de la Thèse	17
1.1 Axe 1 : Développements Théoriques	18
1.2 Axe 2 : Développement de l'Outil de Calcul Numérique	18
1.3 Axe 3 : Validations du Modèle	18
2 Objectifs de la Thèse	19
3 Modélisation du Littoral	19
3.1 Modélisation Morphodynamique des Plages	19
3.2 Hypothèse : la Nature Cherche à Minimiser l'Énergie qu'elle Dépense .	20
3.3 Comment Traiter le Problème de Minimisation ?	22
4 Organisation de la Thèse	25
 0 The emergence of morphodynamic modeling by the principle of constrained minimization	27
1 Introduction	28
2 Optimization on Coastal Protection Structures	28
2.1 Optimal Shape of Structures	28
2.2 Port of la Turballe	30
3 Optimization of the Location of Costal Protection Structures	32
3.1 Position of Optimal Protection Structure at the Pointe de l'Espiguette .	32
3.2 Optimal Position of Geotextiles Tubes	33
3.3 Position of Geotextiles Tubes Structure via OptiMorph	33

TABLE OF CONTENTS

4	Towards a Model of Hydro-Morphodynamism by Optimization	34
4.1	2D Hydro-Morphodynamic Models Based on the Shallow-Water Equations	35
4.2	1D Hydro-Morphodynamic Models Based on Energy Minimization	36
5	Conclusion	36
1	Sandy beach dynamics by constrained wave energy minimization	37
1	Introduction	38
1.1	State of the Art	39
1.2	Hypotheses	39
2	Numerical Model	40
2.1	Model forcing	42
2.2	Hydrodynamic Model	42
2.3	Morphodynamic Model by Wave Energy Minimization	45
2.3.1	Link with Morphodynamic Flux-Based Models and Sediment Characteristics Y	47
2.4	Model Constraints	50
3	Numerical Application	51
3.1	Description of the Experiment	51
3.2	XBeach Model	52
3.3	Hydrodynamic Validation	52
3.4	Numerical Results of the Morphodynamic Simulations	53
4	Extension to Multi-1D Simulations	55
4.1	Applications to a Multi-1D Case Near Montpellier	56
5	Discussion	57
5.1	Robustness Analysis of the Consistency in Time and Space of the Morphodynamic model	57
5.2	The Robustness of the Domain Length	59
5.3	Parameter Robustness Analysis	59
5.4	Mid-term Simulations	60
6	Conclusions	63
2	Hydrodynamic models	65
1	Introduction	66
2	Physics of Wave Motion	67
3	Linear Wave Theory	70
3.1	Celerity	72
3.2	Wave Energy	72
3.3	Shoaling Coefficient	73
3.4	Extended Shoaling Model	74
4	Wave Resolving	75
4.1	Context	76

4.2	Derivation of Shallow-Water Model	77
4.3	Shallow-Water Equations with Variable Bottom	78
4.4	REF/DIF Numerical Model	79
5	Spectral Wave Resolution	81
5.1	Context	81
5.2	Mathematical Background	82
5.3	Interesting Quantity	84
5.4	XBeach Numerical Model	85
5.4.1	Hydrodynamics	85
5.5	SWAN Numerical Model	87
5.6	Hydrodynamic Simulations	89
5.6.1	Flume Experiment	89
5.6.2	Open-Sea Simulation	90
6	Conclusion	91
3	Beaches morphodynamic modeling based on Hadamard Sensitivity Analysis	93
1	Introduction	94
2	Gradient Calculation with Respect to the Shape $\nabla_\psi \mathcal{J}$	95
2.1	Analytical Calculation of $\nabla_\psi H$	95
2.2	Finite Difference Calculation of $\nabla_\psi H$	96
2.3	Automatic Differentiation (AD) Method to Calculate $\nabla_\psi H$	97
2.3.1	Direct and Reverse Modes of Automatic Differentiation	97
3	Using Hadamard for the Calculation of $\nabla_\psi \mathcal{J}$	97
3.1	Principle	97
3.2	Analytical Examples of Hadamard Derivative	98
3.2.1	Flat form	98
3.2.2	Linear Form	99
3.3	Numerical Validation	101
3.4	Validating the Hadamard Solution	102
4	Application of Hadamard Strategy	103
4.1	Description of Flumes Experiments	104
4.1.1	The SANDS Experiments	104
4.1.2	The LIP Experiments	105
4.2	Hydro-Morphodynamic Results on Flume Experiment	106
4.3	Hydro-Morphodynamic Results on Open-sea Configurations	108
5	Discussion	110
5.1	Computation time	110
5.2	Flume Simulation	111
5.3	Open-Sea Simulation	113
5.4	Gamma Sensibility	114
5.5	Limitation of a Wave Resolving Model	114

TABLE OF CONTENTS

6 Conclusion	115
4 Extending the OptiMorph model in 2D	117
1 Introduction	118
2 Upgrade OptiMorph Model to 2D	118
2.1 2D Wave Model	118
2.2 2D Morphodynamic Model	119
2.3 2D Hadamard Derivative	119
2.3.1 Mathematical Background	120
2.3.2 Numerical Validation	120
2.4 2D Constraints	121
2.4.1 Slope Constraint	121
2.4.2 Sand Conservation Constraint	122
3 2D Applications	122
3.1 Presentation of the Copter 2D experience	122
3.2 Application on Copter 2D	124
3.3 Linear Seabed with Geotube	124
4 Discussion	126
5 Conclusion	126
5 OptiMorph 2.0 User Guide	129
1 Introduction	131
1.1 About	131
1.2 Expectation and Objectives	131
1.3 Target Audience	131
2 Processes and Theoretical Formulation	132
2.1 Domain and Definitions	132
2.1.1 Tide	132
2.2 Hydrodynamic Models	133
2.2.1 Extended Shoaling Model	133
2.2.2 XBeach Model	135
2.2.3 SWAN Model	137
2.3 Morphodynamic Model by Wave Energy Minimization	138
2.3.1 Introduction	139
2.3.2 Governing Equation of Seabed Dynamics	139
2.3.3 Parameter Y	139
2.3.4 Parameter Λ	140
2.3.5 Direction of Descent d	141
2.3.6 Choice of Cost Function \mathcal{J}	142
2.3.7 Hadamard Derivative to Compute $\nabla_{\psi}\mathcal{J}$	142
2.3.8 Slope Limiter	143

2.4	Model Constraints	145
2.4.1	Slope Constraint	145
2.4.2	Sand Stock Constraint	146
3	Numerical Model	147
3.1	Presentation	147
3.1.1	Workflow	147
3.1.2	Program Organization	148
3.2	Running OptiMorph	151
3.2.1	Installation of OptiMorph	151
3.2.2	Installation of PAGURE (to install SWAN and XBEACH)	152
3.2.3	Input File	161
4	Applications	163
4.1	1D Linear Seabed Beach Configuration using Hadamard approach with SWAN	164
4.1.1	Setting	164
4.1.2	Input files	164
4.1.3	Load SWAN and Run OptiMorph on Cluster	166
4.1.4	Results	166
4.2	1D Linear Seabed Beach with GeoTube using Hadamard approach with XBeach	169
4.2.1	Setting	169
4.2.2	Input Files	169
4.2.3	Load XBeach and Run OptiMorph on Cluster	171
4.2.4	Results	171
4.3	2D Linear Seabed Beach Configuration using Hadamard approach with Shoaling	174
4.3.1	Setting	174
4.3.2	Input Files	174
4.3.3	Results	176
Conclusion et Perspectives		179
1	Conclusion	179
2	Perspectives	180
Appendix		183
A	A Genetic Algorithm to Solve the Optimization Problem (I.1)	183
A.1	Population Creation	183
A.2	Selection	184
A.3	Mutation	184
A.4	Final Population	185
B	A few Attempts to Improve the Model	185

TABLE OF CONTENTS

B.1	Reference Cases	186
B.1.1	Case 1: Simulation of a One-Week Storm on a Linear Beach .	186
B.1.2	Case 2: Simulation of a Flume Experiment: COPTER	187
B.2	Improvement by Functional Approach	188
B.2.1	Functional with Kinetic Energy Dissipation (CF8)	189
B.2.2	Functional in Terms of Representing Work	192
B.2.3	Functional with Radiation Stress S_{xx}	193
B.2.4	Functional with Memory Term	194
B.3	Adding Transport in the Descent Scheme	195
B.4	Conclusion	197
C	Configuration File of XBeach	197
D	Configuration File of SWAN	198
 List of Figures		201
 List of Tables		206
 List of Simulations		207
 Bibliography		211

NOMENCLATURE

Acronyms

Symbol	Description	Units
$\bar{\eta}$	the mean free surface level	(m)
β	Maximum sand slope	(1)
Δx	Spatial step	(m)
η	Free surface level	(m)
η_{RMS}	Root Mean Square of η	(m)
γ	Munk breaking criterion	(1)
\mathcal{J}	Functionnal to minimize	(J s m^{-1})
Λ	Excitation of the seabed by the water waves	(1)
$\mathcal{E}_{\mathcal{H}}$	Wave energy	(J m^{-2})
\mathcal{E}_{L^2}	Error from L^2 space	
ψ	Sea-bottom profil: bathymetry	(m)
ψ_0	Initial bathymetry	(m)
ψ_f	Final bathymetry	(m)
ρ_w	Water density	(kg.m^{-3})
σ	Pulsation	(s^{-1})
θ	Direction	(rad)

Nomenclature

θ_M	Mean wave direction	(rad)
θ_m	Wave direction for the frequency f	(rad)
Y	Sediment mobility	(m s kg $^{-1}$)
A	Wave action	(J s m $^{-2}$)
a	Wave amplitude	(m)
C_{tide}	The tidal coefficient	(1)
D	Local water depth	(m)
d	Direction of the descent	(J s m $^{-2}$)
D_w	Wave energy dissipation	(J m $^{-2}$ s $^{-1}$)
h	Water depth	(m)
H_0	Offshore water depth	(m)
H_s	Significant wave height	(m)
H_{RMS}	Root Mean Square of H	(m)
k	Wavenumber	(m $^{-1}$)
ka, α	Wave slope	(1)
L	Wave length	(m)
m_p	Moment of power spectrum of order p	(m 2 rad $^{-1}$ s $^{-2-p}$)
M_{eff}	The effective tidal range	(m)
M_{ref}	The reference tidal range	(m)
n_{iter}	Number of iterations	(1)
q	Flow, $q = hu$	(m 2 .s $^{-1}$)
T_0	the wave period	(s)
T_{coupl}	Coupling time interval between hydrodynamic and morphodynamic models	(s)
T_{tide}	The tide duration	(h)

u	Current	(m.s^{-1})
$x - \epsilon$	Previous point of the discretization	(m)
H	Wave height	(m)
h	Water depth	(m)

INTRODUCTION

ACTUELLEMENT, environ 30% de la population mondiale vit proche des côtes (Small et al. 2003). Ces zones constituent des écosystèmes complexes, d'une richesse environnementale inestimable. Au regard de la croissance démographique, des activités économiques grandissantes, des nouvelles infrastructures, du changement climatique ; il n'a jamais été aussi important de comprendre et d'anticiper les phénomènes du littoral. Face à cet enjeu, la modélisation joue aujourd'hui un rôle majeur afin de préserver ces zones côtières. Conjointement, les mathématiques et la physique permettent la compréhension des mécanismes du littoral ainsi que l'anticipation de leur impact sur les côtes. Les mathématiques apportent des outils puissants permettant de résoudre numériquement les problèmes physiques, tandis que la physique et l'hydrodynamique régissent les lois et apportent une expertise pour évaluer la vulnérabilité des côtes aux inondations et à l'érosion.

Ainsi, le développement de modèles numériques est un domaine de recherche en perpétuelle évolution. Ces modèles, ayant souvent l'ambition de prévoir au mieux les répercussions côtières, fournissent des outils de prédiction et d'analyse, contribuant de manière significative à une gestion efficace des zones côtières. Ceux-ci permettent de prendre des décisions rationnelles afin d'atténuer les risques liés au changement climatique, de protéger les populations et écosystèmes ainsi que favoriser un développement durable des régions côtières.

1 Contexte de la Thèse

Cette thèse s'intègre à la suite de nombreux travaux issus du laboratoire de Geosciences et l'institut Montpelliérain Alexander Grothendieck de Montpellier. FRÉDÉRIC BOUCHETTE, BIJAN MOHAMMADI et PASCAL AZERAD ont été à la tête de ces travaux. Ils ont commencé en 2004 en partenariat avec l'entreprise BRLi en travaillant sur la formulation d'une théorie de l'optimisation adaptée au domaine de l'hydromorphodynamisme littoral, des développements numériques basés sur cette théorie et conçus pour inven-

ter des systèmes optimaux de protection du littoral contre la submersion, l'érosion littorale, l'impact des vagues contre les ouvrages, etc. Ces travaux ont vécu à travers 4 thèses : DAMIEN ISÈBE, AFAF BOUHARGUANE, MEGAN COOK et RONAN DUPONT. Ils s'achèveront donc avec cette thèse. Initialement, les travaux se concentraient sur l'optimisation de structures de protections du littoral puis finalement le développement de méthodes de minimisations appliquées à la morphodynamique du littoral. L'historique de ces travaux a été retracé dans l'état de l'art [0](#). Cette thèse vient à la suite de celle de [Cook \(2021\)](#). Le financement de cette thèse provient du CNRS, dans le cadre des Mission pour les initiatives transverses et interdisciplinaires (MITI - www.miti.cnrs.fr). Cette thèse n'a pas l'obligation de traiter des problèmes d'ingénierie, elle a la possibilité d'être un projet exploratoire purement académique. Initialement, les travaux ont été découplés en 3 axes comme explicités ci-dessous.

1.1 Axe 1 : Développements Théoriques

Le premier axe consiste à reformuler les principes physiques de l'hydro-morphodynamisme du littoral sous la forme d'un problème optimal général. Cette nouvelle approche s'appuiera sur un nombre très réduit de paramètres, refondant complètement la manière de penser le transport sédimentaire. Les paramètres ne devront plus référer aux paramètres du transport sédimentaire classique comme le nombre de Shield ([Nielsen 2002](#)). Le modèle ne devra pas s'appuyer sur la paramétrisation des équations de transports issues des approches [Bagnold \(1966\)](#). L'ambition des développements mathématiques est d'étendre le formalisme afin qu'il soit générique: il devra être utilisé très facilement avec n'importe quel modèle hydrodynamique.

1.2 Axe 2 : Développement de l'Outil de Calcul Numérique

Le second axe consiste à développer un outil de calcul numérique implémentant cette nouvelle approche théorique, dans un contexte bien circonscrit, limité, compatible avec la durée du projet et un travail de doctorat. Les développements théoriques de l'axe 1 devront être implémentés dans un code unique. Le modèle morphodynamique doit être capable de se coupler avec différents types de modèles hydrodynamiques.

1.3 Axe 3 : Validations du Modèle

Le troisième axe consistera à valider les développements théoriques de cet outil de modélisation. L'outil sera confronté à la réalité de terrain, dans une démarche typiquement physique. Sur ce point, le projet bénéficie dès son démarrage de la base de données issue de nombreuses expérimentations en milieu contrôlé ou *in situ* déjà réalisées et validées par

le partenaire GLADYS (www.gladys-littoral.org). En effet, l'institut des plages GLADYS a la vocation régionale/nationale de capitaliser et partager des campagnes de mesure littorale à haute valeur académique (donnée très bien contrainte et validée).

2 Objectifs de la Thèse

Les travaux de [Cook \(2021\)](#) ont donné naissance à une première version du modèle OptiMorph utilisant le principe de minimisation appliqué à la modélisation morphodynamique du littoral. Il convient de poursuivre les développements afin d'étendre les validations numériques / expérimentales. Il est très important que les développements mathématiques et expérimentaux soient mis en avant à travers des publications scientifiques. En effet, cette nouvelle méthode de modélisation décrite comme "originale", soulève souvent quelques réticences dans la communauté du littoral. Il est donc très important d'apporter un maximum d'éléments justifiant nos hypothèses ainsi que la pertinence de notre modèle. Un grand travail de médiation est à prévoir à travers des communications, publications, enseignements, etc.

3 Modélisation du Littoral

3.1 Modélisation Morphodynamique des Plages

Durant cette thèse, nous nous sommes principalement concentrés sur la modélisation morphodynamique des plages. Elle caractérise l'évolution de la morphologie d'une plage, à savoir l'évolution de son fond marin. Il existe de nombreuses méthodes permettant de calculer la morphodynamique. Ces modèles englobent les modèles empiriques ainsi que les modèles basés sur le processus. Ces derniers peuvent être classés en plusieurs catégories, telle que i) les modèles d'évolution des profils ([Larson et al. 1989; Larson et al. 1990; Nairn et al. 1993](#)), qui utilisent uniquement le transport transversal, ii) les modèles basés sur des règles ([Storms et al. 2002; McCarroll et al. 2021](#)), fondés sur un certain nombre de règles telles que la règle de [Bruun \(1954\)](#), iii) les modèles morphologiques 2D ([Fleming et al. 1977; Latteux 1980; Coeffe et al. 1982; Yamaguchi et al. 1985; Watanabe et al. 1986; Maruyama et al. 1988; Wang et al. 1993; Johnson et al. 1995; Nicholson et al. 1997; D. J. Roelvink et al. 2009](#)), qui utilisent des équations de vagues et de courants moyennées en profondeur pour modéliser le transport des sédiments tout en négligeant les variations verticales des paramètres dérivés des vagues, ainsi que iv) les modèles 3D et quasi-3D ([J. A. Roelvink et al. 1994; Lesser et al. 2004; D. J. Roelvink et al. 1995; Briand et al. 1993; Zyberman et al. 2002; Ding et al. 2006; Droenen et al. 2007](#)), qui

déterminent l'évolution des sédiments en utilisant les variations horizontales et verticales des paramètres dérivés des vagues.

Le modèle OptiMorph ([Cook 2021](#); [Ronan Dupont et al. 2022](#); [Ronan Dupont et al. 2023](#); [R. Dupont et al. 2024](#)) est basé sur le contrôle optimal. Cette méthode d'optimisation appliquée au littoral est née après de nombreux travaux sur le littoral. Les premières idées de F. BOUCHETTE, B. MOHAMMADI & AL étaient de travailler directement sur l'optimisation de structures de protection du littoral ([Isèbe et al. 2008b](#)). Ensuite, des travaux se sont concentrés sur la position optimale de structures de protection du littoral comme les Géotubes ([Isèbe et al. 2008b](#); [Cook et al. 2021b](#)).

C'est enfin en 2011 que ces applications d'optimisation sur le littoral ont pris une dimension différente. La question qui s'est posée est la suivante : “*Et si on ne cherchait pas cette fois-ci la forme ou l'emplacement optimal d'une structure, mais on cherchait la forme optimale de la plage*” et c'est à partir de là que le concept de morphodynamique des plages par calcul optimal est né. Cette question appelait directement d'autres questions physiquement plus compliquées. En effet, il est intéressant d'optimiser la forme d'une plage, mais ceci nécessite de l'optimiser selon un critère bien particulier que l'on appellera fonction de coût \mathcal{J} . Il a donc été important de déterminer des hypothèses afin de pouvoir optimiser celle-ci. La dernière hypothèse a été d'admettre que “*La nature cherche à minimiser l'énergie qu'elle dépense*” et donc cela reviendrait à dire que la plage cherche à évoluer de telle sorte que l'énergie des vagues soit la minimale. Des explications sur cette hypothèse sont apportées dans la section suivante [3.2](#). C'est sur cette citation que les développements de [Cook \(2021\)](#) ont été effectués pour donner naissance à Optimorph, un code de calcul morphodynamique 1D reposant sur la théorie du transport optimal et nécessitant un nombre très réduit de paramètres physiques. Des premiers travaux de validation ont été effectués notamment en confrontant ce modèle à d'autres modèles très classiques comme XBeach ([Cook 2021](#)). La suite de ces travaux de développement et validation sont présentés dans cette thèse.

3.2 Hypothèse : la Nature Cherche à Minimiser l'Énergie qu'elle Dépense

La vérification de l'hypothèse que la nature cherche à minimiser l'énergie qu'elle dépense peut se faire sur des cas très simples. Par exemple, il est possible d'observer l'évolution de l'énergie des vagues $\mathcal{E}_H = \frac{1}{16}\rho_w g \int_{\Omega} H^2 d\Omega$ sur différents types de fonds marins : des fonds expérimentaux, des fonds provenant de mesures en pleine mer, d'autres s'inspirant de profils de plages à l'équilibre ([Bruun 1954](#)), ... Le but de ces observations vise à montrer que l'énergie des vagues est de plus en plus faible au fil du temps.

Il est très difficile d'obtenir des mesures morphodynamiques et hydrodynamiques complètes. La plupart du temps, les jeux de données morphodynamiques sont complets, mais l'hydrodynamique ne contient que quelques points de mesure : c'est le cas de l'expérience

en canal LIP 1C (Roelvink et al. 1995). Pour ce cas-là, il convient donc de simuler la propagation de vagues avec des modèles comme XBeach (utilisé uniquement en modèle de vagues) (D. J. Roelvink et al. 2010; Daly 2009; Bugajny et al. 2013; Williams et al. 2015) afin de compléter artificiellement la donnée. Avec l'avancée des technologies, la technologie LIDAR permet d'obtenir des jeux de données complets : sur un tronçon entier. C'est le cas de l'expérience DynaRev (Blenkinsopp et al. 2021; Schimmels et al. 2020; Martins et al. 2020).

Nous commençons donc par utiliser les données expérimentales très reconnues LIP11D - 1C (Roelvink et al. 1995) où on observe une barre sédimentaire se déplacer au fil de l'expérience. Ces données ont été cruciales dans la validation du modèle hydro-morphodynamique XBeach. Les conditions de cette expérience sont les suivantes : une période de vague $T_0 = 5\text{ s}$, un forçage de vague de $H_s = 1.4\text{ m}$ et une durée totale de $T_f = 13\text{ h}$. Deux simulations hydrodynamiques seront effectuées, une sur la bathymétrie initiale ψ_0 (correspondant à $t = 0\text{ h}$) et une sur la bathymétrie finale ψ_f (correspondant à $t = 12\text{ h}$). Les résultats de ces simulations ainsi que les calculs de \mathcal{E}_H sont présentés sur la figure I.1.1.

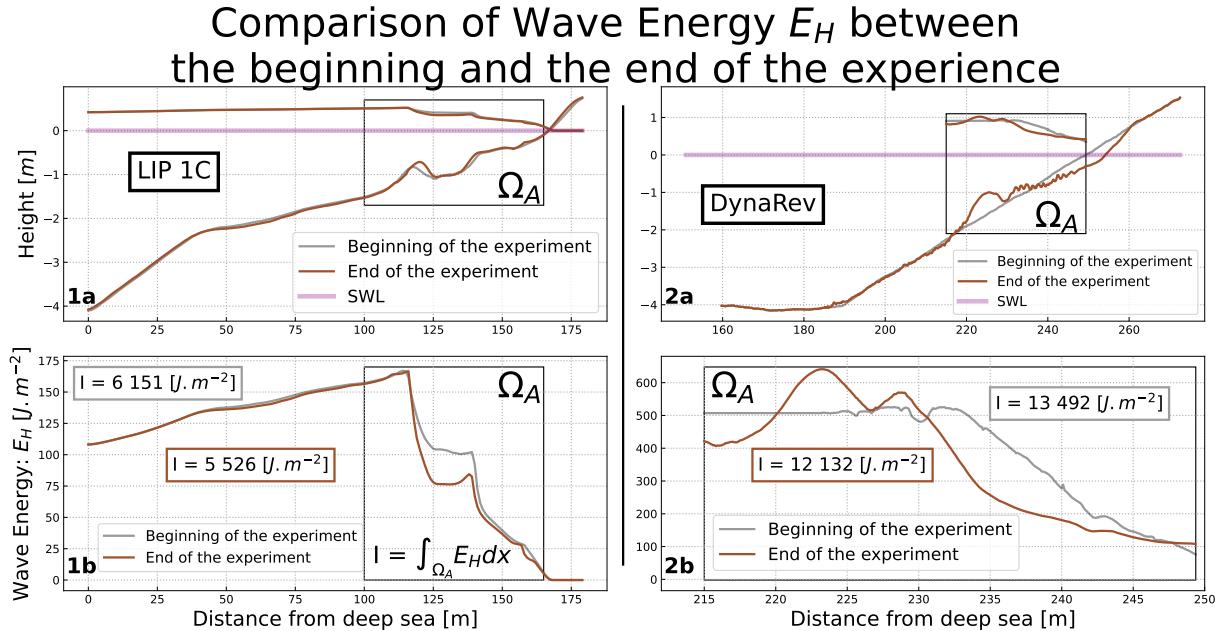


Figure I.1 – 1) Expérience LIP 1C avec H généré par XBeach. 2) Expérience DynaRev avec H mesuré par LIDAR. a) Fond marin et Hauteur d'eau moyennée au début de l'expérience (gris), Fond marin et Hauteur d'eau moyennée à la fin de l'expérience (marron). b) Énergies des vagues associées aux hauteurs d'eaux. L'énergie est calculée sur le rectangle gris.

Ici, d'après la figure I.1.1b, il est évident que l'énergie des vagues à la fin de la simulation $\mathcal{E}_H(t = 12\text{ h})$ est plus faible que l'énergie au début $\mathcal{E}_H(t = 0\text{ h})$. Sur le domaine sélectionné Ω_A (figure I.1.1), celle-ci est 10% plus faible que l'énergie initiale (ie, $\mathcal{E}_H(t = 12\text{ h}) = 0.90 \mathcal{E}_H(t = 0\text{ h})$). L'hypothèse semble valable pour cette expérimentation.

tion.

Une autre vérification possible est d'utiliser les données LIDAR DynaRev (Blenkinsopp et al. 2021; Schimmels et al. 2020; Martins et al. 2020). Dans cette expérience, le bathymétrie est initialement linéaire puis une barre se crée au fil du temps (en 20h d'expérimentation). Cette expérience est très réaliste, car les barres sédimentaires sont très souvent observées dans la nature (Wright et al. 1984). Nous prendrons donc la bathymétrie initiale de l'expérience ψ_0 ($t = 0$ h) et la bathymétrie finale ψ_f ($t = 20$ h) ainsi que les hydrodynamiques associées (Martins et al. 2020). Toutes les données sont expérimentales. Celles-ci sont présentées sur la figure I.1.2.

De même, d'après la figure I.1.2b, il est évident que l'énergie des vagues à la fin de l'expérience $\mathcal{E}_H(t = 20\text{ h})$ est plus faible que l'énergie au début $\mathcal{E}_H(t = 0\text{ h})$. Sur le domaine sélectionné Ω_A (figure I.1.1), celle-ci est 10% plus faible que l'énergie initiale (ie, $\mathcal{E}_H(t = 12\text{ h}) = 0.90 \mathcal{E}_H(t = 0)$). L'hypothèse semble valable pour cette expérimentation également.

3.3 Comment Traiter le Problème de Minimisation ?

Une fois l'hypothèse de la nature qui semble minimiser l'énergie qu'elle dépense vérifiée, il convient de traduire cela par le problème d'optimisation suivant. Il faut chercher ψ , le fond marin de telle sorte que la fonctionnelle $\mathcal{J} = \mathcal{E}_H$ soit la plus faible possible. Ceci ce traduit mathématiquement par l'équation (I.1) suivante.

$$\min_{\psi} \mathcal{J} \quad \text{avec} \quad \mathcal{J} = \frac{1}{16} \rho_w g \int_{\Omega} H^2 d\Omega. \quad (\text{I.1})$$

En l'état, une solution de (I.1) serait d'avoir un mur à l'entrée du domaine, qui empêcherait toute propagation de vague possible (sans eau, la fonction de coût \mathcal{J} est nulle). Certaines contraintes physiques doivent donc être ajoutées. Une pente maximale de notre plage afin de ne pas avoir de pentes aberrantes ; une contrainte de conservation sableuse qui pourra être activée dans un domaine fermé ainsi qu'une contrainte d'excitation sableuse qui augmentera ou limitera la mobilité du sable en fonction de la profondeur de celui-ci ainsi que les conditions de houle (par exemple, il est impossible d'avoir un mouvement sédimentaire à 1000 m de profondeur dû aux vagues à la surface). Les définitions de ces contraintes physiques seront revues dans le chapitre 1.

Dans la littérature, il existe une multitude de méthodes d'optimisation (Mohammadi et al. 2009; D. J. Munk et al. 2015) pour traiter le problème de minimisation (I.1). Il existe deux grandes familles de méthodes : celles déterministes, basées sur le gradient (descente du gradient, Newton, ...) et celles stochastiques (algorithmes évolutionnistes, ...).

Naïvement, un choix pourrait être de résoudre le problème (I.1) par une méthode stochastique basée sur l'évolution des populations. Ici, la population $\psi_{i=0,\dots,N_{pop}}$ est une population de fonds marins satisfaisant les contraintes de pente et de conservation sableuse, afin d'obtenir le résultat le plus réaliste possible. La méthode choisie fait partie des plus renommées : l'utilisation d'un algorithme génétique basé sur des duels ([Biyanto et al. 2016](#)). À chaque génération, un duel a lieu entre deux individus de la population, le perdant, ayant la fonction de coût \mathcal{J} la plus forte, subira une mutation. Cette méthode d'optimisation est illustrée sur le workflow figure I.2. Plus d'explications sur les étapes sont données dans l'annexe A.

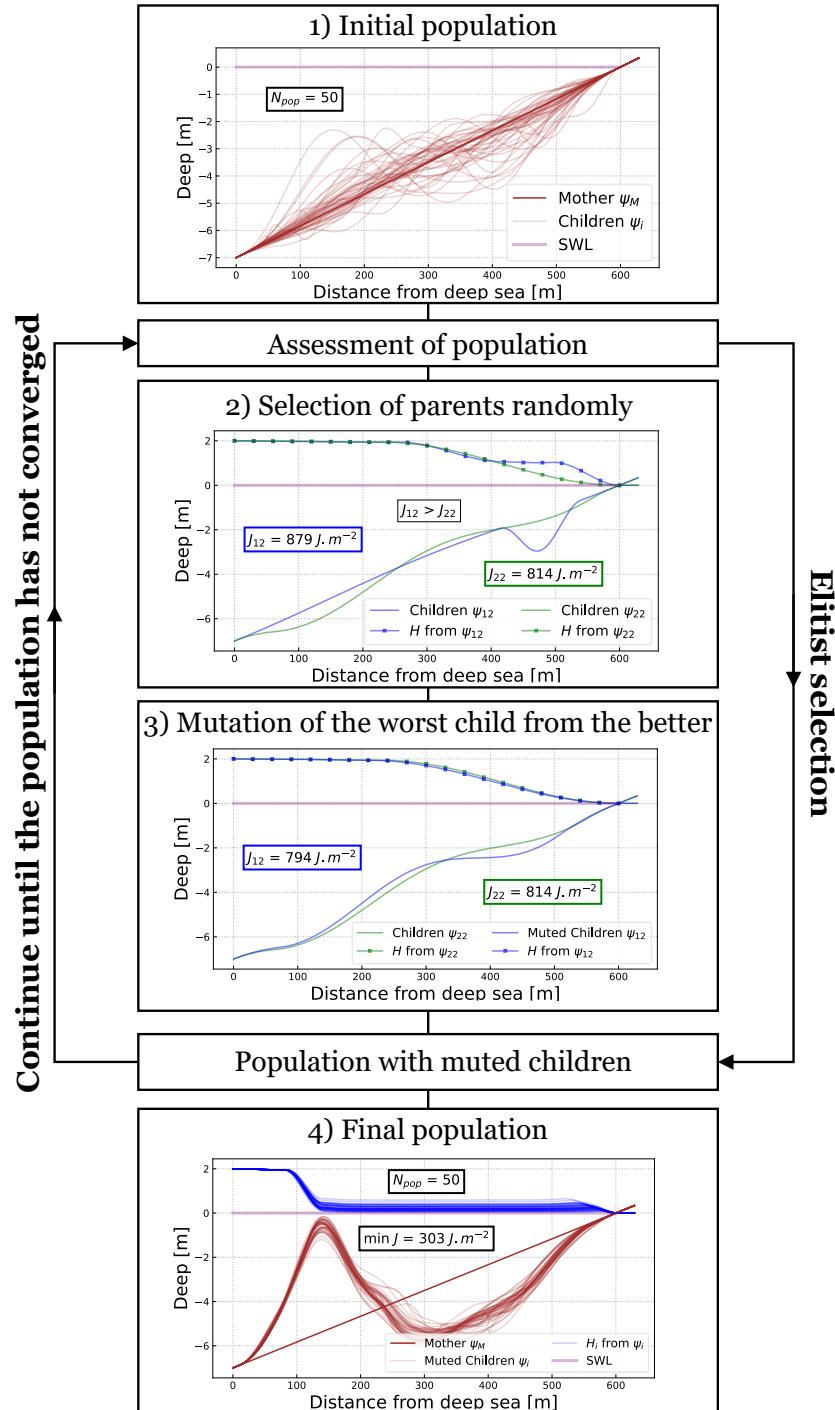


Figure I.2 – Workflow d'un algorithme génétique basé sur les duels pour résoudre le problème de minimisation (I.1). Plus d'explications dans l'annexe A.

Les résultats de l'optimisation (figure I.2.4) montrent que le problème (I.1) a bien été résolu en satisfaisant les contraintes de pente et de conservation sableuse. En effet, le fond

marin va chercher à se rapprocher le plus vite possible de la surface tout en satisfaisant la contrainte de pente ainsi que le point fixe au bord du domaine ($x = 0$ m). On peut de plus observer dans l'annexe la figure A.2 montrant que la fonction de coût \mathcal{J} baisse de génération en génération. Cette résolution du problème de minimisation (I.1), bien qu'elle soit réussite, ne satisfait pas nos exigences de modélisation morphodynamique. En effet, le profil proposé I.2.4) n'est pas très réaliste bien qu'il puisse nous faire penser à un banc de sable. De plus, le problème (I.1) sans contrainte pourrait se résoudre en proposant la solution d'un fond marin supérieure au niveau de l'eau dès $x = 0$. Ceci empêcherait toute vague de se former et l'énergie des vagues serait donc nulle ($\mathcal{J} = 0$).

Il est donc nécessaire de préciser que nous ne cherchons pas la solution absolue du problème de minimisation (I.1). En effet, nous voulons un modèle morphodynamique qui puisse décrire l'évolution du fond marin en fonction du temps. Il convient donc de privilégier les méthodes d'optimisation basées sur le calcul du gradient (Mohammadi et al. 2009) qui vont nous permettre de successivement déterminer l'évolution du fond marin. Le problème d'optimisation (I.1) est donc à considérer au sens du transport optimal. À chaque itération, le fond marin se dirigera vers la solution optimale de ce problème. En faisant tendre le nombre d'itérations à l'infini, la solution trouvée serait celle proposée en figure I.2.4). L'avantage de cette méthode est qu'elle permet l'introduction du temps, et donc, la variation des forçages à chaque instant : ce qui nous rapproche d'un modèle réaliste.

4 Organisation de la Thèse

Le manuscrit commence par un état de l'art 0 qui retrace les travaux initiés en 2007 par F. BOUCHETTE et B. MOHAMMADI. La description des travaux commence par des problématiques d'optimisation de formes ou de position de structures de défense côtière. La finalité des travaux concerne la modélisation morphodynamique par optimisation. C'est sur la suite de ces travaux que la thèse commence.

Le chapitre 1 va se concentrer sur la nouvelle approche pour décrire la morphodynamique côtière, basée sur la théorie de l'optimisation, et plus spécifiquement sur l'hypothèse qu'un profil de plage sableuse évolue afin de minimiser une fonctionnelle liée aux vagues, dont le choix dépend de ce qui est considéré comme la force motrice derrière les processus morphodynamiques côtiers considérés. Une validation numérique et expérimentale sera présentée ainsi qu'une extension du modèle en dimension supérieure.

Le chapitre 2 commencera par les équations fondatrices des mouvements fluides pour arriver à la théorie linéaire ainsi qu'aux deux grandes familles de modèles hydrodynamiques : les modèles à phase résolue et les modèles spectraux. Ceci permet d'introduire les modèles hydrodynamiques utilisés dans OptiMorph dans le chapitre 3. Certains sont

à résolution spectrale comme XBeach ou SWAN, mais aussi à phase résolue comme le modèle Shallow-Water ou REF/DIF. Nous présenterons également un modèle hydrodynamique à faible complexité. Ces modèles sont comparés sur le cas expérimental LIP 1C.

Dans le chapitre 3, nous présenterons la stratégie d’Hadamard pour rendre notre modèle générique. Grâce à ces avancées, le modèle morphodynamique pourra être couplé à n’importe quel modèle de vagues. Nous utiliserons les modèles de vagues de SWAN, XBeach et Shallow-Water dans notre modèle, et nous comparerons les résultats morphodynamiques aux benchmarks hydro-morphodynamique LIP 1C et SANDS ainsi qu’à des simulations en pleine mer.

Dans le chapitre 4, nous étendons l’approche du chapitre 3 à la dimension 2D. Nous effectuerons les développements mathématiques du formalisme en 2D et regarderons la validité de ce formalisme. Nous utiliserons notre modèle morphodynamique couplé avec les modèles REF/DIF et Shoaling en multi-1D. Nous effectuerons une validation morphodynamique sur l’expérience Copter en 2D.

Le chapitre 5 est un guide explicatif de la deuxième version du modèle numérique OptiMorph. La nouvelle version du modèle OptiMorph basé sur l’hydro-morphodynamique côtière par principe de minimisation sera présentée. Des exemples et explications seront apportés afin d’utiliser ce modèle et de le coupler avec n’importe quels modèles de vagues (on pourra facilement prendre XBeach et SWAN).

STATE OF THE ART

The emergence of morphodynamic modeling by the principle of constrained minimization

In this state of the art, we follow the work developed by B. MOHAMMADI, F. BOUCHETTE and P. AZERAD, as part of a collaboration between Montpellier's geosciences and mathematics departments. This work ranges from optimising the shape or position of coastal defence structures to morphodynamic modelling through optimisation.

Current chapter contents

1	Introduction	28
2	Optimization on Coastal Protection Structures	28
2.1	Optimal Shape of Structures	28
2.2	Port of la Turballe	30
3	Optimization of the Location of Costal Protection Structures	32
3.1	Position of Optimal Protection Structure at the Pointe de l'Espiguette	32
3.2	Optimal Position of Geotextiles Tubes	33
3.3	Position of Geotextiles Tubes Structure via OptiMorph	33
4	Towards a Model of Hydro-Morphodynamism by Optimization	34
4.1	2D Hydro-Morphodynamic Models Based on the Shallow-Water Equations	35
4.2	1D Hydro-Morphodynamic Models Based on Energy Minimization	36
5	Conclusion	36

1 \mathcal{J} ntroduction

LONG TIME before the advent of morphodynamic modelling by minimization, a great deal of optimization work applied to the coast was carried out between the Geo-Sciences laboratory and the Alexander Grothendieck Institute in Montpellier. FRÉDÉRIC BOUCHETTE, BIJAN MOHAMMADI and PASCAL AZERAD have initiated these works. They began in 2004 in partnership with the company BRLi, working on the formulation of an optimization theory adapted to the field of coastal hydromorphodynamics, and on numerical developments based on this theory and designed to invent optimal systems for coastal protection against submersion, coastal erosion, the impact of waves against structures, and so on. This shape-optimization work lived through the two theses of DAMIEN ISÈBE and AFAF BOUHARGUANE. They developed numerous optimization models and applied this work to places like Sète in France. In this state of the art, we follow all the coastal optimization concepts that gave rise to the idea of using this in morphodynamic modeling. This concept will be at the heart of the thesis.

2 \mathcal{O} ptimization on Coastal Protection Structures

The first part of this state of the art will show the works that preceded the evolution of beach morphodynamics. At first, these works were mainly concerned with shoreline protection structures such as geotextiles tubes.

2.1 Optimal Shape of Structures

This works of optimization of the shape of coastal defense structures focused at first on structures allowing breaking the wave agitation on a particular domain (Isèbe et al. 2008b). The hydrodynamics on the domain is modeled by Helmholtz (1868) wave equations which assumes a flat bottom, unlike Berkhoff (1972) model. This model is very suitable in deep water. Then, it is necessary to evaluate the cost function \mathcal{J} which is the energy norm L^2 of water waves free surface elevation $\eta(x, t)$ in an admissible domain. Then an optimization method is used to obtain the optimal form. In this case, the cost function is computed so that the water height is the minimum on the whole domain. With this criterion, the optimal shape obtained resembles the following figure 0.1:

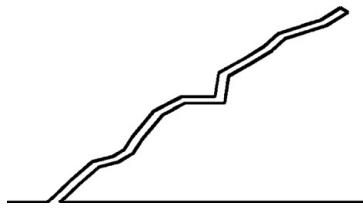


Figure 0.1 – Optimal shape minimizing a cost function \mathcal{J} for a given parameterization

In order to be sure of the viability of this structure, it is convenient to represent the water height η in the 3 configurations of the following figure **0.2**:

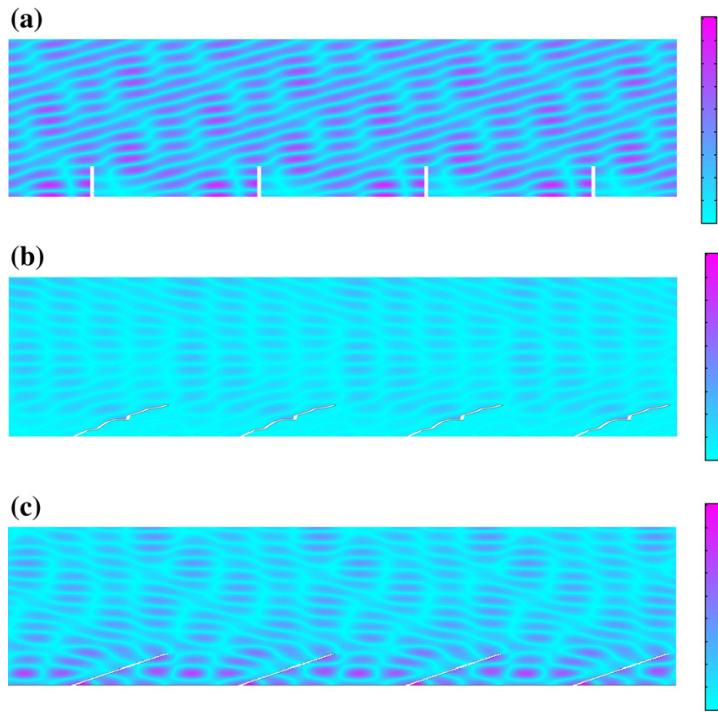


Figure 0.2 – Absolute value of ξ resulting from reflection (a) on rectangular structures perpendicular to the wall, (b) on optimized structures without feasibility constraints, (c) on structures with the same angle as the optimized structures but straight (NW incoming waves with $T=2$ s and $a=0.5$ m).

In the optimal structure configuration **0.2.(b)**, the average η water height is significantly lower than the other configurations. It is also very surprising to see that in a case very similar to the optimal structure **0.2.(c)**, the results seem to be very far from the optimal results.

With the knowledge of the current literature, it was appropriate to reject the shape of the structure figure **0.1**.

2.2 Port of la Turballe

One of the major port development projects of the coming years is the development of the port of La Turballe. For the dimensioning of this work, BRLi, a consulting firm specialized in the fields related to water. They have hired [Cook et al. \(2021c\)](#) in order to confirm their idea on the optimal configuration of the harbor. The purpose of this study is to accommodate more boaters and reduce the agitation inside the harbor. The harbor extensions shown in the following figure [0.3](#) were proposed.

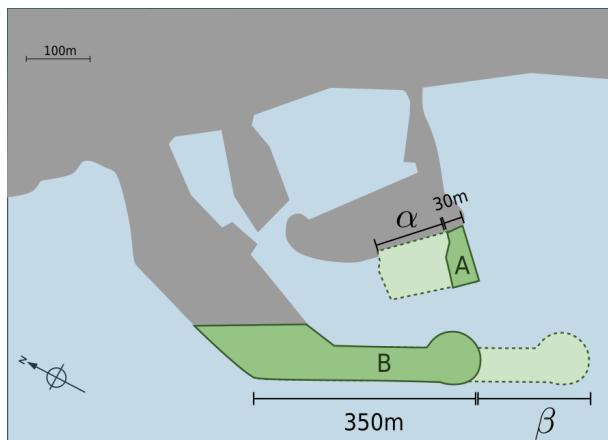


Figure 0.3 – Possible configurations for the port of La Turballe

In this figure, there are two parameters to manage, namely the lengths of the structures A and B . To manage this optimization, it is necessary to model the hydrodynamics of the harbor. To do this, the [Helmholtz \(1868\)](#) model has been solved. Then, it was necessary to create a cost function \mathcal{J} minimizing the agitation in the port. This one was elaborated in the following way in the equation [\(0.1\)](#).

$$\mathcal{J}_n(\psi) = \frac{1}{K(\mathcal{P})} \frac{1}{|\Omega(\psi)|} \int_{\Omega(\psi)} \mathcal{E}_n(\psi, \mathbf{x}) \mathcal{P}(\mathbf{x}) d\mathbf{x} \quad (0.1)$$

The quantity $\mathcal{E}_n(\psi, \mathbf{x})$ is the total surface energy denoted on the domain, associated with the forcing scenario n and the harbor configuration. The function $\mathcal{P}(\mathbf{x})$ is named spatial weight function, which allows prioritizing the minimization of the agitation on some privileged areas of the harbor as can be seen on the figure below [0.4](#):

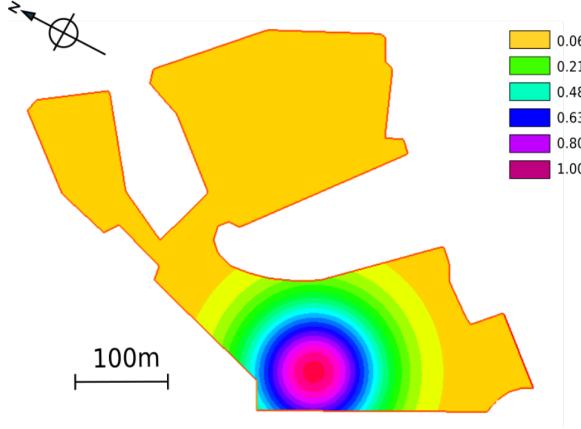


Figure 0.4 – Weight function \mathcal{P} on Ω .

The functions K and $|\Omega(\psi)|$ are equal to $K(\mathcal{P}, \Omega(\psi)) = \int_{\Omega(\psi)} \mathcal{P}(\mathbf{x}) d\mathbf{x}$ and the total area of the domain: this allows scaling the functional.

The functional cumulating the n scenarios is then computed in the following equation (0.2):

$$\mathcal{J}(\psi) = \frac{\sum_{n=1}^N a_i(n) \mathcal{J}_n(\psi)}{\sum_{n=1}^N a_i(n)} \quad (0.2)$$

with the $a_i(n)$ corresponding to the weights of the given n scenario. Once this parameterization is done, a grid of $(\alpha, \beta) \in [0, 150] \times [0, 200]$ is created in order to compute all the values of the functional for all these configurations. Thus, the optimal configuration is the configuration where the couple (α, β) give the lowest functional \mathcal{J} . This solution presents on figure 0.5:

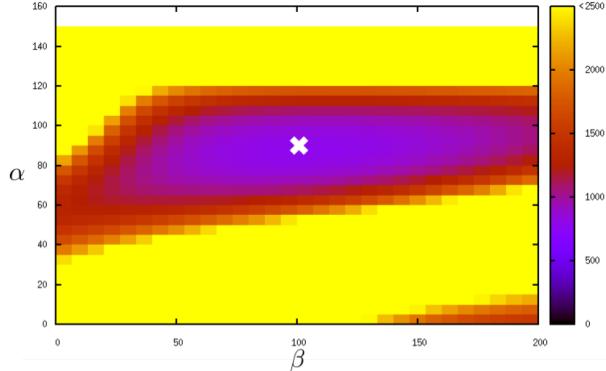


Figure 0.5 – Optimal pair of (α, β) for the configuration of the port of La Turballe.

shows us that there is a unique solution couple in the center of the domain. This work was able to confirm the opinion that BRLi had on the strategic choices of sizing the port. The solution was therefore chosen by BRLi for the completion of this project.

3 Optimization of the Location of Costal Protection Structures

A new approach in terms of optimization of costal protection structures is to look for the optimal position of a costal protection structure. Most of the time, we try to limit the agitation of the waves, beach erosion, etc...

3.1 Position of Optimal Protection Structure at the Pointe de l'Espiguette

The works of [Isèbe et al. \(2008b\)](#), [Isèbe et al. \(2014\)](#) allowed to find the optimal position of a coastal defense structure which allows limiting the erosion of the beach of Le Grau du Roi-Le Boucanet.

Using a numerical model similar to the one of the part [2.1](#), based on the resolution of [Helmholtz \(1868\)](#) equations, the optimal configuration is searched so that the cost function \mathcal{J} limits the erosion of the beach on a domain D .

Before defining the cost function, it is necessary to recall that observations of erosion by oceanographers show that waves can be roughly classified into two categories according to their height H , below or above a critical value H_{lim} . In principle, waves higher than H_{lim} , mainly present during storms, are erosive. They generate a great mechanical energy. On the other hand, when $H < H_{lim}$, the waves favor the reconstruction of eroded beaches. The first class of waves ($H > H_{lim}$) is called erosive and the second-class constructive ($H < H_{lim}$). The cost function \mathcal{J}_θ is defined according to the direction of the wave θ taking into consideration this limit height as follows:

$$\mathcal{J}_\theta = \frac{\int_D E_{H>H_{lim}} dS}{\int_D E_{H<H_{lim}} dS} + \left(\|U_{orb}\| - \|U_{orb}^{\text{initial}}\| \right)_+ + \left(\int_D E_{H<H_{lim}} dS - \int_D E_{H<H_{lim}^{\text{initial}}} dS \right)_+ \quad (0.3)$$

with $(x)_+ = \max(x, 0)$, E the wave energy, H the water height, H_{lim} the limiting water height, U_{orb} the orbital speed. This functional has therefore been defined in such a way that it gives a significant indication of the level of beach erosion.

In order to have a realistic estimate, \mathcal{J} is computed as the sum of the cost functions taking into account in p_θ the probability that a wave of direction θ arrives.

$$\mathcal{J} = \sum p_\theta J_\theta \quad (0.4)$$

Once this cost function has been evaluated, a large number of simulations are performed on the different possible positions of the structure in order to obtain the position corresponding to the minimum \mathcal{J} . The results of the optimal configuration are represented on the following Figures [0.6](#) and [0.7](#):

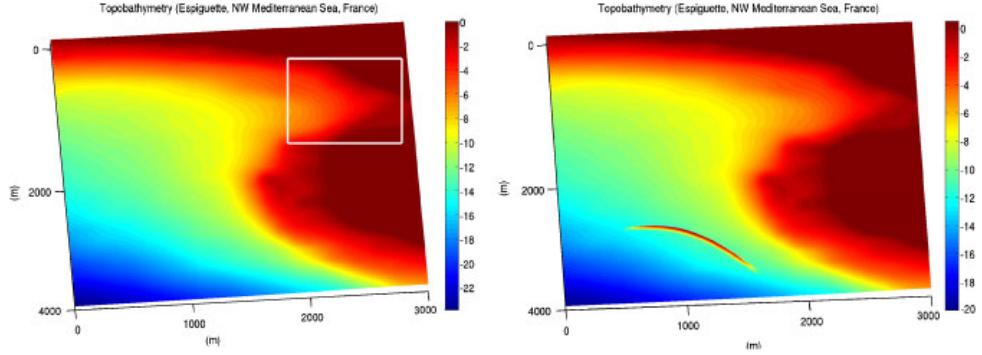


Figure 0.6 – Left: the initial bathymetry of region D for the cost function calculation; right: the modified bathymetry with the optimized protection structure.

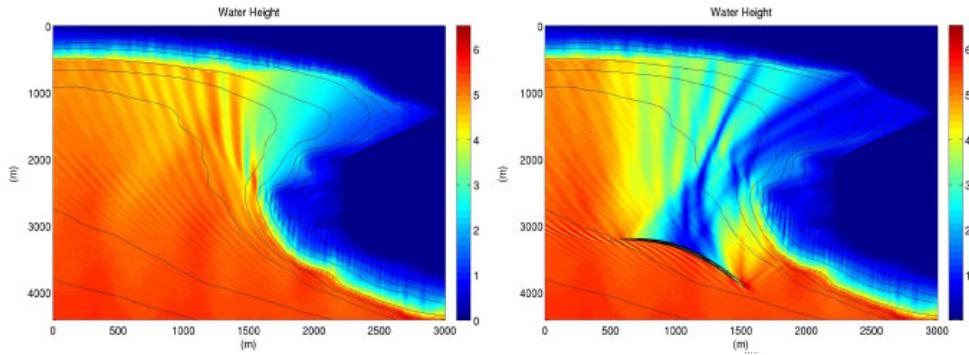


Figure 0.7 – The water height H in the whole domain: (left) for the initial configuration, (right) for the optimized configuration.

The results below clearly show that the protection system attenuates the swell along the coast. Indeed, a large area behind the structure is found with a very low water height H .

3.2 Optimal Position of Geotextiles Tubes

The work of [Isèbe et al. \(2008b\)](#) can be applied to any types of costal protections structures. However, these works were mainly focused on geotextiles tubes. This work inspired [Cook et al. \(2021b\)](#) in his optimization test cases with the new hydro-morphodynamic modeling approach. The details of this approach will be presented in the section [4.2](#).

3.3 Position of Geotextiles Tubes Structure via OptiMorph

One of the applications of [Cook et al. \(2021b\)](#) compares several geotextiles tubes configurations on a linear beach to find the ideal configuration that limits sand displacement. The simulation aims to recreate the real conditions of a 20-day storm with a significant wave height $H = 2$ m and short period $T_0 = 2$ s. The simulation configuration is as follows figure [0.8](#):

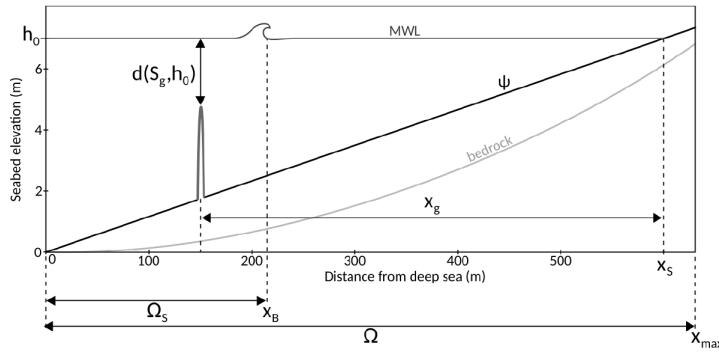


Figure 0.8 – Configuration of the geotube simulation.

Several types of \mathcal{J} cost functions were then evaluated to limit physical phenomena such as beach erosion. For example, the results shown in figure 0.9 below:

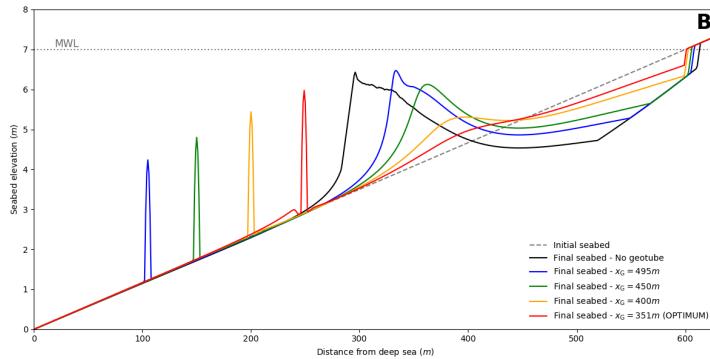


Figure 0.9 – Final seafloor profiles produced by the Opti-Morph morphodynamic model with variable geotube positions on the cross-shore profile.

show us that the optimal position of a geotextiles tubes is 351 m with $\mathcal{J} = \int_{\Omega} (\psi_f - \psi_i) d\Omega$: the sand displacement. Other strategic choices of \mathcal{J} cost functions were made in the article (Cook et al. 2021b). The advantage of this code is that it leaves the user free to parameterize his own \mathcal{J} cost functions, which allows him to obtain optimal configurations according to his precise criteria: practical, economic,...

4 Towards a Model of Hydro-Morphodynamism by Optimization

After having carried out numerous works in the optimization of shapes and locations of shoreline protection structures, the works of MOHAMMADI. B, BOUCHETTE. F AND AZERAD. P has taken on a whole new dimension. This new work focused on optimizing the shape of the sea bottom elevation. The sea bottom elevation acts this time as a flexible structure and adapts to a certain hydrodynamic quantity. It is from there that the hydro-morphodynamic models by the principle of minimization are born.

4.1 2D Hydro-Morphodynamic Models Based on the Shallow-Water Equations

The first work on this new way of modeling morphodynamics was done with a very classical hydrodynamic model, namely the Shallow-Water model, here in 2D ([Mohammadi et al. 2011](#); [Mohammadi et al. 2014](#)).

These two publications focused mainly on theoretical developments of the optimal transport morphodynamic modeling method.

This work has the advantage of directly solving a 2D hydro-morphodynamics. However, it turns out that the numerical methods used for this model are very heavy. The method of solving the equations of Shallow-Water is made in finite volumes, the level-set method is used to locate the structure figure [0.10](#). To perform the optimization, an automatic differentiation is used ([Hascoet et al. 2004](#)). The reason is that this hydrodynamic model is too complex to be derived analytically. Once these formalisms were established, functional were tried to account for realistic physics. The following functional of the equation [\(0.5\)](#) aims to minimize agitation and sandy displacements.

$$\mathcal{J}(\psi) = \int_{t-T_{coupl}}^t \int_{\Omega} \left(\|\nabla_{xy} \mathbf{u}\| + \rho_s g (\psi(\tau) - \psi(t - T_{coupl}))^2 \right) d\tau d\Omega \quad (0.5)$$

with u the velocity, ψ the bathymetry and T_{coupl} indicates a time dependency window (s). It also permits to introduce a difference in time scales between seabed and flow motions: it defines the coupling time interval between hydrodynamic and morphodynamic models. The morphodynamic results obtained with this functional are presented on figure [0.10](#). They represent the morphodynamic evolution taking an initial linear sea bottom and a rigid cylinder.

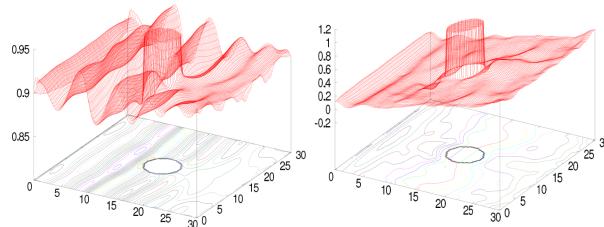


Figure 0.10 – Agitation on the left and final bathymetry on the right (initial bathymetry: linear)

The morphodynamics seems to be reacted in a very realistic way considering the simulation.

These papers conclude that using this method for morphodynamic problems could be a local approach similar to other classical Exner models. It was therefore necessary to go towards a simpler model with more robust physical criteria. It is therefore appropriate to

start with a 1D model, namely Optimorph.

4.2 1D Hydro-Morphodynamic Models Based on Energy Minimization

The new model developed by [Cook \(2021\)](#) is based on the principle that nature seeks to minimize the energy it expands. This time, the \mathcal{J} cost function that governs the evolution of the seafloor has been developed according to the $\mathcal{E}_{\mathcal{H}}$ energy of the waves. This will be presented in the next chapter [1](#).

5 Conclusion

In this state of the art, optimization has been used in a wide range of applications. On the one hand, optimization of the shape or positioning of structures has been used to limit the effect of waves on our coasts. On the other hand, optimization has been used directly in morphodynamic modeling, through work that began in 2011. This latter point is based on the assumption that nature seeks to minimize the energy it expands, which we explained more clearly in the introductory part of this thesis. It is therefore on this point of morphodynamic modelling that this thesis is based.

CHAPTER

Sandy beach dynamics by constrained wave energy minimization

This chapter focuses on a new approach to describe coastal morphodynamics, based on optimization theory, and more specifically on the assumption that a sandy beach profile evolves in order to minimize a wave-related function, the choice of which depends on what is considered the driving force behind the coastal morphodynamic processes considered. The numerical model derived from this theory uses a gradient descent method and allows us to account for physical constraints such as sand conservation in wave flume experiments. Hence, the model automatically adapts to either wave flume or open sea settings and only involves two hyper-parameters: a sand mobility and a critical angle of repose. The ability of OptiMorph to model cross-shore beach morphodynamics is illustrated on a flume configuration. Comparison of the beach profile changes computed with OptiMorph with experimental data as well as the results from the coastal morphodynamic software XBeach demonstrates the potential of a model by wave energy minimization. Numerical robustness and a multi-1D extension are also discussed.

Current chapter contents

1	Introduction	38
1.1	State of the Art	39
1.2	Hypotheses	39
2	Numerical Model	40
2.1	Model forcing	42
2.2	Hydrodynamic Model	42
2.3	Morphodynamic Model by Wave Energy Minimization	45
2.3.1	Link with Morphodynamic Flux-Based Models and Sediment Characteristics Y	47
2.4	Model Constraints	50
3	Numerical Application	51
3.1	Description of the Experiment	51
3.2	XBeach Model	52
3.3	Hydrodynamic Validation	52
3.4	Numerical Results of the Morphodynamic Simulations	53

4	Extension to Multi-1D Simulations	55
4.1	Applications to a Multi-1D Case Near Montpellier	56
5	Discussion	57
5.1	Robustness Analysis of the Consistency in Time and Space of the Morphodynamic model	57
5.2	The Robustness of the Domain Length	59
5.3	Parameter Robustness Analysis	59
5.4	Mid-term Simulations	60
6	Conclusions	63

1 *Introduction*

OPTIMIZATION THEORY is the study of the evolution of a system while searching systematically for the minimum of a function derived from physical properties of the system. In this chapter, we have applied this approach to coastal dynamics, with our primary objective to simulate the interactions between the waves and the sea bottom along a cross-shore profile. Using mathematical optimization theory (Isèbe et al. 2014; Isèbe et al. 2008b; Isèbe et al. 2008a; Bouharguane et al. 2010; Mohammadi et al. 2014; Mohammadi et al. 2011; Cook et al. 2021c; Mohammadi 2017; Bouharguane et al. 2012), we have designed a model that describes the evolution of the sea bottom while taking into account the coupling between morphodynamic and hydrodynamic processes. This study focuses on a theoretical and numerical approach to the modeling of this coupling, based on the assumption that the beach profile adapts to minimize a certain wave-related function. The choice of this function determines the driving force behind the morphological evolution of the beach profile. This optimization problem is subjected to a certain number of constraints, allowing for a more accurate description of the morphodynamic evolution. This study is accompanied by the development of a numerical hydro-morphodynamic model, which has the advantages of being fast, robust, and of low complexity. The model was given the name *OptiMorph*.

The chapter starts with a description of the simple hydrodynamic model used to calculate the driving forces behind the morphodynamic processes. Then, we provide a description of the morphodynamic model (OptiMorph) based on wave-energy minimization. With the purpose of validating OptiMorph, we compare the results of the numerical simulation with that of experimental data acquired in a flume experiment. We also compared the model to another nearshore hydro-morphodynamic model, XBeach (D. J. Roelvink et al. 2009), to see how it fares against existing hydro-morphodynamic models, XBeach being considered to be quite a reputable model in the coastal dynamic community (Zimmermann et al. 2012; Bugajny et al. 2013; Williams et al. 2015).

1.1 State of the Art

Numerical models of morphodynamic processes are seen as a valuable tool for understanding and predicting the evolution of the sediment transport of the morphology over time in coastal areas. Different morphodynamic models exist in the literature, ranging from empirical models (Vriend et al. 1994; Gravens 1997; Kana et al. 1999; Ruessink et al. 2000) to process-based models. The latter can be sorted into several categories, such as i) profile evolution models (Larson et al. 1989; Larson et al. 1990; Nairn et al. 1993), which use only cross-shore transport, ii) rules-based models (Storms et al. 2002; McCarroll et al. 2021), based on a number of rules such as Brunn's rule (Bruun 1954), iii) 2D morphological models (Fleming et al. 1977; Latteux 1980; Coeffe et al. 1982; Yamaguchi et al. 1985; Watanabe et al. 1986; Maruyama et al. 1988; Wang et al. 1993; Johnson et al. 1995; Nicholson et al. 1997; D. J. Roelvink et al. 2009), which use depth-averaged wave and current equations to model the sediment transport while neglecting the vertical variations of wave-derived parameters, as well as iv) 3D and quasi-3D models (J. A. Roelvink et al. 1994; Lesser et al. 2004; D. J. Roelvink et al. 1995; Briand et al. 1993; Zyberman et al. 2002; Ding et al. 2006; Droenen et al. 2007), which determine the sediment evolution using both horizontal and vertical variations of the wave-derived parameters.

The OptiMorph model described in this chapter is based on optimal control. In the past, the use of optimization theory has primarily been used in the design of coastal defense structures, whether in the design of ports and offshore breakwaters (Isèbe et al. 2008b; Isèbe et al. 2008a).

Optimal control has already been considered for the modeling of shallow water morphodynamics, based on the assumption that the seabed acts as a flexible structure and adapts to a certain hydrodynamic quantity (Mohammadi et al. 2011; Bouharguane et al. 2010). These studies were based on somewhat theoretical developments with no direct relationship with real case studies. Our objectives in this work are to produce a physically robust numerical morphodynamic model based on optimal control and to validate it using numerical data from well-established morphodynamics software as well as wave flume experiments.

1.2 Hypotheses

OptiMorph is based on a certain number of assumptions. First, since the model is based on the minimization of a cost function, some hypotheses must be made regarding the choice of this function. This function, which originates from a physical quantity, must be directly linked to the elevation of the seabed. In the current version of the model, we set the quantity to be minimized as the energy of shoaling waves. This implies that the sea bottom reacts to the state of the waves by minimizing the energy of shoaling waves.

Other assumptions assess the behavior of the sea bottom and originate from general observations. In particular, the bed-load sediment transport is controlled by the orbital displacement of water particles (R.L. Soulsby 1987); thus a greater sediment mobility has to be considered in shallower waters. Another natural observation concerns the slope of the seabed, which cannot be overly steep without an avalanching process occurring (Reineck et al. 1973). Last, in an experimental wave flume, the quantity of sand must remain constant over time, with no inflow or outflow of sand to alter the sand stock.

2 Numerical Model

Modeling Framework

For the sake of simplicity, we present the principle of morphodynamics by optimization in a one-dimensional setting. This enables us to compare the numerical results based on this theory with experimental flume data. However, no assumptions are made regarding the dimension of the problem, and as a result, it is straightforward to extend this theory to a two-dimensional configuration.

We consider a coordinate system composed of a horizontal axis x and a vertical axis z . We denote $\Omega := [0, x_{\max}]$ the domain of the cross-shore profile of the active coastal zone, where $x = 0$ is a fixed point in deep-water where no significant change in bottom elevation can occur, and x_{\max} is an arbitrary point at the shore beyond the shoreline, as shown by Figure 1.1. The elevation of the sea bottom is a one-dimensional positive function, defined by: $\psi : \Omega \times [0, T_f] \times \Psi \rightarrow \mathbb{R}^+$ where $[0, T_f]$ is the duration of the simulation (s) and Ψ is the set of physical parameters describing the characteristics of the beach profile. In order to model the evolution over time of ψ and given the assumption that ψ changes over time in response to the energy of shoaling waves, a description of the surface waves is needed.

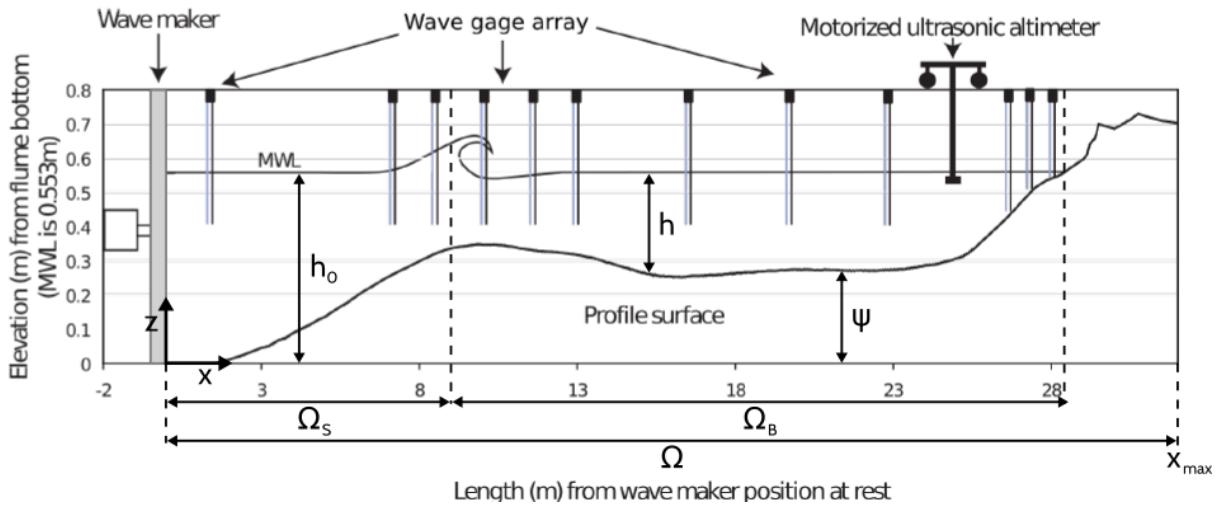


Figure 1.1 – Diagram of a cross-shore profile in the case of an experimental flume.

The model developed by Cook (2021) *et al* is based on the principle that nature tries to minimize the energy it spends. This time, the \mathcal{J} cost function that governs the evolution of the ocean floor is a representative quantity of \mathcal{E}_H , the energy of the waves. The model is based on the following workflow:

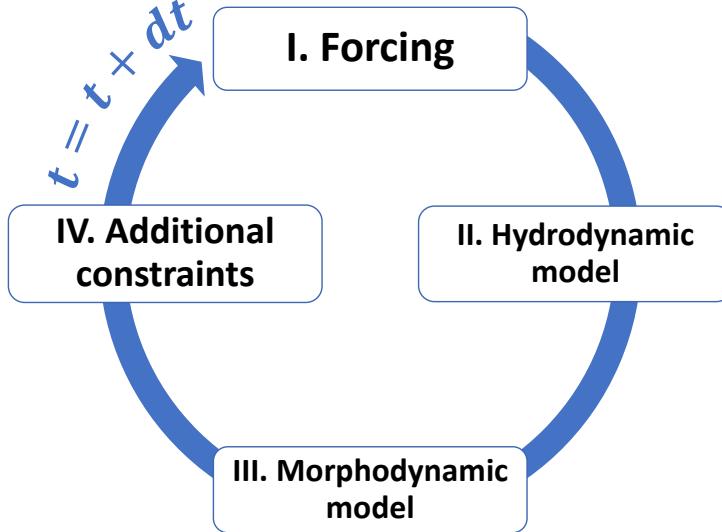


Figure 1.2 – Workflow of Optimorph 2021

with phase I. of forcing 2.1, phase II. of the hydrodynamic model 2.2, phase III. of the morphodynamic model 2.3 and the last phase IV. of constraints 2.4.

2.1 Model forcing

This hydro-morphodynamic model needs a number of forcing parameters. These very classical parameters are divided in 4 parts: numerical, geometric, hydrodynamic and morphodynamic parameters. They are presented in the table below.

Physics	Parameters	Definition
Numerical parameters	Δx	Spatial step [m]
	T_{coupl}	Coupling time between hydro and morpho [s]
	T_f	Simulation time [s]
	n_{iter}	Number of iterations
Domain	Ω	Domain size [m]
	h_0	Offshore water depth [m]
	α	Slope for a linear bathymetry
Hydrodynamic	$H(t)$	Offshore water height [m]
	$T_0(t)$	Wave period [s]
	γ	Breaking criterion
Morphodynamic	ψ_0	Initial Sea bottom elevation
	Y	Sediment mobility [m.s.kg ⁻¹]
	M_{slope}	Maximum slope

Table 1.1 – Table of different forcing parameters

These parameters are very similar to other models. However, there is an original parameter which is the sediment mobility parameter Y . This one will be used in the morphodynamic calculation. It will be defined later.

2.2 Hydrodynamic Model

The literature on hydrodynamic models is vast ([Murray 2007](#)). However, our main focus in this work is a) on the morphodynamic part of the approach and b) on providing evidence of the ability of optimization to perform robust morphodynamic prediction even under weakly constrained hydrodynamics. So we present the procedures with a hydrodynamic model as simple as possible, that is based on the linear wave theory ([Dean et al. 2004](#)), a very basic shoaling equation and some geometrical breaking parameter. It has the advantage of being easy to differentiate compared to more sophisticated models that would need automatic differentiation ([Hascoet et al. 2004; Mohammadi et al. 2011](#)) or huge additional numerical developments. This numerical implementation has a significantly short run-time as shown by the convergence results of the section [5.1](#). This

model has the advantage of expressing wave height as an explicit function of the bottom elevation, which leads to rapid calculations of the morphodynamics.

Let h (m) be the depth of the water from a mean water level h_0 at the point where waves are generated (cf. Figure 1.1). Ocean waves, here assumed monochromatic, are characterized by phase velocity C (m s^{-1}), group velocity C_g (m s^{-1}), and wave number k (m^{-1}), determined by the linear dispersion relation (1.1), where σ is the pulsation of the waves (s^{-1}) and g is the gravitational acceleration (m s^{-2}):

$$\sigma^2 = gk \tanh(kh) \quad (1.1)$$

We define Ω_S as the time-dependent subset of Ω over which the waves shoal and Ω_B the subset of Ω over which the waves break, cf. Figure 1.1. Munk's breaking criterion (W. Munk 1949) enables us to define $\Omega_S(t) = \left\{ x \in \Omega, \frac{H(x,t)}{h(x,t)} < \gamma \right\}$ and $\Omega_B(t) = \left\{ x \in \Omega, \frac{H(x,t)}{h(x,t)} \geq \gamma \right\}$, where γ is a wave breaking index. We have the model below:

Simple Shoaling model

$$H(x, t) = \begin{cases} H_0(t)K_s(x, h) & \text{for } x \in \Omega_S \\ \gamma h(x, t) & \text{for } x \in \Omega_B \end{cases} \quad (1.2a)$$

$$(1.2b)$$

where H is the height of the waves over the cross-shore profile, $H_0(t)$ is the deep-water wave height and K_s is a shoaling coefficient, given by:

$$K_s = \left(\frac{1}{2} \frac{C_0}{C_g} \right)^{\frac{1}{2}} \quad (1.3)$$

where C_0 is the deep-water wave velocity, and:

$$n = \frac{C}{C_g}, \quad C = C_0 \tanh(kh), \quad C_g = \frac{1}{2}C \left(1 + \frac{2kh}{\sinh(2kh)} \right). \quad (1.4)$$

These developments have been detailed in the section 3. This model gives us this type of height H for a linear bathymetry.

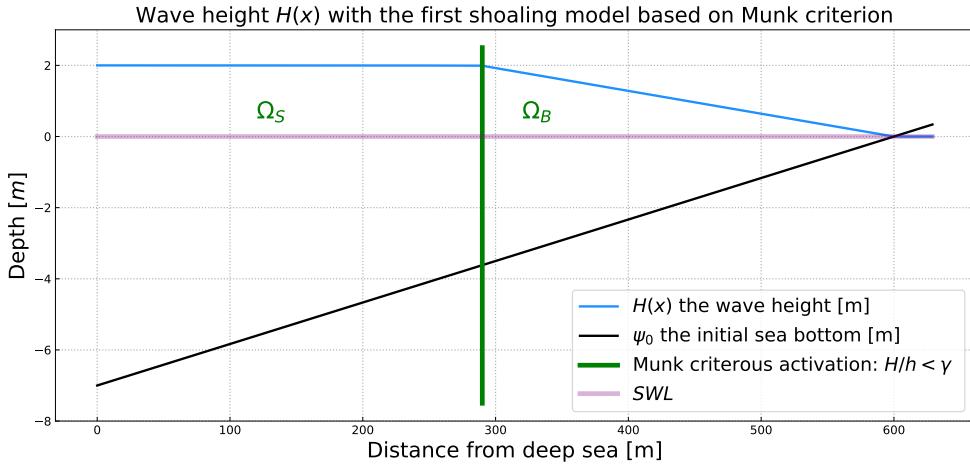


Figure 1.3 – Wave height $H(x)$ with the simple Shoaling model for a configuration with a linear bathymetry of slopes about 0.11, wave period $T_0 = 2\text{ s}$ and offshore water height of $H_0 = 2\text{ m}$.

Instead of considering that waves depend solely on offshore wave height H_0 , this model suggests that shoaling waves are decreasingly influenced by seawards waves. The greater the distance, the less effect it has on the present wave height. As such, we introduce a weighting function w . Assuming that the maximal distance of local spatial dependency of a wave is denoted d_w , the weighting function over the maximal distance d_w is given by $w : [0, d_w] \rightarrow \mathbb{R}^+$ such that $w(0) = 1$, $w(d_w) = 0$ and decreases exponentially.

Equation (1.2a) for shoaling wave height becomes equation (1.5), where H_0^w is defined by (1.6).

$$H(x, t) = H_0^w(x, t)K_S(x, t) \quad (1.5)$$

$$H_0^w(x, t) = \frac{1}{\int_{x-X}^x w(x-y)dy} \int_{x-X}^x w(x-y)H(y)K(y)dy \quad (1.6)$$

Equation (1.5) applies only to the shoaling, nearshore-dependent waves of Ω_S , significant wave height over the cross-shore profile $H : \Omega \rightarrow \mathbb{R}^+$ is defined by (1.7), where $\alpha(x) = \frac{x}{d_w}$ over $[0, d_w]$ to allow a smooth transition between offshore and nearshore-dependent waves.

Complexified Shoaling model

$$H(x, t) = \begin{cases} [(1 - \alpha(x))H_0(t) + \alpha(x)H_0^w(x, t)] K_S(x, t) & \text{if } x \in \Omega_S \text{ and } x < d_w \\ H_0^w(x, t)K_S(x, t) & \text{if } x \in \Omega_S \text{ and } x \geq d_w \\ \gamma h(x, t) & \text{if } x \in \Omega_B \end{cases} \quad (1.7)$$

This gives us this type of height H for a linear bathymetry.

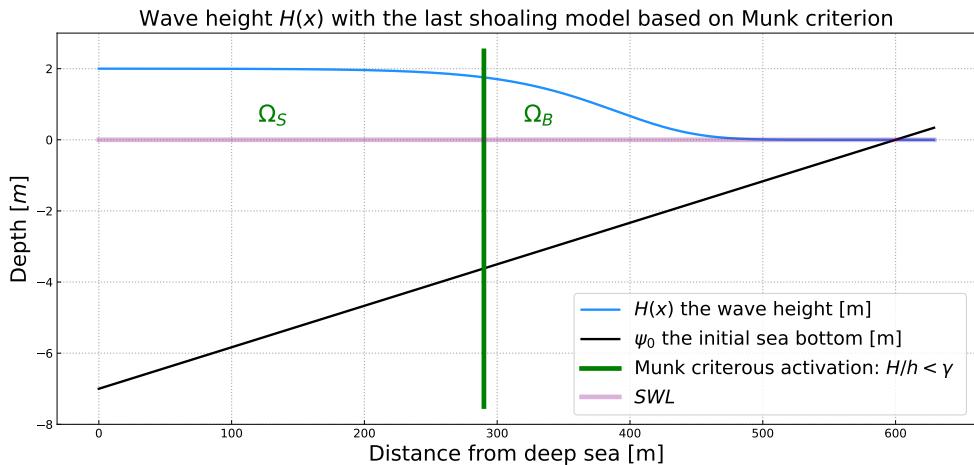


Figure 1.4 – Wave height $H(x)$ with the improved Shoaling model for a configuration with a linear bathymetry of slopes about 0.11, wave period $T_0 = 2$ s and offshore water height of $H_0 = 2$ m.

2.3 Morphodynamic Model by Wave Energy Minimization

The evolution of the sea bottom is assumed to be driven by the minimization of a cost function \mathcal{J} (J s m^{-1}). Recalling the hypotheses made in Section 1.2, the shape of the beach profile is determined by the minimization of the potential energy of shoaling waves, for all $t \in [0, T_f]$:

$$\mathcal{J}(\psi, t) = \frac{1}{16} \int_{t-T_{coupl}}^t \int_{\Omega_S} \rho_w g H^2(\psi, x, \tau) dx d\tau \quad (1.8)$$

where H denotes the height of the waves over the cross-shore profile (m), ρ_w is water density (kg m^{-3}), and g is the gravitational acceleration (m s^{-2}). T_{coupl} (s) defines the coupling time interval between hydrodynamic and morphodynamic models so that we

have T_f/T_{coupl} iterations. In order to describe the evolution of the beach profile, whose initial state is given by ψ_0 , we assume that the sea bottom elevation ψ , in its effort to minimize \mathcal{J} , verifies the following dynamics:

$$\begin{cases} \dot{\psi}_t = Y \Lambda d \\ \psi(t=0) = \psi_0 \end{cases} \quad (1.9)$$

where $\dot{\psi}_t$ is the evolution of the bottom elevation over time (m s^{-1}), Y is a measure of the sand mobility expressed in m s kg^{-1} . This parameter is defined on the basis of flux-based morphodynamic models, as shown in the section 2.3.1. It has the same functionality as XBeach's morphological factor (J.A. Roelvink 2006) where it is possible to divide simulation times by 18 as performed in (Shafei et al. 2023; Marchesiello et al. 2022) on the LIP-1B experiment. Λ measures the excitation of the seabed by the orbital motion of water waves, and d is the direction of the descent (J s m^{-2}), which indicates the manner in which the sea bottom changes. In unconstrained configurations, there would be $d = -\nabla_\psi \mathcal{J}$, which by its definition indicates the direction of a local minimum of \mathcal{J} with respect to ψ as illustrated in figure 1.5.

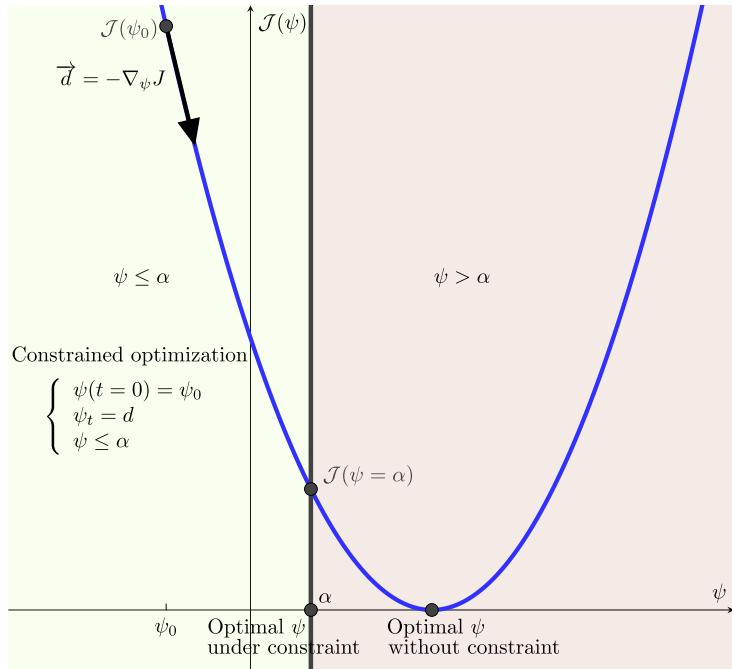


Figure 1.5 – Illustration of gradient descent with $\psi \leq \alpha$. The optimum does not necessarily correspond to the critical point $\nabla_\psi \mathcal{J} = 0$.

The approach involves two parameters with clear physical interpretation: the constraints defined in section 2.4.

Remark: This dynamic described (eq (5.16)), only modifies the bottom elevation and

does not account for lateral displacements. It permits, for instance, the apparition of sedimentary bars but cannot predict their lateral displacements. This will be discussed more thoroughly in the discussion from chapter 3, section 5.2.

2.3.1 Link with Morphodynamic Flux-Based Models and Sediment Characteristics Y

In this section, we show how to link the bed receptivity coefficient in minimization-based to the bed porosity in classical flux-based morphodynamic models. The literature on morphodynamic models is vast (Nielsen 1992; Nielsen 2002; Rooijen et al. 2012; Chen et al. 2023). Modern numerical implementations rely on models which are in a divergence form. For instance, the Exner equation (Paola et al. 2005; Yang et al. 1996) describes the conservation of mass between sediment in the bed of a channel and sediment that is being transported. It states that bed elevation increases (the bed aggregates) proportionally to the amount of sediment that drops out of transport, and conversely decreases (the bed degrades) proportionally to the amount of sediment that becomes entrained by the flow. The model involves the local porosity of the bed $\lambda_p(x) \in [0, 1]$, a function in space x , but not in time. The model writes:

$$\psi_t + \frac{1}{1 - \lambda_p(x)} \operatorname{div}(q(x, t)) = 0,$$

completed with initial and boundary conditions.

In the literature, the expression of q is diverse. But our discussion remains the same whatever may be the formulation of q . For the sake of simplicity, we consider q_x a flux in one dimension of space. Now, let us write the flux-based model and link it to our approach presented through the steepest descent formulation for simplicity:

$$\psi_t = -\frac{1}{1 - \lambda_p(x)} q_x = -Y(x) \Lambda(x) \nabla_\psi \mathcal{J}(x, t). \quad (1.10)$$

There is no explicit boundary condition in the minimization model. In this case, we consider $\Lambda(x) = 1$, the maximum disturbance. As we saw, global sand conservation, can be evaluated through a constraint. In the same way, the local maximum slope is expressed as a constraint. $\nabla_\psi \mathcal{J}(x, t)$ corresponds to the direction of the descent without constraint and d with. The bed receptivity $Y(x)$ is a positive function which we link to the couple bed porosity $\lambda_p(x)$ and flux q as follows.

Locally integrating in space equation (1.10) over a small interval $[x - \varepsilon, x + \varepsilon]$ around x we have:

$$\int_{x-\varepsilon}^{x+\varepsilon} Y(s) \nabla_\psi \mathcal{J}(s, t) ds = \int_{x-\varepsilon}^{x+\varepsilon} \frac{1}{1 - \lambda_p(s)} q_s(s, t) ds.$$

Assuming Y and λ_p constant over this small interval, which is physically realistic, we

have:

$$Y(x) \int_{x-\varepsilon}^{x+\varepsilon} \nabla_\psi \mathcal{J}(s, t) ds = \frac{1}{1 - \lambda_p(x)} \int_{x-\varepsilon}^{x+\varepsilon} q_s(s, t) ds.$$

This leads to:

$$Y(x) \int_{x-\varepsilon}^{x+\varepsilon} \nabla_\psi \mathcal{J}(s, t) ds = \frac{1}{1 - \lambda_p(x)} (q(x + \varepsilon, t) - q(x - \varepsilon, t))$$

which we write as:

$$Y(x) = F(x, t) \frac{1}{1 - \lambda_p(x)},$$

where factor $F(x, t)$:

$$F(x, t) = \frac{q(x + \varepsilon, t) - q(x - \varepsilon, t)}{2\varepsilon \overline{\nabla_\psi \mathcal{J}}|_{(x,t)}}$$

represents the ratio between the local flux difference and the local average shape gradient $\overline{\nabla_\psi \mathcal{J}}|_{(x,t)} = (1/(2\varepsilon)) \int_{x-\varepsilon}^{x+\varepsilon} \nabla_\psi \mathcal{J}(s, t) ds$ at point x .

If the bed porosity does not change in time, this evaluation is made only once at $t = 0$ and hence, given a flux and a bed porosity, the corresponding minimization-based procedure can receive an equivalent pointwise initialization (at the first iteration in an iterative time integration procedure).

In operational conditions however, it is very unlikely to have a pointwise, even inaccurate, estimation of $\lambda_p(x)$. It is more reasonable to look for an 'equivalent' constant bed porosity for a given site knowing that what is important in coastal engineering is not the knowledge of the pointwise bed porosity, but the prediction of beach future behaviour based on this site macroscopic characteristics. Also, in homogeneous bed, as it is often the case in sandy beaches, $\lambda_p(x)$ is a constant. We therefore look for a constant bed receptivity $Y = \bar{F} \frac{1}{1 - \lambda_p}$ over the domain of interest $]x_L, x_R[$ (L, R indicating Left and Right) given constant bed porosity λ_p and flux q with

$$\bar{F} = \frac{q(x_R, 0) - q(x_L, 0)}{\int_{x_L}^{x_R} \nabla_\psi \mathcal{J}(s, 0) ds}, \quad (1.11)$$

which is a scalar, and the ratio between flux variation over $]x_L, x_R[$ and the average of local shape gradients. Here we have defined x_L and x_R as the Left and Right extremities of the domain. So we have $q(x_R, 0)$ and $q(x_L, 0)$ the boundary conditions of the flux-based model. \bar{F} is a measure of how the evaluation of local-based and optimization-based fluxes differs.

2.3.1.1 Illustration Using a Simple Model

Assuming that we are on a configuration of bed load transportation without suspended transport, we can calculate $q(x_R, 0)$ and $q(x_L, 0)$ by using a formula of the bed load transport rate q with (Fredsøe et al. 1992):

$$q = 10 \frac{\pi}{6} d_{50} p U'_f [1 - 0.7 \sqrt{\theta_c / \theta'}] \quad (1.12)$$

with d_{50} the grain diameter, p the fraction of bed surface particles in motion, U'_f the skin friction velocity, θ_c the critical Shields parameter and θ' the Shields parameter. This formula has been chosen as one of the simplest. However, we can choose to take suspended sediment transport into account, simply by changing the expression of q in our model. Combining the equations (1.11) and (1.12), we obtain the following expression of \bar{F} :

$$\bar{F} = \pi d_{50} [10 - 7 \sqrt{\theta_c / \theta'}] \frac{p(x_L) U'_f(x_L) - p(x_R) U'_f(x_R)}{6 \int_{x_L}^{x_R} \nabla_\psi \mathcal{J}(s, 0) ds}.$$

We have shown how a conjunct giving of a bed porosity and a flux permits the initialization of a minimization model according to the parameters of the chosen local flux-based model which is comforting for users familiar with such a more traditional approach. However, a same initialization does not mean that the two models will follow the same path, as the minimization-based approach introduces more physics. Indeed, in previous works, we have already shown how our minimization-based formulation can be seen as an Exner equation with a non-local flux (Mohammadi et al. 2011; Bouharguane et al. 2012) with terms similar to those encountered in Fowler-like models (Fowler 2001; Kouakou et al. 2006). Those terms bring the contribution of some non-local physics to the morphodynamics.

This formulation also permits the comparison of the bed ψ evolution predicted minimizing different physical functional \mathcal{J} . It is thus a very efficient exploratory model as defined by Murray (2007). However, it is not possible to find the functional \mathcal{J} associated to a given flux q because this requires the mathematical concept of integration with respect to the shape to give sense to:

$$\mathcal{J} = \frac{Y}{1 - \lambda_p} \int_\psi \nabla \cdot q \, d\psi.$$

Unfortunately, unlike differentiation with respect to the shape (Mohammadi et al. 2009), the concept of integration with respect to the shape does not exist as of today.

2.4 Model Constraints

The first constraint Y takes into account the physical characteristics of the sand and represents the mobility of the sediment. Simulations with varying Y that reflect variations of the d_{50} grain diameter from 0.25 mm to 2 mm were performed. Changes in the beach profile were observed but no significant alteration of the trends in beach profile evolution through time. The asymptotic behavior of the simulations remains the same although the velocity at which a given profile is reached changes. Further explanation of the nature of the Y parameter will be given at a later stage of the model development. The second parameter Λ is a local function which represents the influence of the relative water depth kh on the beach profile dynamics and is defined after the term describing the vertical attenuation of the velocity potential according to linear wave theory ([R.L. Soulsby 1987](#)):

$$\begin{aligned} \varphi : \Omega \times [0, h_0] &\longrightarrow \mathbb{R}^+ \\ (x, z) &\longmapsto \frac{\cosh(k(x)(h(x) - (h_0 - z)))}{\cosh(k(x)h(x))} \end{aligned} \quad (1.13)$$

In unconstrained circumstances, for instance, if a total sand volume constraint does not need to be enforced, we set $d = -\nabla_\psi \mathcal{J}$, which indicates a direction for local minimization of \mathcal{J} with regards to ψ . The calculation of $\nabla_\psi \mathcal{J}$ is described in [2](#). However, constraints are added to the model to incorporate more physics and to deliver more realistic results. While driving forces behind the morphological evolution of the beach profile are described by the minimization of the cost function \mathcal{J} , secondary processes are expressed by constraints. In the interest of simplicity, we have adopted two physical constraints though more can be introduced if necessary. The first concerns the local slope of the bottom. Depending on the composition of the sediment, the bottom slope is bounded by a grain-dependent threshold M_{slope} ([Dean et al. 2004](#)). This is conveyed by the following constraint on the local bottom slope illustrated by [1.6](#):

$$\left| \frac{\partial \psi}{\partial x} \right| \leq M_{\text{slope}} \quad (1.14)$$

The dimensionless parameter M_{slope} represents the critical angle of repose of the sediment. This angle is based on observed angles in natural beach environments, which are often between 0.01 and 0.2 ([Bascom 1951](#); [Vos et al. 2020](#); [Short 1996](#)). We have considered the observed critical angle of 0.2.

A second example concerns the sand stock in the case of an experimental flume. In a flume, the quantity of sand must be constant over time, as given by [\(5.24\)](#), contrarily to an open-sea configuration where sand can be transported between the nearshore zone and a domain beyond the closure water depth where sediment is definitely lost for beach

morphodynamics ([Hattori et al. 1980](#); [Quick 1991](#)). This constraint can be written as :

$$\int_{\Omega} \psi(t, x) dx = \int_{\Omega} \psi_0(x) dx \quad \forall t \in [0, T_f] \quad (1.15)$$

This constraint is necessary for verifying and validating the numerical model with the wave flume experimental data.

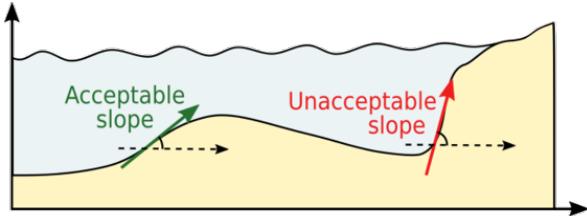


Figure 1.6 – Slope constraint (5.23) from (Cook 2021)

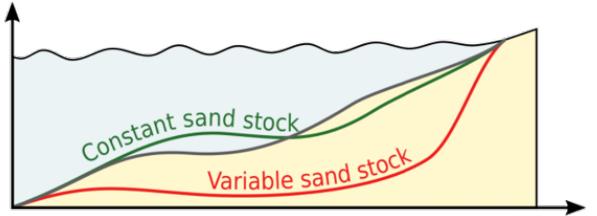


Figure 1.7 – Sand conservation (5.24) from (Cook 2021)

3 Numerical Application

In this section, we present the numerical results produced by the OptiMorph model. For validation purposes, the resulting beach profile is compared to experimental data acquired during a flume experiment. We also conduct a comparative analysis between the beach profiles produced experimentally, by OptiMorph and by XBeach, with the aim of assessing how OptiMorph holds up against existing hydro-morphodynamic models. A brief description of the experiment is provided, as well as the XBeach model.

3.1 Description of the Experiment

The experimental observations have been collected as part of the COPTER project and a series of laboratory wave-flume experiments were performed in order to investigate the morphodynamic impact of introducing solid geotextile tubes in the nearshore ([Bouchette 2017](#)). We use the part of the experiment run without tubes that was devoted to the description of the natural evolution of the beach profile under various wave conditions. Time and length scale ratios are set to 1/3 and 1/10 respectively to that of the field.

A flume measuring 36 m long, 0.55 m wide and 1.3 m deep is equipped with a wave-maker and gauges measuring the elevation of the water surface from which wave properties are derived. Artificial particles are placed inside the flume representing the mobile sea bottom and an ultrasonic gauge is used to measure the experimental beach profile. The experimental beach profile, described in Figure [1.1](#) is subjected to a series of 30-minute storm climates, among which a typical moderate storm event (at the scale of the flume) with a significant wave height and period of $H_s = 135$ mm and $T_s = 2.5$ s.

3.2 XBeach Model

XBeach is an open-source process-based model developed by Deltares, UNESCO-IHE, and Delft University of Technology to simulate the hydro-morphodynamic processes in coastal areas (D. J. Roelvink et al. 2009; Zimmermann et al. 2012; Bugajny et al. 2013; Williams et al. 2015). In brief, XBeach uses four interconnected modules to model near-shore processes (Daly 2009; D. J. Roelvink et al. 2010). The two hydrodynamic modules consist of the short-wave module and the flow module. The first is based on wave action equations (Holthuijsen et al. 1989), and incorporates breaking, dissipation (D. J. Roelvink 1993), and wave current interactions, while the latter is governed by shallow water equations (Andrews et al. 1978; Walstra et al. 2000). One of the two morphodynamic modules is the sediment transport module based on the equilibrium sediment concentration equation (Richard Soulsby 1997) and a depth-averaged advection-diffusion equation (Galappatti et al. 1985). The other is the morphology module which concerns seabed transformations such as the evolution of the sea bottom and avalanching.

For the simulations, the domain Ω is defined over 32 m with a uniform subdivision of 320 cells. The incoming wave boundary condition is provided using a JONSWAP wave spectrum (Daly 2009), with a significant wave height of $H_{m0} = 0.015$ m and a peak frequency at $f_p = 0.4$ s⁻¹. The breaker model uses the Roelvink formulation (D. J. Roelvink 1993), with a breaker coefficient of $\gamma = 0.4$, a power $n = 15$, and a wave dissipation coefficient of 0.5. These parameters were calibrated using the hydrodynamic data produced during the physical flume experiment. Concerning sediment parameters, the d_{50} coefficient is set as 0.0006, and the porosity is 2650 kg m⁻³. No other parameters such as bed friction or vegetation were applied. The model is set to run for a period of 1800 s, as a short-term simulation.

3.3 Hydrodynamic Validation

This section is devoted to the comparison of the two numerical hydrodynamic models to the experimental wave data obtained in the experimental flume of section 3.1. Mean wave height profiles were calculated over the short-term storm simulation, for both Opti-Morph and XBeach, and compared to the mean wave height of the experimental model. The latter was calculated using the measures taken by the gauges of the flume.

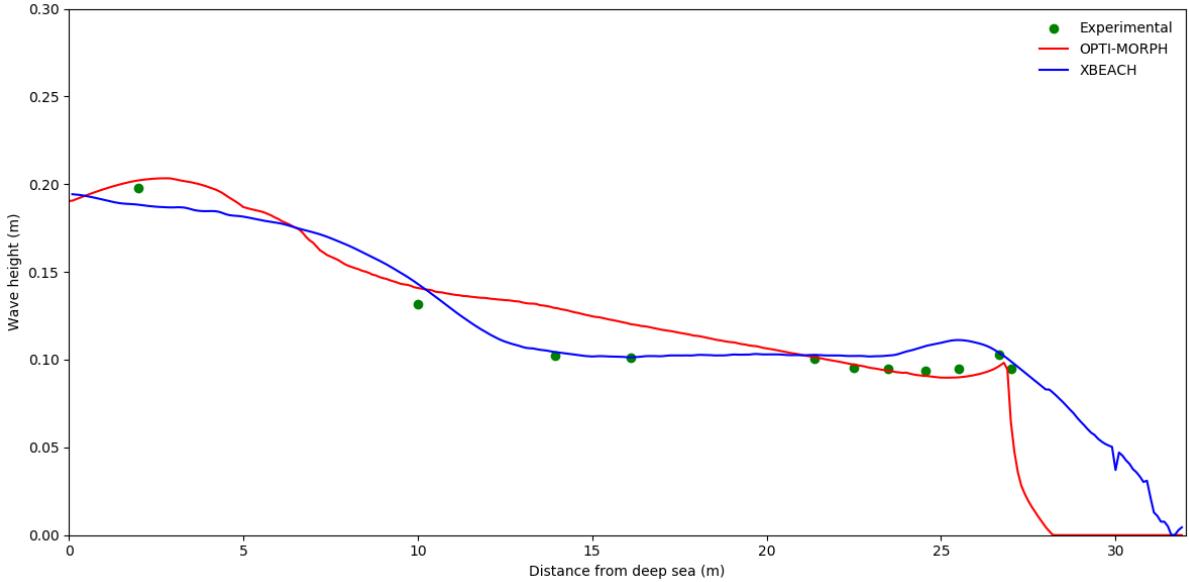


Figure 1.8 – Comparison of mean wave height over a storm simulation. The green points correspond to the mean wave height provided by the gauges of the flume experiment. The mean wave height determined by OptiMorph (red) and XBeach (blue) also appears. The non-zero wave height beyond the shoreline as presented by XBeach is due to wave set-up, which OptiMorph does not handle.

Figure 1.8 shows that the hydrodynamic module of both OptiMorph (red) and XBeach (blue) are both comparable with respect to the experimental measurements (green) excluding, as is often the case, the second point at $x = 6$ m. XBeach demonstrates a close qualitative fit over the 10-22 m section of the flume, whereas OptiMorph excels at the coast (21-27 m), with a near-perfect fit with the experimental data. Despite the simplicity of the hydrodynamic model used by OptiMorph, the resulting wave height is of the same order of magnitude over the cross-shore profile than that measured during the flume experiment, which indicates that the resulting beach profile would be comparable with regard to the forcing energy driving the morphodynamic response.

3.4 Numerical Results of the Morphodynamic Simulations

The OptiMorph model was applied to the configuration of the COPTER experiment of section 3.1, and the resulting beach profile is shown by the red profile, in Figure 1.9.A. The main observation is the decrease of 2.5 cm in height of the sandbar, at $x = 9$ m. We observe a slight lowering of the sea bottom adjacent to the wave-maker, and a slight increase at the plateau, situated at 15-25 m. No mobility is observed at the coast.

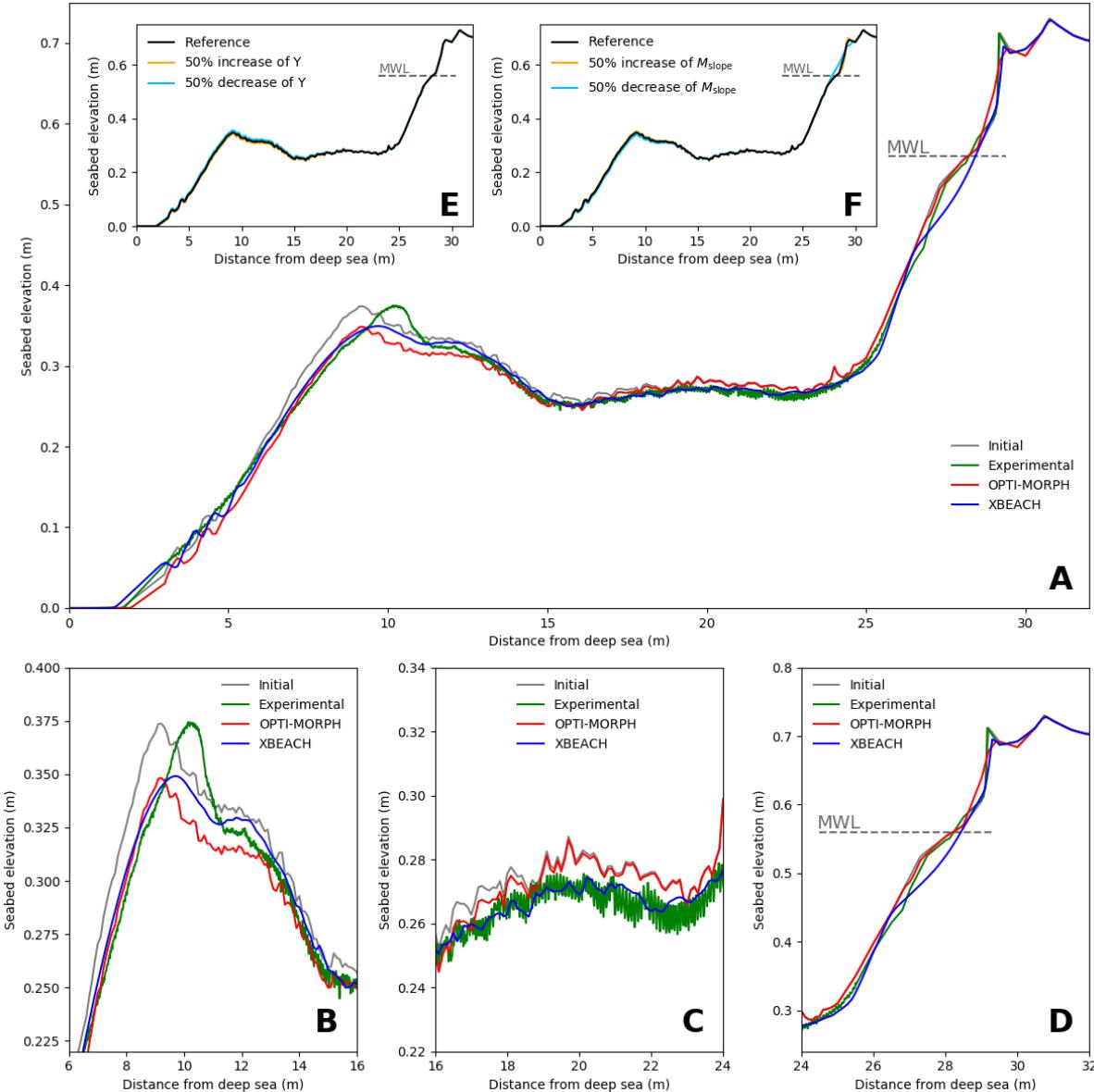


Figure 1.9 – A. Results of the numerical simulation calculated over the initial seabed (gray) using the XBeach morphodynamic module (blue) and the OptiMorph model (red). These are compared with the experimental data acquired during the COPTER project (green). The mean water level is denoted MWL and is set at 0.56 m. **B.** Zoomed in view of the sandbar, located between 6 m and 16 m. **C.** Zoomed in view of the plateau, located between 16 m and 24 m. **D.** Zoomed in view at the shoreline, located between 24 m and 32 m. **E.** Robustness analysis of the mobility parameter Y . The reference profile is depicted in black. The orange (resp. light blue) profile is the result of a 50% increase (resp. decrease) in mobility, with all other parameters remaining the same. **F.** Robustness analysis of the maximal sand slope parameter M_{slope} . The reference profile is depicted in black. The orange (resp. light blue) profile is the result of a 50% increase (resp. decrease) of M_{slope} , with all other parameters remaining the same.

When comparing the results provided by OptiMorph (red), with that of XBeach (blue) and the experimental data (green), as shown on Figure 1.9.A, we observe that the red beach profile provided by the OptiMorph model shows a general quantitative agreement

when compared to the experimental data, as does the XBeach morphological module. In fact, both models produce profiles close to the experimental data over the plateau located at 15-25 m from the wave-maker (Fig. 1.9.C). At the shore, OptiMorph matches the experimental data whereas XBeach shows a vertically difference of up to 3 cm at $x = 27$ m (Fig. 1.9.D). Discrepancies on the part of both models occur in the area surrounding the tip of the sandbar, as both OptiMorph and XBeach fail to predict the shoreward shift of the sandbar (Fig. 1.9.B); the experimental data shows that the height of the sandbar remains unchanged with regards to the initial profile. Both sandbars have a height of 0.375 m; however, the sandbar resulting from the experimental simulation has moved towards the coast, an occurrence that neither numerical model was able to predict.

As such, this new model based on wave-energy minimization shows potential when compared to XBeach, in the case of short-term simulations.

4 **E**xtension to Multi-1D Simulations

The OptiMorph multi-1D model (Ronan Dupont et al. 2022) is an extension of 1D. It slices a 3D bathymetry into n cross-shore transects. They are then launched simultaneously as shown in figure 1.10. They can be run on a classical computer or on a cluster for computation time.

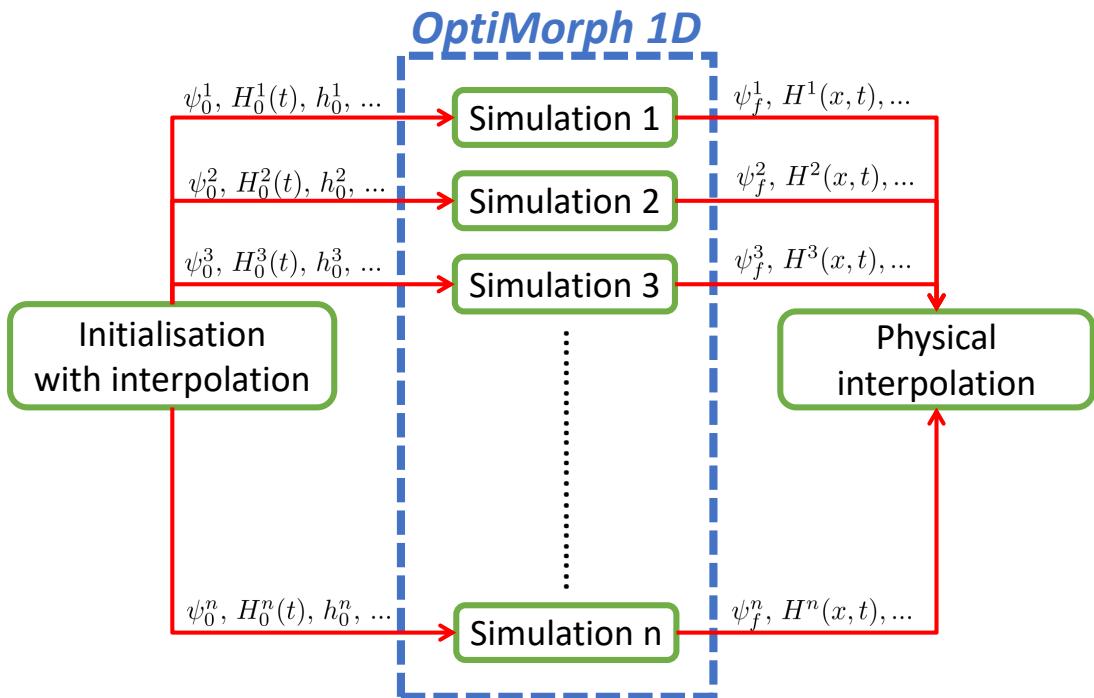


Figure 1.10 – Functional diagram of the multi-1D

Once the n simulations are done, it is essential to link all these simulations by interpo-

lation. This model, the most basic one, has some limitations. Indeed, these simulations assume that all transects have been extracted at locations where there are no long-shore currents. In addition, lateral sandy displacements are not taken into account. In a more advanced version, it could be interesting to take into account these phenomena by adding a source term in the descent equation governing the evolution of ψ .

4.1 Applications to a Multi-1D Case Near Montpellier

The works of Isèbe et al. (2008b) and Bouharguane et al. (2010) are interested in finding the optimal position of geotextile tubes (protection solutions) on a transverse profile to maximize their effect as a wave attenuator. With this type of approach in mind, we wish to perform a set of multi-1D simulations by placing a geotextile tube on a set of real transverse morphological profiles (Montpellier; Figure 1.11) structure.

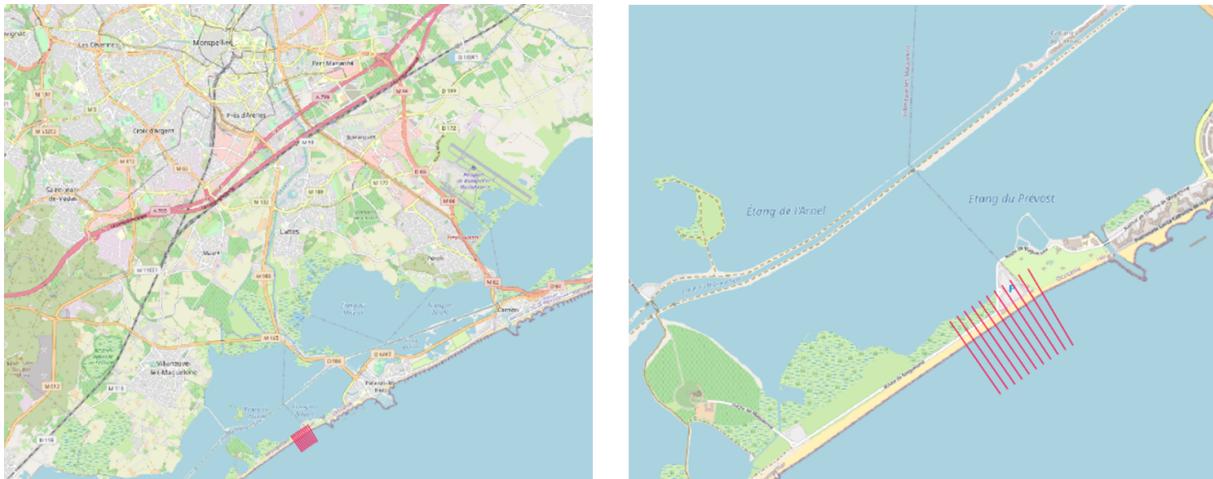


Figure 1.11 – Geography of the simulation near Montpellier

We carry out the simulation of an extreme multi-1D case (on all the profiles); this simulation corresponds to a storm of a few days with waves of a maximum height of $H_{max} = 2\text{ m}$. A geotextile tube is added on the domain in a Gaussian form (whose characteristics vary from one profile to another; figure 1.12.a). Parallelized simulations (HPC computer) are launched and are completed in less than 5 minutes whatever the number of transects (modulo the limit of cores on the cluster). The results are presented in figure 1.12.

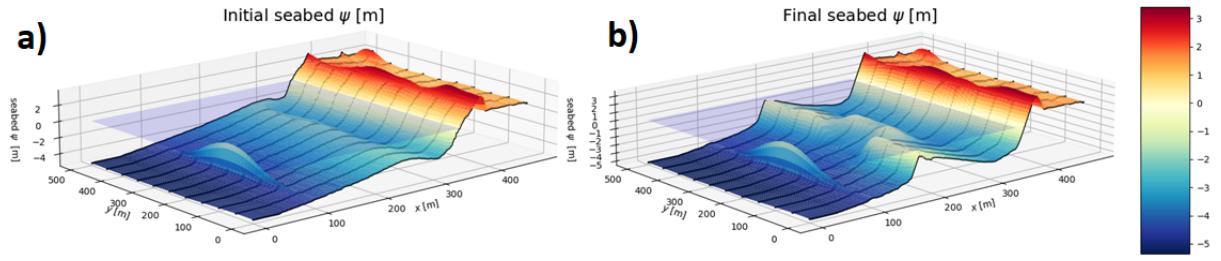


Figure 1.12 – a) Initial sea bottom on the Gulf of Aigues-Mortes by adding a Gaussian geotube with a maximum height of 3m. b) Final bathymetry after a simulation of a storm of several days.

The results show the formation of a pit at the back of the highest and thickest part of the geotextile tube. These results are encouraging as they are very similar to those found in the COPTER experimental campaign ([Bouchette 2017](#)) conducted in 2017 in a 3D wave basin (moving bottom).

5 **D**iscussion

5.1 Robustness Analysis of the Consistency in Time and Space of the Morphodynamic model

We computed a reference OptiMorph simulation using a very small coupling time of 0.05 s which is much smaller than what is usually used in hydro-morphodynamic simulations. The simulation was performed with the original bathymetric profile of the COPTER experiment and the forcing of the wave maker.

This simulation provides a reference computed sea bed $\psi_{ref}(T_f, x)$ at some given time T_f . We would like to see the convergence toward this reference solution of various other OptiMorph simulations with different decreasing time steps. From this series of simulations, we quantify a residual error with L^2 norms as $\mathcal{E}_{L^2} = \|\psi_{ref} - \psi\|_{L^2}$ in m. We performed 10 simulations with time steps ranging in $[0.05; 160]$ s and we get the results described in figure [1.13.a](#)).

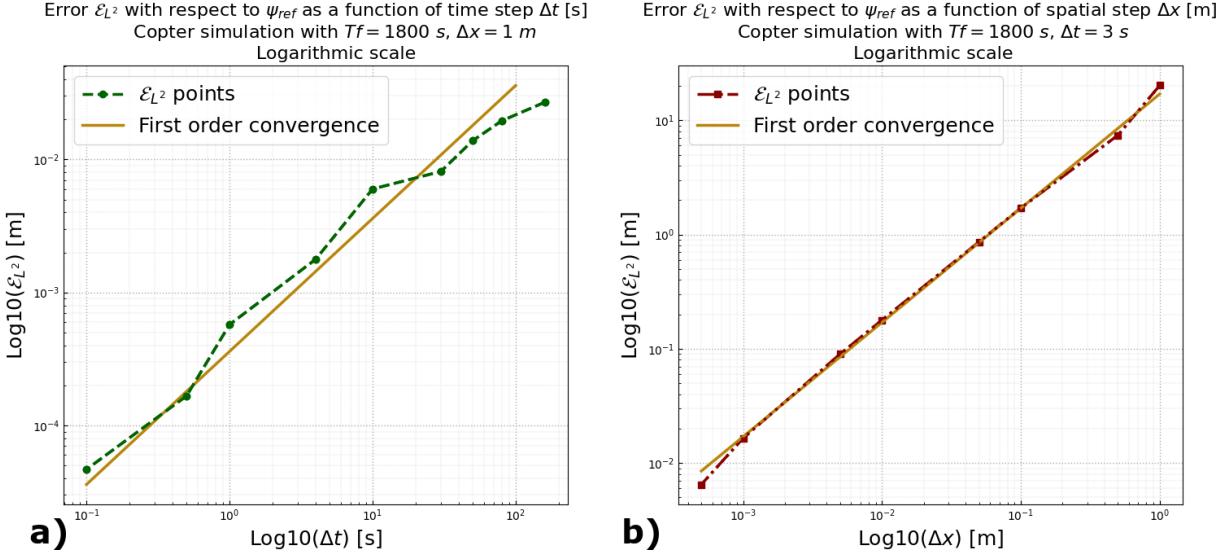


Figure 1.13 – a) Errors \mathcal{E}_{L^2} (green) obtained by simulations of 10 different time steps compared to the reference simulation corresponding to a coupling time of 0.05 s. First order convergence (yellow). b) Errors \mathcal{E}_{L^2} (red) obtained by simulations of 10 different spatial steps compared to the reference simulation corresponding to a spatial step of 0.0002 m. First order convergence (yellow).

In order to analyse the convergences in space and time, we choose, respectively, a reference coupling time of $T_{coupl} = 3$ s and a spatial step size $\Delta x = 1$ m. $T_{coupl} = 3$ s corresponds to the kind of time steps we would like to use in simulations. But, we will use larger spatial resolution in practice. The results in figure 1.13 shows first order (illustrated by the continuous line) convergence rates in both time and space.

To understand why a coupling time of 3 seconds is interesting for computing efficiency, it is useful to look at the CFL stability condition analysis for the shallow-Water model (Marche et al. 2007). The analysis provides a typical upper bound for the time step of the form:

$$\Delta t = \min_i \left(\frac{\Delta x}{2 \max_i(|u_i \pm \sqrt{gh_i}|)} \right) = \frac{\Delta x}{2 \max_i(|u_0 \pm \sqrt{gh_0}|)},$$

where subscript i indicates the mesh node which means that the minimum is taken over all the nodes of the mesh. In our situation, it corresponds to the off-shore position (subscript $i = 0$). Typical values in our simulation are: $u_0 = 10$ m s $^{-1}$, $\Delta x = 1$ m, $h = 0.55$ m and $g = 9.81$ m s $^{-2}$. This gives us $\Delta t = 0.04$ s, which is about two orders of magnitude smaller than our reference time step of $\Delta t = 3$ s. In addition, the costs of one iteration of the Shallow-Water and OptiMorph models are comparable.

5.2 The Robustness of the Domain Length

An important step in the analysis of the numerical behavior of the model is the validation of its behavior with respect to the characteristics of the domain and with respect to the forcing. In this part, we are particularly interested in the comparison of simulations of the morphodynamic evolution for identical forcing but different domains. A multi-day storm is studied by varying the length of the domain while keeping the same linear range of slope 1e-2. The results obtained are presented in Figure 1.14.b and show the formation of a realistic stable bar for domain lengths in the interval [600, 1200, 1800, 3000] m.

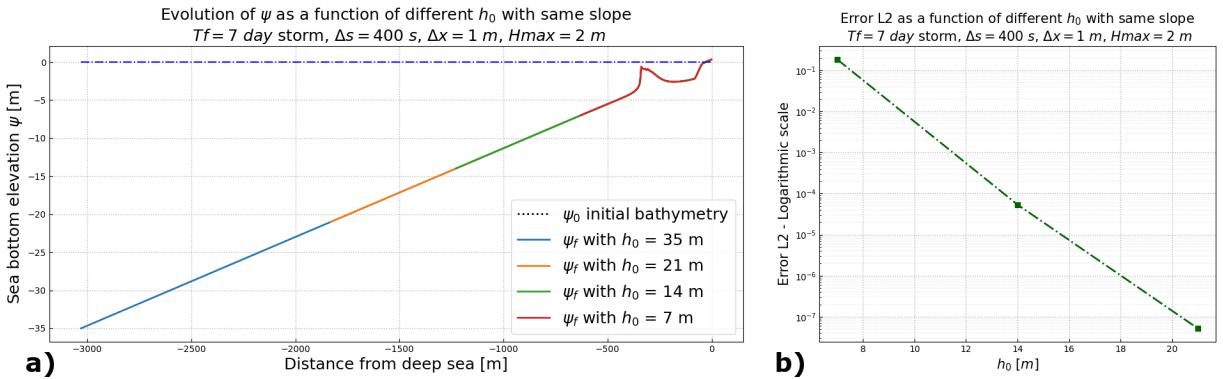


Figure 1.14 – a) Different final bottom profiles for domains of sizes [600, 1200, 1800, 3000]m corresponding to offshore water heights of $h = [7, 14, 21, 35]$ m. The representation is truncated at 7 m depth. **b)** L^2 error comparing each simulation to the reference solution corresponding to the 3000 m domain (the longest, represents deep-water conditions).

Figure 1.14.b shows the convergence of the solution as a function of the change in domain length. As for the convergence on spatial resolution, we consider a reference simulation which is that of a domain of length 3000 m. The observed convergence thus ensures that regardless of the domain size, the shallow water swell transition always occurs at the same point and the morphodynamic response occurs in the same way.

5.3 Parameter Robustness Analysis

One of the advantages of the OptiMorph model is the low number of morphodynamic hyper-parameters required. At the present time, OptiMorph requires two hyper-parameters: the mobility parameter Y and the maximal slope parameter M_{slope} . Here, an assessment on these parameters is conducted. In Figure 1.9.E, three simulations were performed in identical settings with changes made solely to the mobility parameter. Initially, this parameter Y has a value of $5 \times 10^{-6} \text{ m s kg}^{-1}$. Figure 1.9.E shows no significant difference despite a 50% increase ($Y = 7.5 \times 10^{-6} \text{ m s kg}^{-1}$) (orange) or decrease ($Y = 2.5 \times 10^{-6} \text{ m s kg}^{-1}$) (light blue) of Y with regard to the baseline beach profile (black). A similar conclusion can be deduced for the maximal slope parameter M_{slope} , whose reference value here is 0.2. The corresponding parameter of XBeach is *wetslp*,

described in the XBeach manual as the critical avalanching slope under water, and is also set to 0.2. In Figure 1.9.F, we observe little difference between the reference seabed (black), the seabed resulting from a 50% increase ($M_{slope} = 0.3$) (orange) and the seabed resulting from a 50% decrease ($M_{slope} = 0.1$) (light blue). The only apparent discrepancy can be found at $x = 28$ m, where the bottom slope is at its steepest, and therefore the sand slope constraint is more prone to be active. The reduction of the critical angle of repose results naturally in a less steep slope. The robustness of OptiMorph in relation to both the mobility parameter and the slope parameter, despite a significant increase or decrease of their value, is apparent. Further simulations show that the robustness of these parameters is not specific to this particular flume configuration, but can be observed regardless of the initial configuration.

5.4 Mid-term Simulations

This section is devoted to a medium-term behavior of OptiMorph, the main question being, is this numerical model capable of creating an equilibrium state after being subjected to a great number of repeated events. Five forcing scenarios, lasting either 2 or 6 days, were applied to the same initial seabed in the same parametric configuration. The current OptiMorph code is in Python. Typically, using time steps of 1 s simulating a day of forcing requires about 1.5 hours on a 2GHz PC computer. Each time iteration gathering the steps presented in this chapter requires therefore about 63 ms. Regarding the section 5.1, we could use 3 s time step and divide the simulation time by 3. An analysis of the resulting beach profiles is performed as well as their behavior throughout the simulation. The latter is achieved through a comparative study of four-time series, focusing on: (1), the vertical evolution of bottom elevation at the tip of the sandbar; (2), the vertical evolution of bottom elevation at a point of the plateau; (3), the distance between the wave-maker and the onset of the sea bottom; and (4), the location of the shoreline position.

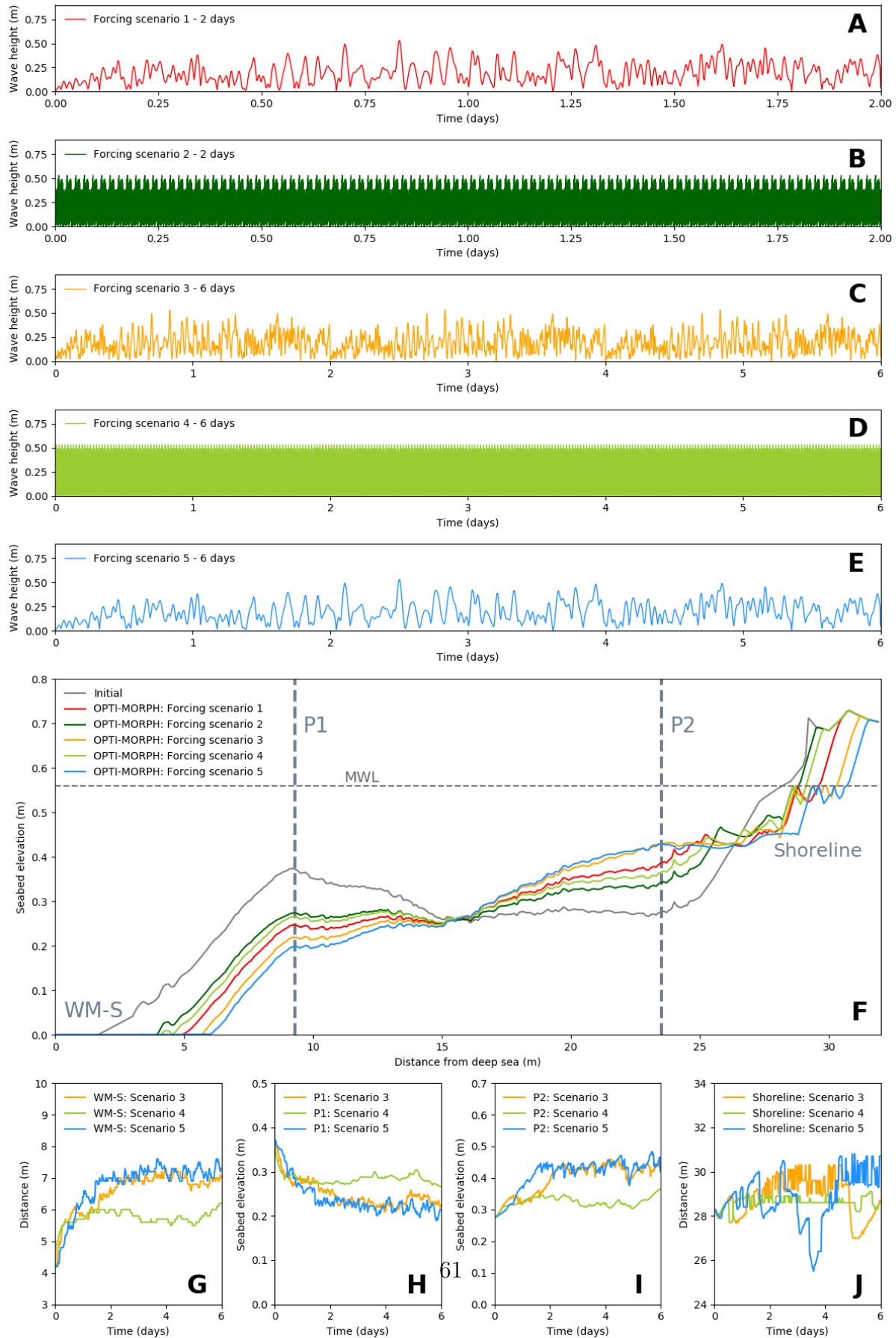


Figure 1.15 – Mid-term simulation of OptiMorph. **A.** Forcing wave height for scenario 1, composed of several mid-term events over a 2-day period. **B.** Forcing wave height for scenario 2, composed of numerous short-term events over a 2-day period. **C.** Forcing wave height for scenario 3, composed of several mid-term events over a 6-day period. **D.** Forcing wave height for scenario 4, composed of numerous short-term events over a 6-day period. **E.** Forcing wave height for scenario 5, composed of few mid-term events over a 6-day period. **F.** Seabeds resulting from the different forcing scenarios produced by OptiMorph. Two points of interest have been identified: P1 located at $x = 9.3$ m and P2 located at $x = 20.1$ m. **G.** Evolution of the distance, devoid of sediment, between the wave-maker (located at $x = 0$ m) and the seabed (WM-S), regarding forcing scenarios 3, 4, and 5. **H.** Vertical evolution of seabed elevation at P1, driven by the 6-day forcing scenarios 3, 4, and 5. **I.** Vertical evolution of seabed elevation at P2, driven by the 6-day forcing scenarios 3, 4, and 5. **J.** Evolution of shoreline position, driven by the 6-day forcing scenarios 3, 4, and 5.

Applying OptiMorph over a longer time-series leads to the results of Figure 1.15. The two 2-day forcing scenarios are shown in Figures 1.15.A and 1.15.B. In both cases, we observe that the resulting beach profiles in Figure 1.15.F are subjected to the destruction of the sandbar and have a tendency to evolve progressively towards an equilibrium beach profile (Engineers 2002). Simulations over a 6-day period were conducted to confirm this tendency. These scenarios are depicted in Figures 1.15.C, 1.15.D, and 1.15.E; the resulting profiles given in Figure 1.15.F show once again the destruction of the sandbar, the elevation of the plateau, and some erosion at the shoreline. Furthermore, all three tend towards an equilibrium state. This is confirmed by the four time-series analysis presented in Figures 1.15.G, 1.15.H, 1.15.I, and 1.15.J. The vertical elevation of the seabed at both points P1 and P2 show initial variations over the first 2 days: a decrease in the case of P1 (cf. Figure 1.15.H) and an increase in the case of P2 (cf. Figure 1.15.I). However, both studies show a stabilization of the sea bottom elevation over the last 4 days of the 6-day period. Similar conclusions can be drawn regarding the length of the zone containing no sediment adjacent to the wave-maker (cf. Figure 1.15.G). An initial increase between 2 and 3 meters can be observed, with stability achieved in the later stages of the simulations. Finally, Figure 1.15.J shows the evolution of the shoreline position. Initially found at $x = 28.3$ m, all scenarios provoke a retreat of the shoreline: 0.4 m in scenario 3, 0.3 m in scenario 4, and 2 m in scenario 5. The shorelines of the latter two converge, whereas scenario 3 shows an abrupt advance of the shoreline at day 5, with an attempt to return back to its stable state of $x = 30$ m. The seabed has been flattened, the sandbar has been destroyed and erosion can be observed at the coast (Grasso et al. 2011). This tendency to evolve towards an equilibrium state (Dean et al. 2004) is consistent with the choice of morphogenic and constant storm-like forcing conditions.

The comparisons made between the two 2-day simulations and the three 6-day simulations, in this quite limited configuration, also reveal the little influence heritage has on the morphodynamic response. Both scenarios 1 and 2 have a comparable cumulative incoming wave energy density $E_H = \frac{1}{16} \int_0^T \rho g H_0^2 dt$ of 0.0591 J m^{-2} . The resulting beach profiles evolve towards similar profiles (reduction of the sandbar, increase of elevation of the plateau, and erosion at the coast), despite two different forcing conditions. Similar conclusions can be drawn regarding the 6-day simulations, where the cumulative energy

density of all three is equal to 0.177 J m^{-2} .

6 **C**onclusions

OptiMorph shows potential as a fast, robust, and low complexity morphodynamic model involving only two hyper-parameters. Despite using a basic hydrodynamic model for the description of the complex coupling of hydrodynamic and morphodynamic processes, we can nevertheless observe that a numerical model based on an optimization theory works effectively, with comparable results to a state of the art hydro-morphodynamic model requiring the tuning of dozens of hyper-parameters. Mid-term simulations also show typical morphodynamic behavior, with the tendency of the seabed to evolve towards an equilibrium state. Moreover, the results of the multi-1D code are very encouraging. These results demonstrate the tremendous potential of OptiMorph, a constrained energy minimization morphodynamic model.

Chapter key points

- A fast, robust, low-complexity morphodynamic model based on the minimization principle.
- Valid comparisons with well-known morphodynamic software such as XBeach.
- Results showing a long-term equilibrium solution.
- A numerical validation has been done (time, space and domain length consistency).
- An extension to Multi-1D has been tested on a case off Montpellier.
- A strong constraint in the choice of the hydrodynamic model which must be analytically differentiated.

CHAPTER

Hydrodynamic models

In the last chapter, we developed the OptiMorph model. In this version, the code only works with a local hydrodynamic model (shaoling). This hydrodynamic model has its limitations. In this chapter, we review the literature on existing hydrodynamic models. Next, we look at several types of hydrodynamic model, with a view to incorporating them into our OptiMorph code. These models are XBeach, SWAN, REF/DIF and Shallow-Water. Another model based on the shoaling criterion is developed, and these models are tested on well-known configurations, namely LIP-1C and open-sea simulations.

Current chapter contents

1	Introduction	66
2	Physics of Wave Motion	67
3	Linear Wave Theory	70
3.1	Celerity	72
3.2	Wave Energy	72
3.3	Shoaling Coefficient	73
3.4	Extended Shoaling Model	74
4	Wave Resolving	75
4.1	Context	76
4.2	Derivation of Shallow-Water Model	77
4.3	Shallow-Water Equations with Variable Bottom	78
4.4	REF/DIF Numerical Model	79
5	Spectral Wave Resolution	81
5.1	Context	81
5.2	Mathematical Background	82
5.3	Interesting Quantity	84
5.4	XBeach Numerical Model	85
5.4.1	Hydrodynamics	85
5.5	SWAN Numerical Model	87
5.6	Hydrodynamic Simulations	89
5.6.1	Flume Experiment	89
5.6.2	Open-Sea Simulation	90
6	Conclusion	91

1 **I**ntroduction

COASTAL HYDRODYNAMICS is an important field of research, as it has major implications for coastal engineering, oceanography, and marine ecology. In order to accurately simulate the complex physical processes that occur in coastal waters, numerical models have become increasingly important.

The governing equations that best represent the coastal hydrodynamic processes are the Navier-Stokes equations, which describe the conservation of momentum in a fluid. These equations can be used to model the flow of water in coastal regions, including wave propagation, sediment transport, and water circulation. In addition, the equations can be used to calculate the forces acting on the water, such as the Coriolis force and the drag force. However, these equations are far too complex and are therefore rarely used as is. Some rather heavy models like CROCO ([Marchesiello et al. 2021](#)) use the Reynolds-averaged Navier–Stokes equations (RANS) but most of the models are derivatives of the Navier-Stokes equations like the well-known Shallow-Water equation.

The first hydrodynamic models were based on linear wave theory ([Dean et al. 2004](#)). It is a concept that describes waves thanks to certain quantities: the dispersion, the group velocity and the wave action conservation. It has the advantage of giving an explicit general solution.

Nowadays, we have two major analyses: wave-wave analysis and spectral analysis. Wave-wave analysis is particularly adapted for studies focusing on phenomena linked to celerity thresholds or surface-curve like the wave breaking. In the opposite the spectral approach is more adapted for wave forecasting. We will focus mainly on those with spectral resolution that allows us to obtain directly an averaged water height H name H_s or $H_{1/3}$. On a larger scale, we find oceanographic ocean modelling models that solve the primitive ocean equations with models like NEMO ([Breivik et al. 2015](#)) or ROMS ([Shchepetkin et al. 2005](#)).

The aim of this chapter is to highlight various hydrodynamic models so that they can be exported to the morphodynamic model in chapter 3. It begins with an overview of the physic of waves and a discussion of what type of model can be a relevant choice for our morphodynamic model. These hydrodynamic models must produce a significant height H_s as output. They can be temporal or spectral. In order to verify their validity, we will test them on a known benchmark, namely LIP11D (Table S.6) ([Roelvink et al. 1995](#)). We will also perform simulations on open sea configurations on linear, convex and concave seabed. These will also be performed in the next chapter 3 on morphodynamic aspects.

2 Physics of Wave Motion

This section introduces the mathematical background of wave fluid motion analysis. With the figure below,

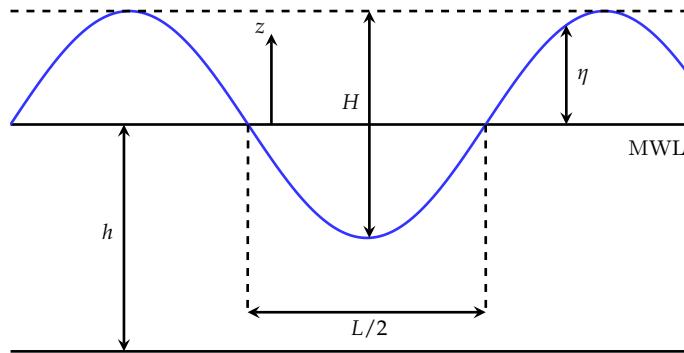


Figure 2.1 – Diagram of the linear theory

The variation of surface elevation with time, from the still water level, is denoted by η and given by:

$$\eta = \frac{H}{2} \cos \left(\frac{x}{L} - \frac{t}{T_0} \right) \quad (2.1)$$

with the hydrodynamic notations as follows:

- H the wave height (m),
- T_0 the wave period (s),
- L the wave length (m),
- θ the direction (rad),
- $k = 2\pi/L$ the wavenumber (m^{-1}),
- a the wave amplitude, $a = H/2$ (m),
- ka the wave slope (1),
- h the water depth (m),
- $\bar{\eta}$ the mean free surface level (m),
- $D = h + \bar{\eta}$ the local water depth (m).

We then introduce the Eulerian notations with the position with the horizontal vector having two components $\mathbf{x} = (x, y)$ and the vertical position z . Celerities are their respective temporal derivatives denoted $\mathbf{u} = (u, v)$ and w .

The equation that perfectly governs the motion of water and therefore of a wave is the Navier-Stockes equation:

$$\begin{cases} \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} + w \frac{\partial \mathbf{u}}{\partial z} = -\frac{1}{\rho_w} \nabla p + \nu \left(\nabla^2 \mathbf{u} + \frac{\partial^2 u}{\partial z^2} \right), \\ \frac{\partial w}{\partial t} + \mathbf{u} \cdot \nabla w + w \frac{\partial w}{\partial z} = -g - \frac{1}{\rho_w} \frac{\partial p}{\partial z} + \nu \left(\nabla^2 w + \frac{\partial^2 w}{\partial z^2} \right), \\ \nabla \cdot \mathbf{u} + \frac{\partial w}{\partial z} = 0, \end{cases} \quad \begin{aligned} (2.2a) \\ (2.2b) \\ (2.2c) \end{aligned}$$

with ρ_w the water density, ∇ the horizontal gradient, ν the viscosity, g the gravity constant, p the pressure, $\mathbf{u} = (u, w)$ is a velocity vector of x and y components. The equation (2.2a), (2.2b) are the momentum equations and the equation (2.2c) is the continuity equation. We could simplify these equations using some assumptions:

- The pressure is uniform $\implies -\frac{1}{\rho_w} \nabla p = 0$ and $\frac{\partial p}{\partial z} = 0$,
- The density is constant $\implies \rho_w = cte$,
- The bottom is horizontal $\implies \frac{\partial}{\partial x} = cte$,
- The fluid is incompressible $\implies \nabla \cdot \mathbf{v} = 0$,
- The fluid is non-viscous $\implies \nu = 0$,
- The fluid is irrotational $\implies \mathbf{u} = \nabla \phi$ and $w = \frac{\partial \phi}{\partial z}$.

By neglecting the viscosity, we obtain the Euler (1752) equations:

$$\begin{cases} \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} + w \frac{\partial \mathbf{u}}{\partial z} = -\frac{1}{\rho_w} \nabla p, \end{cases} \quad (2.3a)$$

$$\begin{cases} \frac{\partial w}{\partial t} + \mathbf{u} \cdot \nabla w + w \frac{\partial w}{\partial z} = -g - \frac{1}{\rho_w} \frac{\partial p}{\partial z}, \end{cases} \quad (2.3b)$$

$$\begin{cases} \nabla \cdot \mathbf{u} + \frac{\partial w}{\partial z} = 0, \end{cases} \quad (2.3c)$$

that can be express in vectorial form:

$$\frac{\partial \vec{V}}{\partial t} + (\vec{V} \cdot \nabla) \vec{V} = -\frac{1}{\rho_w} \nabla P - g \vec{k} \quad (2.4)$$

with $\vec{V} = (u, v, w)$ and P the vector of each pressure component in (x, y, z) and $\vec{k} = (0, 0, 1)$. Using vector calculus, we have:

$$\begin{aligned}\nabla(\vec{V} \cdot \vec{V}) &= 2\vec{V} \cdot \underbrace{(\vec{\nabla} \cdot \vec{V})}_{\text{incompressible}} + 2(\vec{V} \cdot \nabla)\vec{V}, \\ (\vec{V} \cdot \nabla)\vec{V} &= \frac{1}{2}\nabla(\vec{V} \cdot \vec{V}),\end{aligned}$$

and then, replacing the potential $\vec{V} = \nabla\phi'$, Euler (1752) equation becomes:

$$\frac{\partial \nabla\phi'}{\partial t} + \frac{1}{2}\nabla(\nabla\phi' \cdot \nabla\phi') = -\frac{1}{\rho_w}\nabla P - g\vec{k}, \quad (2.5)$$

however, we have: $\nabla(gz) = g\vec{k}$, then:

$$\nabla\left(\frac{\partial\phi'}{\partial t} + \frac{1}{2}(\nabla\phi' \cdot \nabla\phi')\right) = -\nabla\left(\frac{P}{\rho_w} - gz\right), \quad (2.6)$$

by integrating, making a change of variable between ϕ' and ϕ , we obtain the classical BERNOULLI energy equation:

$$\frac{\partial\phi}{\partial t} + \frac{1}{2}\left[|\nabla\phi|^2 + \left(\frac{\partial\phi^2}{\partial z}\right)\right] + \frac{p}{\rho_w} + g\eta = C(t), \quad z = \eta \quad (2.7)$$

with $C(t)$ a function depending on time which will be considered as null. Making the assumptions that $H \ll L$ and $H \ll h$ results in the linearized boundary conditions (in which the smaller, higher order and product terms are neglected). The resulting kinematic and dynamic boundary equations are then applied at the still water level, given by,

$$w = \frac{\partial\eta}{\partial t}, \quad g\eta + \frac{\partial\phi}{\partial t}, \quad z = 0. \quad (2.8)$$

The resulting solution for ϕ is given by:

$$\phi = -g \int \eta(t) dt \quad (2.9)$$

which gives us with the equation of η (2.22a):

$$\phi = -gH \frac{T_0}{4\pi} \frac{\cosh\left[\left(\frac{2\pi}{L}\right)(h+z)\right]}{\cosh\left[\left(\frac{2\pi}{L}\right)h\right]} \sin\left(\frac{2\pi x}{L} - \frac{2\pi t}{T_0}\right). \quad (2.10)$$

We can state that the condition of continuous surface celerities is

$$w = \frac{\partial \phi}{\partial z} = \mathbf{u} \cdot \nabla \eta + \frac{\partial \eta}{\partial t} = \nabla \phi \cdot \nabla \eta + \frac{\partial \eta}{\partial t}, \quad \text{for } z = \eta \quad (2.11)$$

and that (2.2c) becomes equivalent to the Laplace equation for ϕ :

$$\nabla \cdot \mathbf{u} + \frac{\partial w}{\partial z} = \nabla^2 \phi + \frac{\partial^2 \phi}{\partial z^2} = 0, \quad \text{for } -h \leq z \leq \eta \quad (2.12)$$

and for a horizontal bottom the equation of continuous vertical velocity is

$$w = \frac{\partial \phi}{\partial z} = 0, \quad \text{for } z = -h. \quad (2.13)$$

Additionally, after a few manipulations on (2.3a) and (2.3b) we obtain the beginning of a wave equation:

$$\frac{\partial^2 \phi}{\partial t^2} + g \frac{\partial \phi}{\partial z} = g \nabla \phi \cdot \nabla \eta - \frac{1}{2} \frac{\partial \eta}{\partial t} \frac{\partial^2 \phi}{\partial z \partial t} - \left(\frac{\partial}{\partial t} + \frac{\partial \eta}{\partial t} \frac{\partial}{\partial z} \right) \left[\nabla \phi \cdot \nabla \phi + \left(\frac{\partial \phi}{\partial z} \right)^2 \right] + C(t), \quad \text{for } z = \eta \quad (2.14)$$

3 Linear Wave Theory

The notion of uniform motion (linear motion of a wave) must respect certain assumptions: the wave slope is small ($ka \ll 1$), the quantity a/D is small too: ($a/D \ll 1$). These are small-amplitude waves.

Regarding the small-amplitude wave assumptions: ($ka \ll 1$) and ($a/D \ll 1$). We know that it has been proved that the non-linear term of equation (2.14) can be ignored and that from a first order Taylor development to get equation (2.14) for $z = 0$ instead of $z = \eta$, the linearized wave equation is now:

$$\frac{\partial^2 \phi}{\partial t^2} + g \frac{\partial \phi}{\partial z} = 0, \quad \text{for } z = 0. \quad (2.15)$$

From a Fourier decomposition, then replacing the solution into the Laplace equation (2.12) and taking into account the bottom boundary condition, we obtain the relation

$$\frac{\partial^2 \Phi}{\partial t^2} + gk \tanh(kD) \Phi = 0 \quad (2.16)$$

with the solution:

$$\Phi(t) = \mathcal{R}(\Phi_k e^{-i\sigma t}) \quad (2.17)$$

which give the dispersion relation given by Laplace,

$$\sigma^2 = gk \tanh(kD) \quad (2.18)$$

with the wave length L , wave period T_0 , $\sigma = 2\pi/T_0$ and $k = 2\pi/L$. By replacing, we obtained:

$$\left(\frac{2\pi}{T_0}\right)^2 = g \frac{2\pi}{L} \tanh kh. \quad (2.19)$$

We introduce elevation surface phase as:

$$\Theta = \mathbf{k} \cdot \mathbf{x} - \sigma t + \Theta_0 \quad (2.20)$$

with $0 \leq \Theta_0 \leq 2\pi$ the phase shift and a the amplitude

$$a = i \frac{\sigma}{g} \Phi_k \quad (2.21)$$

the solution is of [Airy \(1845\)](#) wave for free surface elevation are:

$$\eta = a \cos \Theta, \quad (2.22a)$$

$$\mathbf{u} = a \frac{\mathbf{k}}{k} \sigma \frac{\cosh(kz + kh)}{\sinh(kD)} \cos \Theta, \quad (2.22b)$$

$$w = \sigma \frac{\cosh(kz + kh)}{\sinh(kD)} \sin \Theta, \quad (2.22c)$$

$$p = \bar{p}^H + \rho_w g a \frac{\cosh(kz + kh)}{\cosh(kD)} \cos \Theta, \quad (2.22d)$$

where the mean hydrostatic pressure $\bar{p}^H = -\rho_w g(z - \bar{\eta}) + \bar{p}_a$ with p_a being the atmospheric pressure. These is the linear approach of the wave propagation.

After Airy's Works, [Stokes \(1847\)](#) extended the Airy's solution to take into account the neglected non-linear terms in equation (2.14). Even if the latter improves the fit with actual observations of waves, the Airy's solution is a reliable approximation for deep-bottom wave propagation which is almost irrotational, without being so far from the reality to the coast on wave-breaking zones.

In order to demonstrate shoaling we must make two assumptions:

- Period is the variant of propagation and remains constant as depth changes;

- The energy remains constant until it reaches breaking up.

3.1 Celerity

Substituting this solution for ϕ equation (2.10) into the two linearized surface boundary conditions yields the surface profile given in Equation (2.22a) and the wave celerity C given by:

$$C = \frac{gT_0}{2\pi} \tanh\left(\frac{2\pi h}{L}\right) = C_0 \tanh(kh) \quad (2.23)$$

where $C_0 = g/\omega$, $\omega = 2\pi/T_0$ and $k = 2\pi/L$.

3.2 Wave Energy

We have the potential and kinetic energy called E_i with the integration and L : the wave length:

$$E_i = \frac{1}{L} \left(2 \int_0^L (\rho g \eta dx) \frac{\eta}{2} \right) = \frac{1}{16} \rho g H^2 \quad (2.24)$$

which gives us both the formulation:

- Potential energy: $E_p = \frac{1}{16} \rho g H^2$
- Kinetic energy: $E_c = \frac{1}{16} \rho g H^2$

And then the total wave energy:

$$\mathcal{E}_H = \frac{1}{8} \rho g H^2 \quad (2.25)$$

One might expect that wave power (or the rate of transmission of wave energy) would be equal to wave energy times the wave celerity. This is incorrect, and the derivation of the equation for wave power leads to an interesting result which is of considerable importance. Wave energy is transmitted by individual particles which possess potential, kinetic and pressure energy. Summing these energies and multiplying by the particle velocity in the x-direction for all particles in the wave gives the rate of transmission of wave energy or wave power (P), and leads to the result (for an [Airy \(1845\)](#) wave)

$$P = \frac{\rho g H^2}{8} \frac{C}{2} \left(1 + \frac{2kh}{\sinh 2kh} \right) = EC_g \quad (2.26)$$

where C_g is the group wave celerity, given by

$$C_g = \frac{C}{2} \left(1 + \frac{2kh}{\sinh 2kh} \right). \quad (2.27)$$

In deep-water ($h/L > 0.5$) the group wave velocity $C_g = C/2$, and in shallow-water $C_g = C$. Hence, in deep-water wave energy is transmitted forward at only half the wave celerity.

3.3 Shoaling Coefficient

Therefore, we can say that the offshore flow of energy will be constant in its propagation until it comes to breaking near shore. We call \mathcal{E}_{F0} the offshore flow and \mathcal{E}_{Fh} the flow relative to depth h . Thus:

$$\mathcal{E}_{F0} = \mathcal{E}_{Fh}$$

and thus

$$\mathcal{E}C_{g0} = \mathcal{E}C_g$$

Where: C_{g0} is the group velocity offshore and C_g is the group velocity relative to the depth H . Considering that $C_{g0} \neq C_g$, we have with (2.25):

$$\rho g \frac{H_0^2}{8} \cdot C_{g0} = \rho g \frac{H^2}{8} \cdot C_g \rightarrow \frac{H^2}{H_0^2} = \frac{C_{g0}}{C_g}$$

This ratio is called the shoaling coefficient:

$$K_s^2 = \frac{H^2}{H_0^2} = \frac{C_{g0}}{C_g} \rightarrow K_s = \sqrt{\frac{C_{g0}}{C_g}}.$$

Thanks to the equations (2.27), (2.23) and $C_{g0} \xrightarrow[h \rightarrow +\infty]{} \frac{C_0}{2}$, we deduce:

$$K_s = \left[\tanh(kh) \left(1 + \frac{2kh}{\sinh(2kh)} \right) \right]^{-1/2}. \quad (2.28)$$

This coefficient allows us to use the Shoaling model (1.2a) in a very simplistic approach of our model. This model will be improved in the extended version of the following part 3.4.

3.4 Extended Shoaling Model

The Shoaling model 1.7 did not succeed to model wave breaking with wave periods $T_0 > 2\text{ s}$. This model was therefore improved to give birth to the extended model below:

Extended Shoaling model

$$H(x, t) = \begin{cases} H_0(x, t)K_S(x, t) & \text{for } x \in \Omega_S \\ \mathcal{F}(\gamma h(x, t)) & \text{for } x \in \Omega_B \end{cases} \quad (2.29a)$$

$$(2.29b)$$

where \mathcal{F} is a numerical parameterization function of the breaking defines below (5.3):

$$\mathcal{F}(\gamma h(x, t)) = H(x_{start}) + [H(x_{stop}) - H(x_{start})] \cdot f\left(\frac{x - x_{start}}{x_{stop} - x_{start}}\right) \cdot g\left(\frac{h_{max} - h}{h_{max} - h_{min}}\right) \quad (2.30)$$

with $x \in \Omega_B = [x_{start}, x_{stop}]$, $h \in [h_{min}, h_{max}]$ and the following notations:

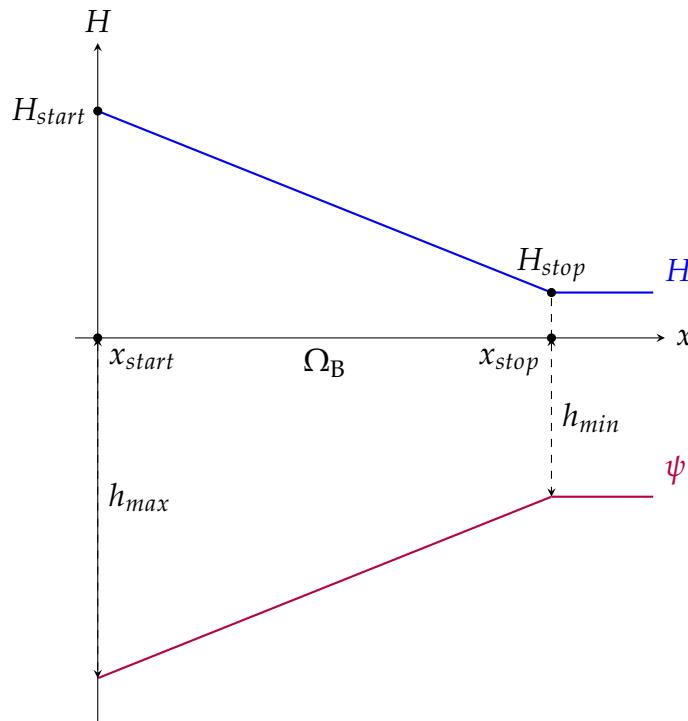


Figure 2.2 – Illustration of notations.

H_{start} and H_{stop} are the water heights at the beginning and end of the breaking on the

domain $\Omega_B = [x_{start}, x_{stop}]$. The first function f gives an account of breaking without taking into account the bed shape. It simply gives the appearance of breaking. The second function g takes into account the seabed and interacts with it. Note that if f and g are the affine functions $x \mapsto x$, we find the breaking $\gamma h(x, t)$ illustrated on figure 5.2. We can present below (figure 5.3) some of these functions that set the breaking:

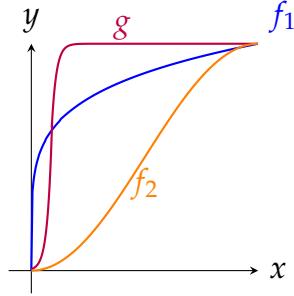


Figure 2.3 – Illustration of f_1 , f_2 and g defined in $[0, 1] \rightarrow [0, 1]$.

These functions were chosen to try to capture a natural breaking. They have no physical meaning.

It is necessary to stipulate that the model will first locate all the Ω_B domains and then apply the equation (5.3) on each of them.

This type of model gives us the following breaking for a simulation with a wave period $T_0 = 10$ s and an offshore water height of $H_0 = 2$ m.

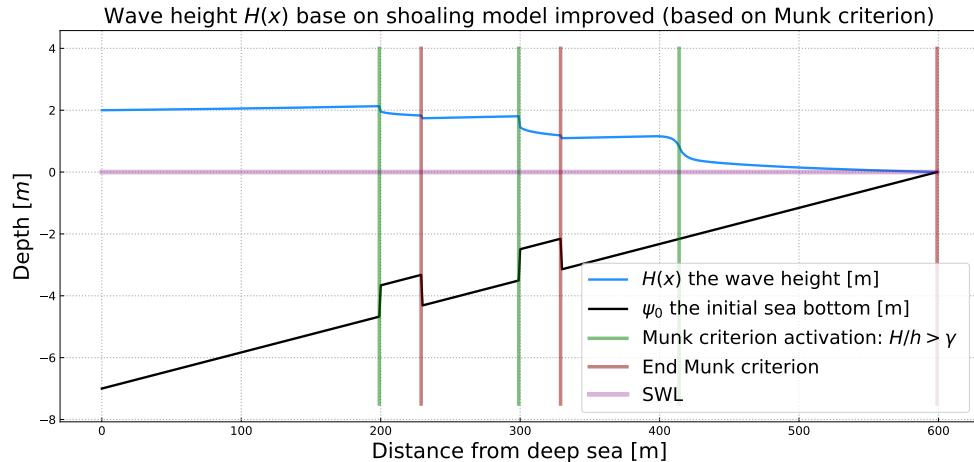


Figure 2.4 – Wave height $H(x)$ with the Shoaling model with the last improve (using f and g) for a configuration with a linear sea bottom of slopes about 0.11, wave period $T_0 = 10$ s and offshore water height of $H_0 = 2$ m.

This breaking uses the function $f : x \mapsto x^{\frac{1}{5}}$ but it can be changed to the desired shape.

4 Wave Resolving

4.1 Context

The alternative method of resolving phases for shallow water waves often relies on Boussinesq-type (BT) equations, with notable contributors including ([Peregrine 1967](#); [Freilich et al. 1984](#); [Madsen et al. 1992](#); [Liu et al. 2002](#)). The mild-slope equation, as proposed by [Berkhoff \(1972\)](#), is also commonly used as a port agitation model. It can be used to isolate the eigen modes of vibration within it. These models aim to reconstruct the sea surface's characteristics in both space and time, encompassing horizontal and vertical flow velocity. They inherently account for phenomena like refraction, diffraction, and, in some instances, triad and quadruplet wave interactions. Dissipation processes, such as bottom friction and depth-induced wave breaking, can be parameterized in these models. However, they do not incorporate wind-induced wave generation.

Despite these models' effectiveness in capturing vital coastal and nearshore wave phenomena, numerical models based on BT equations have grown indispensable in the field of coastal engineering, as highlighted by [Rusu et al. \(2012\)](#). The standard BT equations for variable water depths were initially formulated by [Peregrine \(1967\)](#), though they were limited to addressing weakly non-linear and weakly dispersive shallow water flows. To address this limitation, [Madsen et al. \(1991\)](#) and [Nwogu \(1993\)](#) expanded upon the standard BT equations, creating a practical tool for simulating the non-linear transformation of irregular, multi-directional waves in varying water depths before wave breaking occurs.

The most advanced BT models are versatile, applicable in both deep and shallow waters, as well as when dealing with highly non-linear waves. However, one limitation of traditional BT models is their inability to accurately describe the overturning and turbulence generation during wave breaking. Researchers have consequently developed semi-empirical approaches to address these challenges.

Widely used in the coastal community, depth-integrate models are very popular because ; i) They are a simpler set of equations than the full three-dimensional ones, and so allow for a much more straightforward analysis of sometimes complex problems ; ii) In spite of their simplicity, the equations provide a reasonably realistic representation of a variety of phenomena in atmospheric and oceanic dynamics. As these models are vertically integrated, it is assumed that they cannot model transfers between horizontal layers. Some models, such as the Shallow-Water (SW) model ([Marche et al. 2007](#)), neglect dispersion

$$\mu = \frac{H_0^2}{\lambda^2} \ll 1, \quad (2.31)$$

and remain easy to solve numerically. Other more advanced models, such as Green-Naghdi (GN), Serre-Green-Naghdiand (SGN) KdV ([Kim et al. 2001](#)), take dissipation into account but are numerically difficult to solve.

In our work, we'll be using the most popular model, the Shallow-Water model ([Marche](#)

et al. 2007) and the REF/DIF model (James Thornton Kirby et al. 1994).

4.2 Derivation of Shallow-Water Model

Taking back the equation Euler equation (2.4), we consider the 1D Euler's equations without surface tension:

$$\left\{ \begin{array}{l} \text{free surface condition : } p = 0, \quad \frac{Dh}{Dt} = \frac{\partial h}{\partial t} + u \frac{\partial h}{\partial x} = w, \quad \text{on } z = h(x, y, t) \\ \text{momentum equation : } \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + \frac{1}{\rho_w} \frac{\partial P}{\partial x} + gz = 0 \\ \text{continuity equation : } \frac{\partial u}{\partial x} = 0, \\ \text{bottom boundary condition : } u \frac{\partial z}{\partial x} = 0, \quad \text{on } z = 0. \end{array} \right. \begin{array}{l} (2.32a) \\ (2.32b) \\ (2.32c) \\ (2.32d) \end{array}$$

Here, p is the pressure, h the vertical displacement of the free surface, u the x velocity, ρ_w the density, g the acceleration due to gravity. (Figure 2.5).

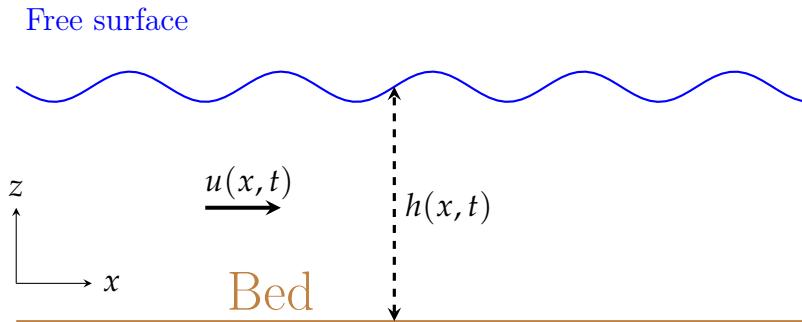


Figure 2.5 – Shallow-Water diagram for a 1D flow.

For the first step of the derivation of the shallow-water equations, we consider the global mass conservation. We integrate the continuity equation (2.32c) vertically as follows,

$$0 = \int_0^h \frac{\partial u}{\partial x} dz \quad (2.33a)$$

$$= \frac{\partial}{\partial x} \int_0^h u dz - u|_{z=h} \frac{\partial h}{\partial x}, \quad (2.33b)$$

where the bottom boundary condition (2.32d) was used in the fourth row. With the surface condition (2.32a), equation (2.33b) becomes

$$\frac{\partial h}{\partial t} + \frac{\partial}{\partial x} \int_0^h u \, dz = 0 \quad (2.34)$$

In the following stage, we proceed with the long-wave approximation, where we consider the wavelength to be significantly greater than the fluid's depth. It's important to note that we do not make any assumptions regarding the perturbations of having small amplitudes, ensuring that non-linear terms are accounted for rather than being disregarded. By employing the long-wave approximation, we can derive the hydrostatic pressure by performing the integration of the vertical component of the momentum equation.

$$\int_z^h \frac{\partial p}{\partial z} \, dz = - \int_z^h \rho g \, dz \quad (2.35a)$$

$$p(x, h, t) - p(x, z, t) = -\rho g(h(x, t)) - z \quad (2.35b)$$

$$p(x, z, t) = \rho g(h(x, t) - z). \quad (2.35c)$$

We applied to the surface condition represented as $p(x, h, t) = 0$. Utilizing this equation for the hydrostatic pressure (2.35c) and additionally supposing that there are no vertical variations in the parameter u (\Rightarrow equation (2.32a) = 0), the horizontal momentum equations of the shallow-water system can be derived in the following manner:

$$\partial_t u + u \partial_x u + g \partial_x h = 0. \quad (2.36)$$

The conservation of mass given by (2.34) becomes

$$\partial_t h + \partial_x(hu) = 0. \quad (2.37)$$

Then, equations (2.36) and (2.37) are the shallow-water equations. More details on the derivation are available on ([Fabien Marche 2007](#)).

4.3 Shallow-Water Equations with Variable Bottom

Previously, we presented the flat-bottomed Shallow-Water model. However, our model requires a variable-bottom hydrodynamic model. Indeed, our model runs a large number of simulations with different geometries before completing. It is therefore essential that the model can run a hydrodynamic simulation on any bottom. A variable bottom on the Shallow-Water model adds an additional source term and complexity to the numerical implementation. The hyperbolic system of equations (2.36) and (2.37) thus becomes:

$$\partial_t h + \partial_x q = 0 \quad (2.38a)$$

$$\partial_t q + \partial_x \left(\frac{q^2}{h} + \frac{1}{2} gh^2 \right) = -gh\partial_x Z \quad (2.38b)$$

$$+ \text{Boundary condition}, \quad (2.38c)$$

with h the water height, u the velocity and $q = hu$ the flow. We use the notation $d_i = \frac{\partial}{\partial i}$. These quantities are presented in figure 2.6.

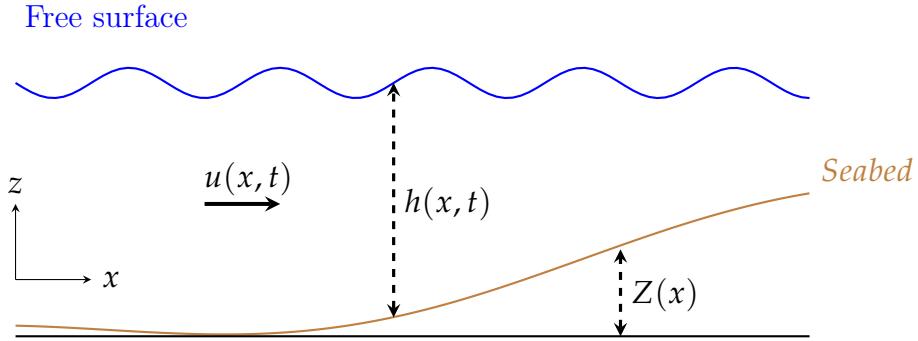


Figure 2.6 – Shallow-Water diagram for a 1D flow and variable bottom.

The equation (2.38a) represents the continuity. The second (2.38b) is the moment and the third the boundary conditions. Generally, the resolution of these equations is done using the finite volume method. Although the addition of source terms (variable background (Berthon et al. 2018; Marche et al. 2007), Exner coupling (Serrano-Pacheco et al. 2012), ...) makes the resolution more complex, the resolutions remain mainly on this principle. In our case, we use finite volume solving with a VFRoe numerical scheme from (Marche et al. 2007).

4.4 REF/DIF Numerical Model

REF/DIF is a phase-resolving parabolic refraction-diffraction model for ocean surface wave propagation. It was originally developed by JIM KIRBY and TONY DALRYMPLE starting in 1982, based on KIRBY's dissertation work. This work led to the development of REF/DIF 1, a monochromatic wave model (James Thornton Kirby et al. 1994).

This model is solving the Berkhoff (1972) equation, known as the mild slope equation. It is written in terms of the surface displacement, $\eta(x, y)$. The equation, in terms of horizontal gradient operator, is

$$\nabla_h \cdot (CC_g \nabla_h \eta) + \sigma^2 \frac{C_g}{C} \eta = 0 \quad (2.39)$$

where C and C_g are defined in (2.27) and (2.23).

In contrast to the mild slope model which is valid for varying bathymetry, researchers in the area of wave diffraction were developing models for constant bottom applications. For example, [Mei et al. \(1980\)](#) developed a simple parabolic equation for wave diffraction and applied it to the diffraction of waves by a slender island. Their equation is

$$\frac{\partial A}{\partial x} = \frac{i}{2k} \frac{\partial^2 A}{\partial y^2}$$

where A is a complex amplitude related to the water surface displacement by

$$\eta = Ae^{i(kx-\sigma t)}.$$

[Yue et al. \(1980\)](#), using a multiple-scale approach, developed a nonlinear form of this equation, which accurately predicts the propagation of a third order Stokes wave.

The parabolic model of REF/DIF, described below, combines the essential features of the two approaches described above. The variable depth features of the mild-slope equation (along with extensions to include effects of wave-current interaction) are retained, but the model is developed in parabolic form and in terms of a complex amplitude A .

$$(C_g + U) A_x + V A_y + i(\bar{k} - k) (C_g + U) A + \frac{\sigma}{2} \left\{ \left(\frac{C_g + U}{\sigma} \right)_x + \left(\frac{V}{\sigma} \right)_y \right\} A - \frac{i}{2\sigma} \left((p - V^2) A_y \right)_y - \sigma \frac{k^2}{2} D |A|^2 A = 0$$

where $p = CC_g$ and \bar{k} = reference wave number, taken as the average wave number along the y axis, and U is the mean current velocity in the x coordinate direction and V is in the y direction. The nonlinear term includes D , which is

$$D = \frac{(\cosh 4kh + 8 - 2 \tanh^2 kh)}{8 \sinh^4 kh}$$

Finally, [James T Kirby \(1986a\)](#) derived the above equation for a wide-angle parabolic approximation, which allows the study of waves with larger angles of incidence with respect to the x axis. This more precise equation was extended ([James T. Kirby 1986b](#)) and used in REF/DIF 1 ([James Thornton Kirby et al. 1994](#)).

Note : The mild slope equation, excluding additional terms for bed slope provides accurate results for the wave field over bed slopes ranging from 0 to approximately 1/3 ([N. Booij](#)

1983). This must be taken into account when using the REF/DIF model.

5 Spectral Wave Resolution

5.1 Context

We define a wave has a height H , a period T , a direction θ , and so forth. modelling waves with statistical is realistic because it has been shown that the N individual wave heights H_1, H_2, \dots, H_N of a certain time series follows a Rayleigh distribution (Figure 2.7), expressed here with its survival

$$P(H > h) = e^{-(h/H_{\text{rms}})^2} \quad (2.40)$$

where $H_{\text{rms}} = \sqrt{\frac{1}{N} \sum_{i=1}^N H_i^2}$ and the H_i denote the individual wave height in a certain time series. The Rayleigh distribution is generally suitable for commonly observed waves. As soon as the waves are quite high, the distribution of Tayfun (1980) must be considered. Tayfun distribution is more realistic since most of the non-linear waves effects are taken into account.

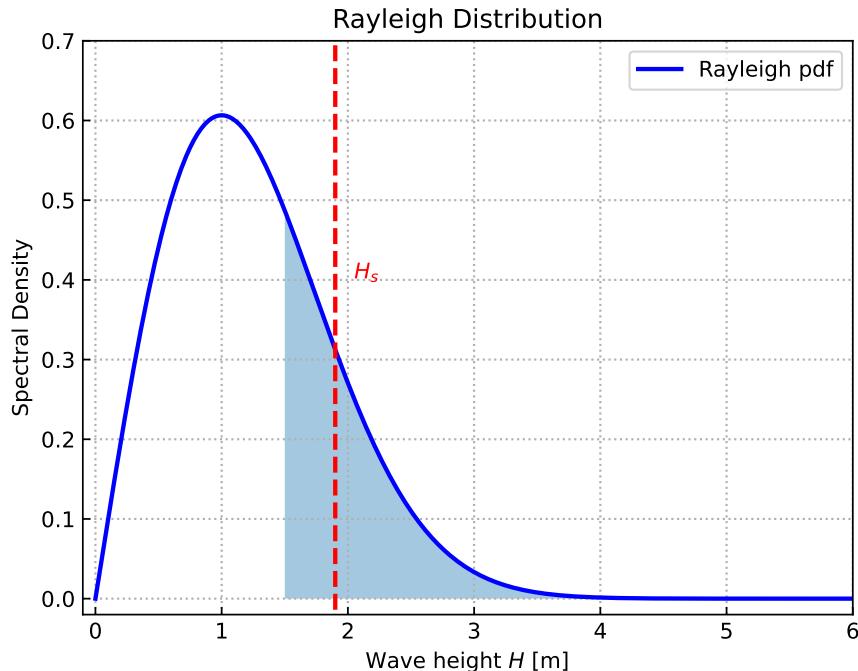


Figure 2.7 – Rayleigh density function $f(H, 1.9)$. The red area is the third-highest observed waves in a certain time series. The dashed line represents $H_{1/3}$, the mean of the third-highest observed waves also assimilated to the significant wave height H_s .

With that kind of distribution, a common variable named the significant wave height

H_s is determined. From wave-wave analysis, H_s is defined as $H_{1/3}$: the mean of the third-highest waves of the time series. It corresponds to a sea state in a stationary condition. Another variable used is H_{\max} which is the maximum wave height observed, it is highly depending on the length of the time series.

5.2 Mathematical Background

An important problem occurs in the analysis of the wave motion when the wave has a length greater than L , which requires the treatment of irregularities. One can use the statistical analyses developed by Fourier (FFT) ([Nussbaumer et al. 1982](#)) because the waves are supposed to satisfy the linear wave theory (locally). Indeed, the water level signal can be decomposed into superimposed sinusoidal waves. And thus be decomposed into Fourier series in the form of a given direction α (for simplicity's sake):

$$\eta(x, t) = \sum_{n=0}^{\infty} \eta_n(x) = \sum_{n=0}^{\infty} A_n \cos(k_n x + \phi_n) \quad (2.41)$$

where A_n denote Fourier's amplitude is associated with a particular frequency k_n and a particular phase shift ϕ_n . The variance of a sinusoidal surface is the average over one wavelength L of the surface elevation squared (assuming that the mean surface is at zero):

$$V(\eta_1) = \frac{1}{L} \int_0^L \left[A \sin\left(\frac{2\pi x}{L}\right) \right]^2 dx = \frac{1}{L} \int_0^L A^2 \left[\frac{1}{2} - \cos\left(\frac{4\pi x}{L}\right) \right] dx = \frac{1}{2} A^2 \quad (2.42)$$

Thus the energy per unit area of a sinusoidal wave can also be written in terms of its variance:

$$\frac{\text{energy}}{\text{area}} = \rho g V(\eta_1). \quad (2.43)$$

We have therefore with (2.42) and (2.41),

$$V(\eta) = \sum_{n=0}^{\infty} \frac{1}{2} A_n^2. \quad (2.44)$$

Now let Δk_n be a frequency interval centered on frequency k_n , whose sinusoidal has amplitude A_n . We then define

$$E(k = k_n) \equiv \lim_{\Delta k_n \rightarrow 0} \frac{\frac{1}{2} A_n^2}{\Delta k_n}. \quad (2.45)$$

In this definition, keep in mind that each A_n is associated with a particular frequency k_n , and that the limit operation holds for each value of n . We are thus defining a function

of the spatial frequency, which becomes a as the bandwidth Δk_n goes to zero. The continuous function $E(k)$ is called the omnidirectional elevation variance spectrum. “Omnidirectional” means that there is no reference direction (e.g., a direction of wave propagation relative to the wind direction) included in the quantity. Equations (2.44) and (2.45) show that integrating the omnidirectional variance spectrum over all frequencies gives the total elevation variance:

$$V(\eta) = \langle \eta^2 \rangle = \int_0^\infty E(k) dk \quad (2.46)$$

which gives us in multi-directional, the well known formula

$$E = \int_0^\infty \int_0^{2\pi} E(k, \theta) dk d\theta \quad (2.47)$$

of the wavenumber-direction spectrum. Other expressions of the wave spectrum are also used. In particular, wave lengths and wave frequencies are interrelated via the dispersion equation (2.18) and

$$E(k, \theta) dk d\theta = E(f, \theta) df d\theta. \quad (2.48)$$

The wave density spectrum (i.e. right side of (2.18)) defines the repartition of the wave energy ² along frequencies and direction. Unlike the signal of the free surface elevation, the density spectrum is relatively regular and allows compressing the information of the full signal.

As recalled in (Tolman et al. 2009), without current E is a conserved quantity. In case of current, spectral component is no longer conserved due to the work done by current on the mean momentum transfer of waves. Hence, ones are also interested in using the wave action spectrum A as

$$A(k, \theta) = \frac{E(k, \theta)}{\sigma} \quad (2.49)$$

which is conserved whatever the case (G. á. Whitham 1965). Thanks to that definition, we can express the wave propagation as

$$\frac{DA}{Dt} = \frac{S}{\sigma} \quad (2.50)$$

where A is the action wave spectrum, D/Dt is here the total derivative and S is the net effect of sources and sinks for the wave spectrum E . Since left part of (2.50) considers linear wave propagation as presented before, any perturbing effects are gathered in the expression of S . In the next subsection we identify physical phenomena in charge of the balance of (2.50).

5.3 Interesting Quantity

The first quantity of interest is generally the significant wave height H_s . From the spectrum we define

$$H_{m0} = 4E^{1/2} = 4 \left[\int_0^\infty \int_0^{2\pi} E(f, \theta) df d\theta \right]^{1/2}. \quad (2.51)$$

In practice $H_{1/3} \simeq H_{m0}$ ([Longuet-Higgins 1963](#)). H_{m0} is therefore the spectral representation of the significant wave height. The denotation $m0$ stands for the zero moment of power spectrum, which is more generally defined for the order p as

$$m_p = \int_0^\infty \int_0^{2\pi} f^p E(f, \theta) df d\theta \quad (2.52)$$

Several other quantities often used in ocean engineering derive from the spectrum. In particular, f_p is the peak frequency, with $E(f_p) = E_{\max}$ and the peak period $T_p = 1/f_p$. Other famous periods $T_{m0,1}$, $T_{m0,2}$ and $T_{m0,-1}$ stem from the period of order p defined as

$$T_{m0,p} = \left[\frac{\int_0^{f_{\max}} \int_0^{2\pi} f^p E(f, \theta) df d\theta}{\int_0^{f_{\max}} E(f) df} \right]^{-1/p} \quad (2.53)$$

with f_{\max} the highest frequency observed. Finally, if we define

$$a_1(f) = \int_0^{2\pi} E(f, \theta) \cos \theta d\theta / \int_0^{2\pi} E(f, \theta) d\theta, \quad (2.54a)$$

$$b_1(f) = \int_0^{2\pi} E(f, \theta) \sin \theta d\theta / \int_0^{2\pi} E(f, \theta) d\theta, \quad (2.54b)$$

then the mean wave direction for the frequency f is

$$\theta_m(f) = \arctan \left(\frac{b_1(f)}{a_1(f)} \right). \quad (2.55)$$

In particular, $\theta_m(f_p)$ is the main wave direction (or peak wave direction). Ones are also interested in the mean wave direction θ_M defined by integrating over the direction as

$$\theta_M = \arctan \left(\frac{\int_0^\infty b_1(f) df}{\int_0^\infty a_1(f) df} \right). \quad (2.56)$$

To reconstruct the signal from a statistical approach as detailed here is valid in mostly all applications. However a wave-wave analysis would be preferred for applications when the phases of waves are of first interest, such as in the breaking zone.

5.4 XBeach Numerical Model

The XBeach model is a process-based model developed by the Delft University of Technology. It is a two-dimensional, depth-integrated numerical model that simulates the hydrodynamics, sediment transport, and morphological changes of coastal systems. XBeach is a flexible model that can be used to simulate a variety of coastal processes, including wave breaking, bedload transport, and nearshore morphological changes. The model is based on the principles of conservation of mass, momentum, and energy and uses a finite-difference numerical scheme to solve the governing equations. XBeach has been widely used in coastal studies due to its flexibility and accuracy, and it has been applied to a wide range of coastal systems, including estuaries, beaches, and coastal wetlands. The model can be used as a profile model in 1D ([Pender et al. 2013](#)), or as an area model in 2D ([McCall et al. 2010](#)), and today, there are three modes in which the hydrodynamics can be resolved in XBeach, being:

- **Stationary** – All wave group variations, and thereby all infragravity motions, are neglected, and only the mean motions are included. This type can be applied for modelling morphological changes under moderate wave conditions;
- **Surfbeat** – This in-stationary, hydrostatic mode, is wave group resolving, and is hence also applicable when one is interested in the swash zone processes;
- **Non-hydrostatic** – The non-linear Shallow-Water equations are solved, and hence individual short-wave propagation and transformation is resolved.

In our case, we will focus on the **Stationary** mode.

5.4.1 Hydrodynamics

The wave action balance is solved to obtain the wave forcing:

$$\frac{\partial A}{\partial t} + \frac{\partial c_x A}{\partial x} + \frac{\partial c_y A}{\partial y} + \frac{\partial c_\theta A}{\partial \theta} = -\frac{D_w}{\sigma} \quad (2.57)$$

Where A is the wave action, C the wave propagation speed (where the subscripts refer to the x - and y -directions, and θ -space), θ is the angle of incidence, D_w the wave

energy dissipation per directional bin and σ the intrinsic wave frequency. The wave action as above (5.5) by:

$$A(x, y, t, \theta) = \frac{S_w(x, y, t, \theta)}{\sigma(x, y, t)} \quad (2.58)$$

In which the S_w is the wave energy density per directional bin. The total wave energy E_H is obtained by integration of the wave energy density S_w over all directional bins:

$$E_H = \int_0^{2\pi} S_w(x, y, t, \theta) d\theta \quad (2.59)$$

The distribution of the total wave energy dissipation \bar{D}_w over all directional bins is calculated proportional to the energy density distribution as follows:

$$D_w(x, y, t, \theta) = \frac{S_w(x, y, t, \theta)}{E_w(x, y, t)} \bar{D}_w(x, y, t) \quad (2.60)$$

The total wave energy dissipation is calculated using a method described by [D. J. Roelvink \(1993\)](#) as the product of the dissipation per breaking event and the fraction of broken waves Q_b . The energy dissipation per wave breaking event is assumed to take place over half of the representative wave period T_{rep} , resulting in the following expression for the total, directionally integrated, wave energy dissipation:

$$\bar{D}_w = \alpha \frac{2}{T_{rep}} Q_b E_H \quad (2.61)$$

Where α is a calibration factor and E_w the total wave energy (Equation (5.6)). The fraction of breaking waves Q_b is estimated from a Rayleigh distribution ([Battjes et al. 1978](#)):

$$Q_b = 1 - \exp \left(- \left(\frac{H_{rms}}{H_{max}} \right)^n \right) \quad (2.62)$$

Where the root-mean-square wave height H_{rms} is calculated from the wave energy E_H , and the maximum wave height H_{max} is calculated using the breaker index γ (the ratio between the breaking wave height and the water depth, usually given the value 0.88).

$$E_H \sim \frac{1}{8} \rho g H_{rms}^2 \Rightarrow H_{rms} = \sqrt{\frac{8E_H}{\rho g}}, \quad H_{max} = \gamma_b h \quad (2.63)$$

This closes the set of equations for the wave action balance (Equation (5.4)). From the wave energy, the wave-induced radiation stresses can be determined using linear wave theory. Similar to the wave action balance, a roller balance is solved and coupled to the wave energy balance, where the wave energy dissipation forms a source of energy in

the roller balance. The roller-induced radiation stress is calculated and together with the wave-induced radiation stress they are used to calculate the wave forcing: The flows are calculated using a depth-averaged formulation of the Shallow-Water equations, taking into account wave-induced mass flux and return flows. This Generalized Lagrangian Mean (GLM) formulation uses Lagrangian velocities ([Andrews et al. 1978](#)):

$$\frac{\partial u^L}{\partial t} + u^L \frac{\partial u^L}{\partial x} + v^L \frac{\partial u^L}{\partial y} - fv^L - v_h \left(\frac{\partial^2 u^L}{\partial x^2} + \frac{\partial^2 u^L}{\partial y^2} \right) = \frac{T_{sx}}{\rho h} - \frac{T_{bx}^E}{\rho h} - g \frac{\partial \eta}{\partial x} + \frac{F_x}{\rho h} \quad (2.64a)$$

$$\frac{\partial v^L}{\partial t} + u^L \frac{\partial v^L}{\partial x} + v^L \frac{\partial v^L}{\partial y} + fu^L - v_h \left(\frac{\partial^2 v^L}{\partial x^2} + \frac{\partial^2 v^L}{\partial y^2} \right) = \frac{T_{sy}}{\rho h} - \frac{T_{by}^E}{\rho h} - g \frac{\partial \eta}{\partial y} + \frac{F_y}{\rho h} \quad (2.64b)$$

$$\frac{\partial \eta}{\partial t} + \frac{\partial u^L h}{\partial x} + \frac{\partial v^L h}{\partial y} = 0 \quad (2.64c)$$

Where the Lagrangian velocity components (denoted by the superscript L) are the superposition of the Eulerian velocity and the Stokes' drift velocity:

$$u^L = u^E + u^S \quad \text{and} \quad v^L = v^E + v^S. \quad (2.65)$$

In this thesis, we use version 1.23 of XBeach.

5.5 SWAN Numerical Model

SWAN, in its third version, is in stationary mode (optionally non-stationary) and is formulated in Cartesian or spherical coordinates. The unconditional numerical stability of the SWAN model makes its application more effective in shallow water. In SWAN, the waves are described with the two-dimensional spectrum of the wave action density A ,

$$A(x, y, \sigma, \theta) = \frac{E(x, y, \sigma, \theta)}{\sigma} \quad (2.66)$$

where x and y are the horizontal components of geographic space, σ is the relative frequency, θ is the wave direction, and E is the energy density.

The spectrum considered in the SWAN model is that of the wave action density $A(\sigma, \theta)$ rather than the spectrum of the energy density $E(\sigma, \theta)$. This is because, in the presence of currants for the reasons we mentioned above (non-conservation of E_H) ([G. B. Whitham 2011](#)). Because wave action propagates in both geographic and spectral space under the influence of genesis and dissipation terms, wave characteristics are described in terms of two-dimensional wave action density. The action density spectrum balance equation relating the propagation term to the effects of the source and sink terms, in Cartesian

coordinates, is (Hasselmann et al. 1973)

$$\frac{\partial A}{\partial t} + \frac{\partial (C_x A)}{\partial x} + \frac{\partial (C_y A)}{\partial y} + \frac{\partial (C_\sigma A)}{\partial \sigma} + \frac{\partial (C_\theta A)}{\partial \theta} = \frac{S}{\sigma}. \quad (2.67)$$

On the left-hand side of Equation (5.14), the first term represents the local temporal variation of the wave action density, the second and third terms represent the propagation of wave action in the geographical space of velocities C_x and C_y , the fourth term represents the shifting of the relative frequency due to variations in bathymetry (with propagation velocity C_σ) and currents (with propagation velocity C_θ), and the fifth term represents the refraction induced by the combined effects of depth and currents. $C_x, C_y, C_\sigma, C_\theta$ propagation velocities are obtained from linear wave theory. The term in the right-hand side of Equation (5.14) represents processes that generate, dissipate, or redistribute wave energy, and S can be expressed as (Lv et al. 2014)

$$S = S_{in} + S_{wc} + S_{brk} + S_{bot} + S_{n14} + S_{n13} \quad (2.68)$$

where S_{in} is the wind energy input. The dissipation terms of wave energy is represented by the contribution of three terms: the white capping S_{wc} , bottom friction S_{bot} , and depth induced breaking S_{brk} . S_{n14} and S_{n13} represent quadruplet interaction and triad interactions, respectively.

A finite difference scheme is used for each of the five dimensions: time, geographic space, and spectral space made the numerical implementation in SWAN effective. The following guidelines must be followed in order to obtain the discretization adopted at the SWAN model level for numerical computation:

1. time of a constant and identical time step Δt for the propagation term and the source term,
2. geographical space of a rectangular grid with constant spatial steps Δx and Δy ,
3. spectral space of a constant directional step $\Delta\theta$ and a constant relative frequency step $\Delta\sigma/\sigma$,
4. frequencies between a fixed minimum maximum values of 0.04 Hz and 1 Hz respectively,
5. the direction θ can also be delimited by the minimum and maximum values θ_{\min} and θ_{\max} (as an option).

In this thesis, we use version 41.31 of SWAN.

5.6 Hydrodynamic Simulations

In this section, we focus on two types of simulations. The simulations in the flume and those in the open sea.

5.6.1 Flume Experiment

For this section, we will concentrate on a reduced number of simulations. We will limit ourselves to simulate the hydrodynamics on two specific cases. The case LIP11D - 1C [S.6](#) which is a flume experiment. This case is part of the XBeach benchmark ([Roelvink et al. 1995](#)) and has allowed the validation of this one. Other models like CROCO ([Marchesiello et al. 2022](#)) based their validation on this benchmark. This experiment present hydrodynamic and morphodynamic data. For the second case, we choose a very simple bathymetry: linear. This one is the result of the linear interpolation on the bathymetry LIP11 1C. For these 2 cases, we perform hydrodynamic simulations with the SWAN model [5.5](#), XBeach model [5.4](#), an extended Shoaling model presented in [3.4](#) and Shallow-Water model presented in [4.3](#). The results are evaluated with the criterion H_{RMSE} which corresponds to the RMSE (root mean square) between experimental and numerical H .
In order to configure our models, we set the domain Ω over 180 m with a uniform subdivision of 180 cells. The incoming wave boundary condition is provided using a JONSWAP wave spectrum ([Hasselmann et al. 1973](#)), with a significant wave height of $H_s = 0.6$ m and a peak frequency at $f_p = 5 \text{ s}^{-1}$. The breaker model of XBeach uses the [D. J. Roelvink \(1993\)](#) formulation, with a breaker coefficient of $\gamma = 0.4$, a power $n = 15$, and a wave dissipation coefficient of 0.5. For the breaker model of SWAN and Shoaling, we simply use the breaker coefficient of $\gamma = 0.4$. The parameter files are present in the appendix [A.1](#). The results of LIP11D - 1C and its linear interpolation are in figures [2.8](#) and [2.9](#) below.

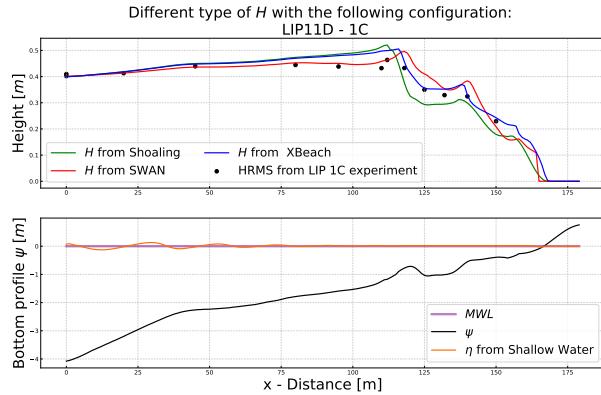


Figure 2.8 – Hydrodynamic results obtained with the Shoaling, SWAN, XBeach and Shallow-Water models. Bathymetric configurations from the LIP 1C channel experiment. Black points, measured HRMS, black bathymetry, green H from extended shoaling ($H_{RMSE} = 0.045$ m), red H from SWAN ($H_{RMSE} = 0.033$ m), blue H from XBeach ($H_{RMSE} = 0.028$ m) and orange from Shallow-Water model.

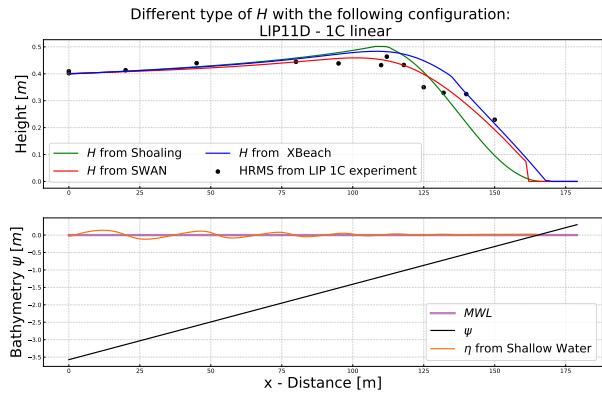


Figure 2.9 – Hydrodynamic results obtained with the Shoaling, SWAN, XBeach and Shallow-Water models. Bathymetric configurations from a linear interpolation of the LIP 1C channel experiment. Black bathymetry, green H from extended shoaling ($H_{RMSE} = 0.059$ m), red H from SWAN ($H_{RMSE} = 0.024$ m), blue H from XBeach ($H_{RMSE} = 0.043$ m) and orange from Shallow-Water model.

We notice that XBeach reproduces best the experimental results ($H_{RMSE} = 2.8$ cm). However, the 3 results are still very good and very realistic. For the linear profile, we have chosen to leave the experimental points of the previous experiment. The results are surprising because SWAN remains very close to these points ($H_{RMSE} = 2.4$ cm). We noticed with another case that the points were almost superimposed, whether in experimental or linear bathymetry.

5.6.2 Open-Sea Simulation

For this section, we perform realistic open-sea simulations. These simulations will allow us to introduce a bathymetry set that can be used in our morphodynamic model. We perform simulations on configurations such as linear, convex and concave bathymetry. For these 3 cases, we perform hydrodynamic simulations with the SWAN model 5.5, XBeach model 5.4, an extended Shoaling model presented in 3.4 and Shallow-Water model presented in 4.3.

In order to configure our models, we set the domain Ω over 600 m with a uniform subdivision of 600 cells. The incoming wave boundary condition is provided using a $H_0 = 2$ m parameter, with a wave period $T_0 = 12$ s. The breaker model of XBeach uses the D. J. Roelvink (1993) formulation, with a breaker coefficient of $\gamma = 0.55$, a power $n = 15$, and a wave dissipation coefficient of 0.5. For the breaker model of SWAN and Shoaling, we simply use the breaker coefficient of $\gamma = 0.55$. The parameter files are present in the appendix A.1. This gives us the following results:

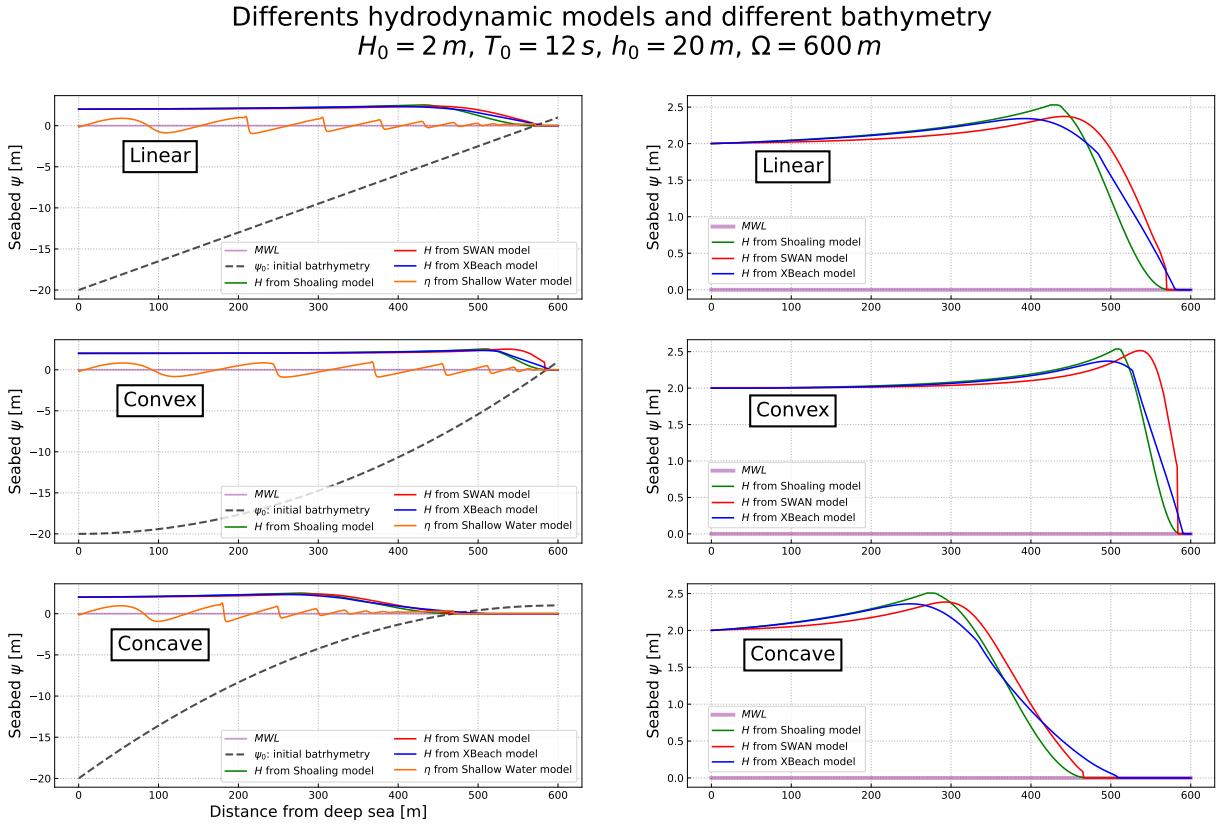


Figure 2.10 – Hydrodynamic results obtained with the Shoaling, SWAN, XBeach and Shallow-Water models. Bathymetric configurations from a linear, convex and concave open-sea configurations. Black bathymetry, green H from extended shoaling, red H from SWAN, blue H from XBeach and orange η from Shallow-Water model.

Also, we notice that the 3 spectral hydrodynamics are relatively close. The extended shoaling model seems to be slightly different from the two others.

6 Conclusion

In this chapter, we have seen the mathematical formalism on which coastal hydrodynamic models are based. We have chosen to present in detail the spectral and temporal resolution models and the way to generate a significant water height H_s . We present more precisely the well known models SWAN, XBeach, REF/DIF and Shallow-Water. Moreover, we have introduced an extended shoaling model which allows us to have correct results at a very low complexity. The hydrodynamic simulations presented for the benchmark ([Roelvink et al. 1995](#)) give the results very close to the experimental data. Open-sea simulations have also been introduced in order to highlight the versatility of our model.

Chapter key points

- We have introduced the mathematical formalisms of coastal hydrodynamics.
- We design a new hydrodynamic model which we call extended shoalling model.
- We performed simulations of a benchmark: LIP11D-1C with SWAN, XBeach and extended shoaling models.
- We performed simulations on Open-Sea with linear, concave and convex configuration using SWAN, XBeach, extended shoaling models and Shallow-Water model.

CHAPTER

Beaches morphodynamic modeling based on Hadamard Sensitivity Analysis

In the last chapter, we have selected some interesting wave models for incorporation into our OptiMorph model. However, the OptiMorph model in its chapter 1 version couldn't couple just any wave model, as there was the need to calculate a particular gradient (with respect to shape), which was done analytically. In this chapter, we present a morphodynamic model which can be coupled with any wave model capable of producing time/spectral averaged wave quantities. This model based on a wave energy minimization principle highlights the morphodynamic phenomenology, such as the sandbar creation. Such a model can be used in solving engineering optimization problems. It is also developed to illustrate the idea that beach sand transport can be considered as a non-local phenomenon. We used wave calculations from SWAN, XBeach and Shallow-Water in our model, and we compared the morphodynamic results to LIP and SANDS hydro-morphodynamic benchmark as well as open-sea simulations. Using supplementary mathematical development, we improved the minimization method using the Hadamard derivative.

Current chapter contents

1	Introduction	94
2	Gradient Calculation with Respect to the Shape $\nabla_\psi \mathcal{J}$	95
2.1	Analytical Calculation of $\nabla_\psi H$	95
2.2	Finite Difference Calculation of $\nabla_\psi H$	96
2.3	Automatic Differentiation (AD) Method to Calculate $\nabla_\psi H$	97
2.3.1	Direct and Reverse Modes of Automatic Differentiation	97
3	Using Hadamard for the Calculation of $\nabla_\psi \mathcal{J}$	97
3.1	Principle	97
3.2	Analytical Examples of Hadamard Derivative	98
3.2.1	Flat form	98
3.2.2	Linear Form	99
3.3	Numerical Validation	101
3.4	Validating the Hadamard Solution	102
4	Application of Hadamard Strategy	103

4.1	Description of Flumes Experiments	104
4.1.1	The SANDS Experiments	104
4.1.2	The LIP Experiments	105
4.2	Hydro-Morphodynamic Results on Flume Experiment	106
4.3	Hydro-Morphodynamic Results on Open-sea Configurations	108
5	Discussion	110
5.1	Computation time	110
5.2	Flume Simulation	111
5.3	Open-Sea Simulation	113
5.4	Gamma Sensibility	114
5.5	Limitation of a Wave Resolving Model	114
6	Conclusion	115

1 \mathcal{J} ntroduction

MORPHODYNAMIC MODELS are generally very complex and highly parameterized. They separately solve the physical equations of hydrodynamics and morphodynamics at a very small scale of the order of second in time and of the wave length in space. The OptiMorph model that we presented in Cook (2021) and Ronan Dupont et al. (2023) proposes a more global approach based on an optimization principle.

The optimization theory is the study of the evolution of a system while searching systematically for the minimum of a function derived from some of its physical properties. Using a certain number of mathematical optimization developments devoted to coastal sciences (Isèbe et al. 2014; Isèbe et al. 2008b; Isèbe et al. 2008a; Bouharguane et al. 2010; Mohammadi et al. 2014; Mohammadi et al. 2011; Cook et al. 2021c; Mohammadi 2017; Bouharguane et al. 2012; Ronan Dupont et al. 2022; Ronan Dupont et al. 2023), we have designed a model that describes the evolution of the sea bottom elevation while taking into account the coupling between morphodynamic and wave processes. This study is based on the assumption that the sea bottom adapts in time to minimize a certain wave-related function. The choice of this function determines the driving force behind the morphological evolution of the seabed. This optimization problem is subjected to a limited number of constraints, allowing for a more accurate description of the morphodynamic evolution.

The purpose of this study is to use the Hadamard (1914) derivative in order to calculate the gradient of any cost function \mathcal{J} with respect to the shape ψ , which allows us to solve the optimization problem at the core of the model. This strategy aims to create a generic morphodynamic model that can be used with any wave model.

The chapter starts with the Hadamard's strategy by presenting the different ways to compute the gradient with respect to the shape ψ . Hadamard's strategy is verified

with analytical cases. Finally, applications are performed with OptiMorph model using Hadamard strategy. We show that we can therefore use complex wave models such as XBeach (D. J. Roelvink et al. 2009), SWAN (Booij et al. 1996) and Shallow-Water (Marche et al. 2007). Part of the simulations are linked to the LIP and SANDS flume experiments (Roelvink et al. 1995; Eichendorf et al. 2018). Another part concerns simulations in open-sea configurations.

2 Gradient Calculation with Respect to the Shape $\nabla_\psi \mathcal{J}$

Calculation of $\nabla_\psi \mathcal{J}$ is necessary to do shape optimization with descent method equation (5.16). This quantity is not easy to compute since we do not differentiate on an axis but on a shape ψ . \mathcal{J} depends on wave height H ; it is thus advisable to have a very simple wave model in order to differentiate it easily. We assume at first that \mathcal{J} is of the form $\mathcal{J}(H(\psi(x)))$ involving dependencies with respect to wave quantities H . This sensitivity is given by:

$$\begin{aligned}\nabla_\psi \mathcal{J} &= \nabla_H \mathcal{J} \nabla_\psi H, \\ &= \nabla_H \left(\frac{1}{16} \rho g H^2 \right) \nabla_\psi H, \\ &= \frac{1}{8} \rho g H \nabla_\psi H.\end{aligned}\tag{3.1}$$

Calculating $\nabla_\psi \mathcal{J}$ reduces to that of $\nabla_\psi H$. It can be done analytically using the simple shoaling model described in equations (1.2a) and (1.2b) as described later in section 2.1. One can also use a heavy formalism like automatic differentiation (Hascoet et al. 2004; Mohammadi et al. 2011). These strategies are described below and in section 3, where we show how to obtain $\nabla_\psi H$ to whatever may be our functions H and ψ .

2.1 Analytical Calculation of $\nabla_\psi H$

The analytical method is the most precise (because it gives the exact value) and the fastest in calculation time. To illustrate the purpose, we take equation (1.2a) of H and we differentiate them in the following way:

$$\nabla_\psi H = \begin{cases} H_0(t) \nabla_\psi Ks(x, h) & \text{pour } x \in \Omega_S \\ \gamma \nabla_\psi h(x, t) & \text{pour } x \in \Omega_B \end{cases}.\tag{3.2}$$

The problem is reduced to the calculation of $\nabla_\psi K_S(x, t)$ and $\nabla_\psi h(x, t)$. The relation $h = h_0 - \psi$ ensures that $\nabla_\psi h(x, t) = -1$. Moreover, we have:

$$K_S = \left[\tanh(kh) \left(1 + \frac{2kh}{\sinh(2kh)} \right) \right]^{-1/2}. \quad (3.3)$$

Let $U(X) = \tanh(X) \left(1 + \frac{2X}{\sinh(2X)} \right)$ and $X = kh$. Introducing U in equation (3.3) and derivating ψ results in:

$$\nabla_\psi K_S = -\frac{1}{2} U^{-3/2} \nabla_\psi U. \quad (3.4)$$

By trigonometric transformation, we can demonstrate that:

$$\nabla_\psi U = \nabla_\psi X \frac{2 \cosh^2(X) - X \sinh(2X)}{\cosh^4(X)}, \quad (3.5)$$

we also have:

$$\nabla_\psi X = h \nabla_\psi k + k \nabla_\psi h = h \nabla_\psi k - k. \quad (3.6)$$

Moreover, differentiating both sides of the dispersion equation $\sigma^2 = gk \tanh(kh)$ by ψ gives

$$\nabla_\psi k = \frac{k^2}{\cosh(kh) \sinh(kh) + kh}. \quad (3.7)$$

Combining (3.4), (3.5), and (3.7), we obtain $\nabla_\psi K_S$, and therefore $\nabla_\psi H$, on Ω_S .

This method is the most accurate and robust because it gives the analytical solution directly. However, it is applicable to a very limited number of wave models. Indeed, they must be very simple to be differentiated by hand. Our ambition is to have a strategy that may allow differentiating the mathematical representation of any hydrodynamic.

2.2 Finite Difference Calculation of $\nabla_\psi H$

Finite difference methods are based on the idea of approximating the derivative of a function at a point by taking the difference between the values of the function at two adjacent points. Considering the directional gradient formula at ψ along direction l :

$$\nabla_\psi H(\psi) = \lim_{\varepsilon \rightarrow 0} \frac{H(\psi + \varepsilon l) - H(\psi)}{\varepsilon}. \quad (3.8)$$

We could define a first order finite difference approximation of the gradient at $\psi \in \mathbb{R}^N$

taking for l the vectors $e_{i=1..N}$ of the canonical basis of \mathbb{R}^N . The i^{th} evaluation provides the corresponding component of the gradient vector. This method requires $N + 1$ evaluation of the wave model which makes the method computationally expensive, as it can be classically of the order of several thousand runs in practice.

2.3 Automatic Differentiation (AD) Method to Calculate $\nabla_\psi H$

Automatic differentiation (AD) of programs is an important tool for numerical optimization and scientific computing. It is a technique for computing derivatives of a given program by successive derivation of the lines of the code. AD can be used to compute derivatives of functions with respect to both scalar and vector variables (Griewank et al. 2008).

2.3.1 Direct and Reverse Modes of Automatic Differentiation

Direct AD uses the chain rule to compute derivatives of a program with respect to the input parameters of the code. The direct AD method can be used to compute derivatives of functions of any order, including higher-order derivatives. This method is relatively simple to implement, and is often used when the number of input variables is small. On the other hand, when the size of input variables is large, the reverse mode of AD is used. The computation cost is independent of the size of the inputs. A typical AD tool is the TAPENADE program (Hascoet et al. 2004) which provides Fortran or C codes for the derivatives of programs in direct and reverse modes. This means that we need to provide the source code. As a consequence, the main limitation of this approach is that it cannot be applied to a commercial code when the source code is not provided. Even when the code is provided (open source), it is written in a modular way, which makes it very difficult to isolate the variables to differentiate.

3 Using Hadamard for the Calculation of $\nabla_\psi \mathcal{J}$

In this section, we focus on the calculation of $\nabla_\psi H$ in order to obtain $\nabla_\psi \mathcal{J}$ (as illustrated in section 2). This method can be applied whatever may be the variables: we can directly calculate $\nabla_\psi \mathcal{J}$. However, in this case, the approximation would be less good because the analytical derivative of \mathcal{J} (equation (3.1)) is always more accurate.

3.1 Principle

We use the approximation described in (Hadamard 1914; Mohammadi 2007; Mohammadi 2010). We consider $\nabla_\psi H$ in the sense of Hadamard following the definition:

$$\nabla_\psi H = \lim_{\varepsilon \rightarrow 0} \frac{H(\psi + \varepsilon n) - H(\psi)}{\varepsilon}, \quad (3.9)$$

where n is the normal to the shape ψ . This can be seen as applying a Gâteaux (1913) derivation in the direction normal to the shape. The principle is illustrated in figure 5.7.

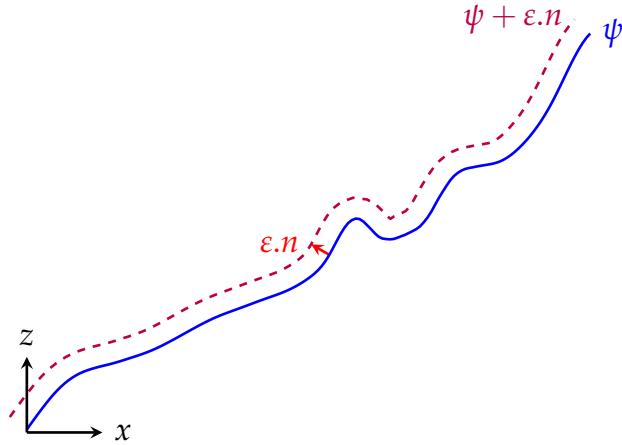


Figure 3.1 – Representation of two sea bottom profiles ψ and $\psi + \varepsilon n$. To calculate the gradient, we need to calculate at all points the associated normal vector n .

Using the Taylor-Young formula at order 1, we consider the following approximation:

$$\begin{aligned} \nabla_\psi H &= \lim_{\varepsilon \rightarrow 0} \frac{H(\psi) + \varepsilon \nabla_X H \cdot n - H(\psi)}{\varepsilon}, \\ &\approx (\nabla_X H) \cdot n, \end{aligned} \quad (3.10)$$

with $X = (x, z)^\top$. This approximation is illustrated in the following section 3.2 on simple analytical examples; and also on the simple shoaling model in section 3.4.

3.2 Analytical Examples of Hadamard Derivative

In this section, we illustrate analytical examples of derivation of ψ on a quantity A ; concretely, we calculate $\nabla_\psi A$.

3.2.1 Flat form

We consider the relation $A = \psi^2$. We set in a general way $\psi = \{(x, y) \in \mathbb{R}^2 \mid y - f_\psi(x) = 0\}$ the space of ψ with f_ψ the function describing the bottom.

The flat form $\psi = \{(x, y) \in \mathbb{R}^2 \mid y - c = 0\}$ deformed from εn is given by $\psi + \varepsilon n = \{(x, y) \in \mathbb{R}^2 \mid y - c - \varepsilon = 0\}$. It could be illustrated by the figure 3.1.

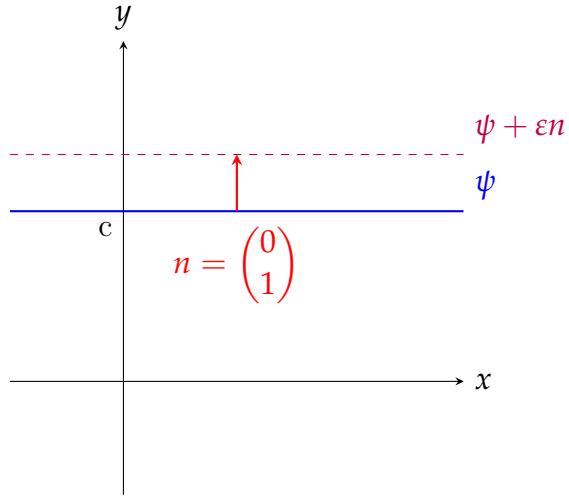


Figure 3.1 – Illustration of ψ and $\psi + \varepsilon n$ with the function $\psi : x \rightarrow c$

Here we have:

$$\begin{aligned}\psi &= \{(x, y) \in R^2 \mid y - c = 0\} \text{ and} \\ A &= \psi^2 = \{(x, y) \in R^2 \mid y - c^2 = 0\},\end{aligned}$$

then, we have, thank to the definition, on the one hand:

$$\begin{aligned}\nabla_\psi A &= \lim_{\varepsilon \rightarrow 0} \left(\frac{1}{\varepsilon} [A(\psi + \varepsilon n) - A(\psi)] \right), \\ &= \lim_{\varepsilon \rightarrow 0} \left(\frac{1}{\varepsilon} [\cancel{\psi^2} + 2\psi\varepsilon + \underbrace{\varepsilon^2}_{\rightarrow 0} - \cancel{\psi^2}] \right) \quad \text{because} \quad A(\psi + \varepsilon n) = (\psi + \varepsilon)^2, \\ &= 2\psi,\end{aligned}$$

on the other hand:

$$\nabla_X A \cdot n = 2\psi \nabla_X(\psi) \cdot n = 2\psi \begin{pmatrix} 0 \\ 1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = 2\psi, \quad (3.1)$$

and therefore

$$\nabla_X A \cdot n = \nabla_\psi A. \quad (3.2)$$

3.2.2 Linear Form

The linear form $\psi = \{(x, y) \in \mathbb{R}^2 \mid y - ax - b = 0\}$ deformed by εn is given by $\psi + \varepsilon n$. We consider the relation $A = \cos(\psi)$. We set in a general way $\psi = \{(x, y) \in \mathbb{R}^2 \mid y - f_\psi(x) = 0\}$ the space of ψ with f_ψ the function describing the bottom. It could be illustrated by

the figure 3.2.

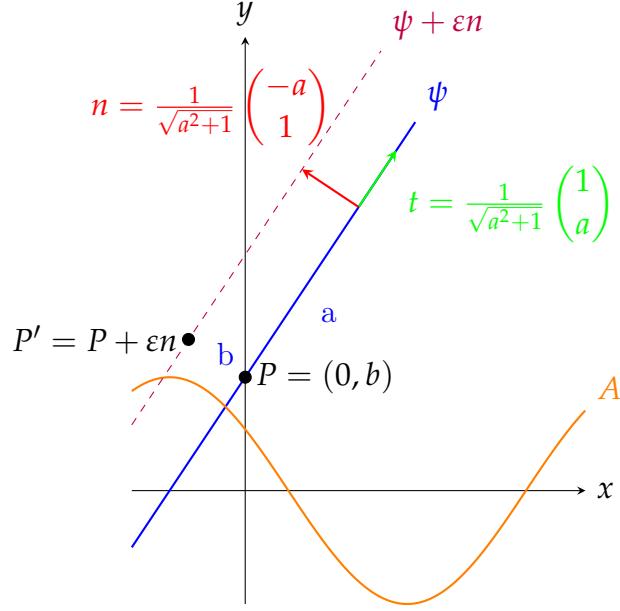


Figure 3.2 – Illustration of ψ , $\psi + \varepsilon n$ and A with the function $\psi : x \rightarrow ax + b$ and $A = \cos(\psi)$

We know the point $P(0, b)$ is contained on the line. The point $P' = P + \varepsilon n$ is therefore contained on the new translated line. We deduce the equation rapidly:

$$\psi + \varepsilon n = ax + \varepsilon\sqrt{a^2 + 1} + b. \quad (3.3)$$

Let us check the equation (5.21) for $A = \cos(\psi)$. On the one hand, we have:

$$\begin{aligned} \nabla_\psi A &= \lim_{\varepsilon \rightarrow 0} \left(\frac{1}{\varepsilon} [A(\psi + \varepsilon n) - A(\psi)] \right), \\ &= \lim_{\varepsilon \rightarrow 0} \left(\frac{1}{\varepsilon} [\cos(ax + \varepsilon\sqrt{a^2 + 1} + b) - \cos(ax + b)] \right), \\ &= \lim_{\varepsilon \rightarrow 0} \left(\frac{1}{\varepsilon} [\cos(ax + b) \underbrace{[\cos(\varepsilon\sqrt{a^2 + 1}) - 1]}_{\rightarrow 1 - \varepsilon^2(a^2 + 1)} - \sin(ax + b) \underbrace{\sin(\varepsilon\sqrt{a^2 + 1})}_{\rightarrow \varepsilon\sqrt{a^2 + 1}}] \right), \\ &= \lim_{\varepsilon \rightarrow 0} \left(\underbrace{-\cos(ax + b)\varepsilon(a^2 + 1)}_{\rightarrow 0} - \sin(ax + b)\sqrt{a^2 + 1} \right), \\ &= -\sin(\psi)\sqrt{a^2 + 1}. \end{aligned}$$

On the other hand, we have:

$$\nabla_X A = -\sin(\psi) \nabla_X \psi = -\sin(\psi) \begin{pmatrix} -a \\ 1 \end{pmatrix}, \quad (3.4)$$

and therefore:

$$\nabla_X A \cdot n = -\frac{\sin(\psi)}{\sqrt{a^2 + 1}} \begin{pmatrix} -a \\ 1 \end{pmatrix} \begin{pmatrix} -a \\ 1 \end{pmatrix} = -\sin(\psi) \frac{(a^2 + 1)}{\sqrt{a^2 + 1}} = -\sin(\psi) \sqrt{a^2 + 1}. \quad (3.5)$$

The equality: $\nabla_\psi A = \nabla_X A \cdot n$ is still verified.

3.3 Numerical Validation

The approximation (5.21) can be verified by calculating numerically the solution of the analytical example presented in the section 3.2.2. We calculate the error L^2 named $\mathcal{E}_{L^2} = \|(\nabla_\psi H)_{exact} - (\nabla_\psi H)_{numerical}\|_{L^2}$ for points which correspond to the spatial steps $dx = [10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 0.1, 1, 10, 100]$ for a length $L = 1000$ m. We obtain the curves in figure 3.3.

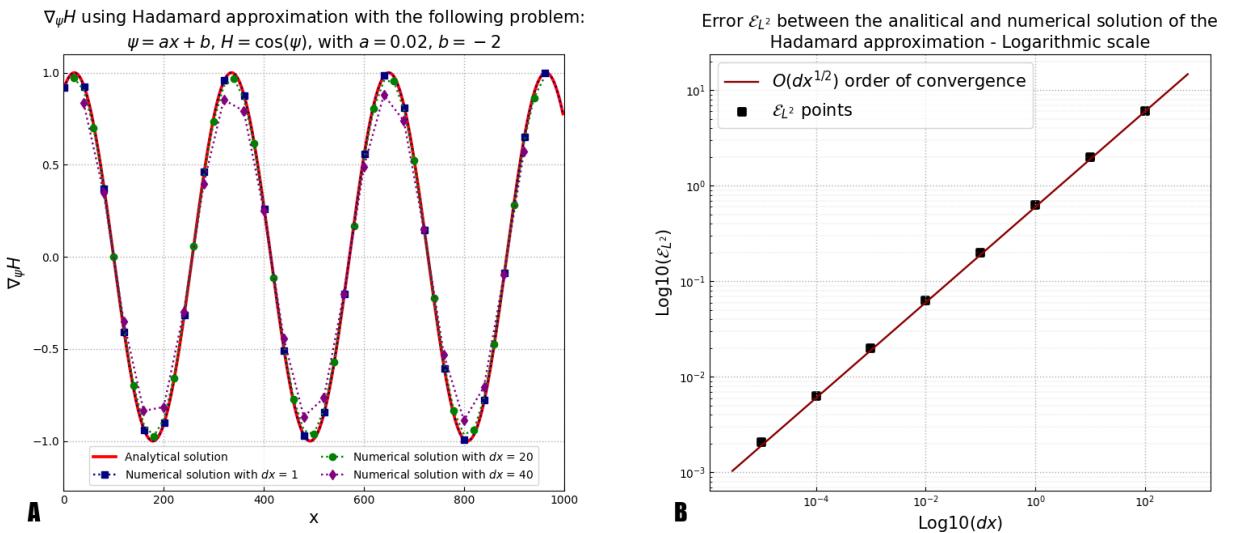


Figure 3.3 – A) Calculation of $\nabla_\psi H$ using Hadamard approximation with the following problem (see 3.2.2): $\psi = ax + b$, $H = \cos(\psi)$, with $a = 0.02$, $b = -2$. **B)** L2 error and order of convergence for a comparison between the analytical solution of the simple problem described in 3.2.2 with $H = \cos(\psi)$.

We notice that the approximation becomes good very quickly. On the figure 3.3.A, we see that an increment $dx = 20$ m is enough to reach an almost perfect approximation. The figure 3.3.B shows that the error is very small and converges to the order $\mathcal{O}(dx^{1/2})$. The sources of error for this calculation could be a) the approximation in the calculation of the vector n (in this case, it is null because ψ is linear), b) the computation of the

gradient by finite differences (order 1).

3.4 Validating the Hadamard Solution

Historically, the OptiMorph model used the shoaling equation equations (1.2a) and (1.2b) and was based on the analytical differentiation of this equation (section 2.1). The Hadamard strategy allows us to obtain a calculation of $\nabla_\psi H$ in a numerical way, as with finite differences. To implement this approach practically, we simply need to use the equation (5.21) with: $\nabla_X H = \begin{pmatrix} \frac{\partial H}{\partial x} \\ \frac{\partial H}{\partial \psi} \end{pmatrix}$ and $n = \frac{1}{\sqrt{d\psi^2 + dx^2}} \begin{pmatrix} -d\psi \\ dx \end{pmatrix}$ and we obtain:

$$\nabla_\psi H \approx \frac{\partial H}{\partial x} n_x + \frac{\partial H}{\partial \psi} n_z, \quad (3.6)$$

with n_x and n_z the x and z component of n . In OptiMorph, we implement equation (5.22) and we compare the calculations of $\nabla_\psi H$ using the simple shoaling model presented in equations (1.2a) and (1.2b). The figure 3.4 shows a comparison of the Hadamard and exact solutions on a representative example: an offshore water level $H_0 = 2$ m, an offshore water depth $h_0 = 10$ m, a wave period $T_0 = 10$ s and a linear bottom profile ψ . The figure 3.4.A corresponds to a simple case and the figure 3.4.B to a case with small scales perturbations of the sea bottom.

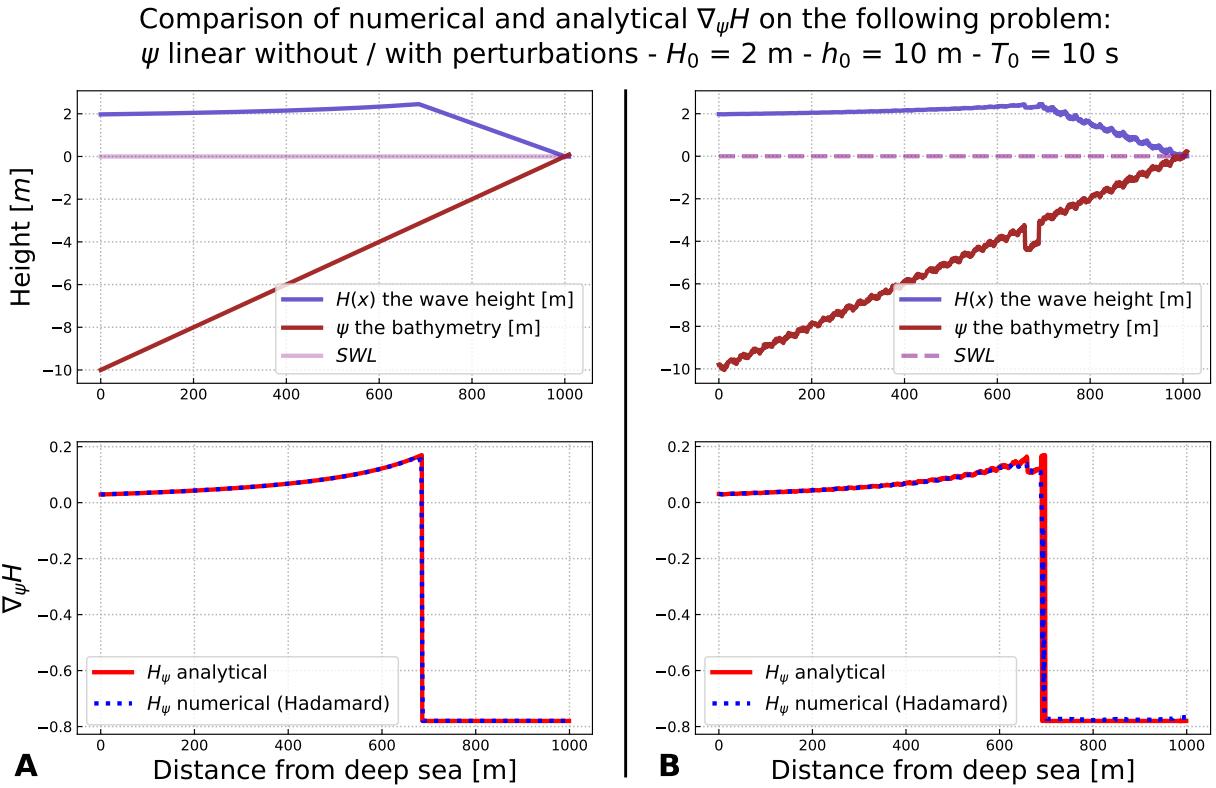


Figure 3.4 – Comparison of numerical and analytical solution of $\nabla_\psi H$ using OptiMorph model. Configuration without (A) / with (B) perturbations, $H_0 = 2$ m, $h_0 = 10$ m and the wave period $T_0 = 10$ s. In dodgerblue, the wave height H , in brown the bottom profile ψ , in red $\nabla_\psi H$ calculated analytically, in blue $\nabla_\psi H$ calculated by Hadamard strategy.

We notice that the approximation is very good. There is still one point that has a defect in the non-linearity at $x = 670$ m. However, this does not alter the morphodynamic results. To be sure of the robustness, we add non-linearity with a random function that induces perturbations. These are composed of sinusoidal functions and random translations between $[-0.2, 0.2]$, on the entire domain. We also set a hole at $x = 650$ m. We obtain the simulation figure 3.4.B. Even with all these perturbations, the Hadamard approximation remains very robust.

4 Application of Hadamard Strategy

To go further, we can use the Hadamard strategy to couple any wave model to the morphodynamic model based on the gradient descent equation presented in (5.16). The figure 3.5 shows the detailed implementation of this coupling.

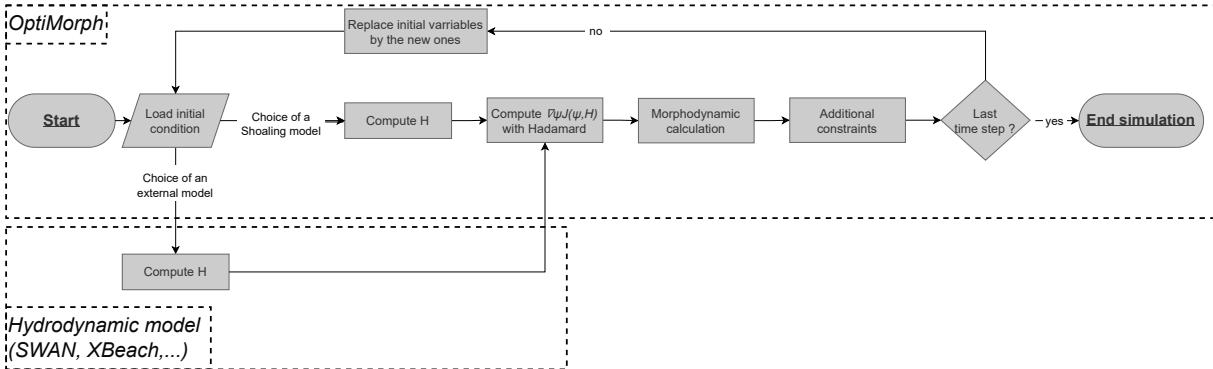


Figure 3.5 – OptiMorph workflow coupled with wave model.

Note that in all cases, it is necessary to calculate an averaged H to use it in the morphodynamic calculation. For spectral models such as SWAN and XBeach, this H is output directly from the model. However, wave-resolved models such as Shallow-Water 4.3 outputs a η . To integrate this quantity into the model, it must be averaged as a η_{RMS} with

$$\eta_{RMS} = \sqrt{\frac{1}{k T_0} \int_0^{k T_0} \eta^2 dt}, \quad k \in \mathbb{N} \quad (3.7)$$

and T_0 the wave period. In this case, we have $\eta_{RMS} \sim H$ provided that k is sufficiently large.

In this section, we perform Hadamard morphodynamics simulations forced by three distinct hydrostatic models: our extended shoaling model presented in appendix 3.4, SWAN and XBeach. Simulations are performed on 5 different experimental data sets: (i) one configuration from the SANDS experience ([Eichentopf et al. 2018](#)); (ii) one configuration from the LIP 11D experience flume experiment presented in table S.6 (part of the XBeach benchmark ([Roelvink et al. 1995](#))); (iii) three from open-sea configurations with linear, concave and convex bottom profiles.

4.1 Description of Flumes Experiments

In this section, we briefly present the LIP 11D ([Roelvink et al. 1995](#)) and SANDS ([Eichentopf et al. 2018](#)) experiments. These morphodynamic experiments are necessary to validate our model.

4.1.1 The SANDS Experiments

The experimental setup for this study was conducted at the Canal d’Investigació i Experimentació Marítima (CIEM), a large-scale wave flume located within the Universitat Politècnica de Catalunya (UPC) in Barcelona, Spain. The CIEM is a large-scale wave

flume of 100 m length, 3 m width and 4.5 m depth with a working water depth of 2.47 m and 2.5 m. Waves were generated using a hydraulic wave paddle positioned at the end of the deep-water section in the wave flume. The initial beach profile was carefully crafted by hand using well-sorted, commercial sand with a narrow grain size distribution ($d_{50} = 0.25$ mm, $d_{10} = 0.154$ mm, $d_{90} = 0.372$ mm), resulting in a measured sediment fall velocity of $w_s = 0.034$ m/s. The active portion of the beach profile featured a slope of 1/15. The experimental configuration of the SANDS project in Barcelona was meticulously documented in (Alsina et al. 2011).

This experiment (Eichentopf et al. 2018) is composed of two parts, an erosive part on a linear beach with slopes 1/15 with a forcing of $H_s = 0.53$ m and $T_0 = 4.14$ s for an experiment duration of 23 hours and 30 minutes. An accretionary section on the final beach profile of the erosive section, with a forcing of $H_s = 0.32$ m and $T_0 = 5.44$ s for an experiment duration of 20 hours and 25 minutes.

4.1.2 The LIP Experiments

The Large Installations Plan (LIP) experiments were conducted in the Delta Flume of Delft Hydraulics (now Deltares) (Roelvink et al. 1995), which is a large-scale facility measuring 225 x 7 x 5 m. During these experiments, various parameters such as water levels, wave-averaged velocity and suspended concentration profiles, orbital velocities, and bed levels were measured.

Three types of experiments were carried out in LIP under different types of irregular waves, resulting in three distinct beach states: stable (LIP 1A), erosive (LIP 1B), and accretive (LIP 1C).

In LIP 1A, the initial profile was linear with a slope of 1/30 and a median grain size of 0.22 mm. This part of the experiment represented a pre-storm event with the creation of a sedimentary bar under moderate wave conditions ($H_s = 0.9$ m, $T_0 = 5$ s).

The LIP 1B part of the experiment used the final profile from LIP 1A and represented a storm event with larger waves ($H_s = 1.4$ m, $T_0 = 5$ s). The bar moves seaward under the action of large waves, highlighting the physical process of erosion.

Finally, the LIP 1C part of the experiment used the final profile from LIP 1B and represented a post-storm event with smaller waves ($H_s = 0.6$ m, $T_0 = 8$ s). The bar moved back towards the coast asymmetrically, highlighting the physical process of accretion.

The bed profile was measured using a profile follower that used an automated sounding system. The LIP experiments provided valuable insights into the morphodynamic behaviour of sandy beaches under different wave conditions and have been widely used to validate numerical models of beach morphodynamics.

4.2 Hydro-Morphodynamic Results on Flume Experiment

To begin, we perform hydro-morphodynamic simulations with our morphodynamic approach using Hadamard’s calculation of $\nabla_\psi H$. To highlight the phenomenological aspect of our model, we start by performing simulations on SANDS erosive experience ([Eichentopf et al. 2018](#)).

In this case, we set up the models as follows. We set a domain Ω of 53 m in length with a uniform subdivision of 530 cells. For XBeach and SWAN, the incoming wave boundary condition is provided using a JONSWAP wave spectrum ([Hasselmann et al. 1973](#)), with a significant wave height of $H_s = 0.53$ m and a peak frequency at $f_p = 4.14 \text{ s}^{-1}$. For the extended shoaling model (appendix 3.4), we use directly H_s and a wave period $T_0 = 4.14$ s. The breaker model of XBeach uses the [D. J. Roelvink \(1993\)](#) formulation, with a breaker coefficient of $\gamma = 0.4$, a power $n = 15$, and a wave dissipation coefficient of 0.5. The breaker model of SWAN is based on the [Battjes et al. \(1978\)](#) breaking parametrization and the extended Shoaling model is simply based on a Munk breaking criterion $\gamma = 0.4$. The mobility parameter Y of our morphodynamic model has a value of $5 \times 10^{-3} \text{ m.s.kg}^{-1}$. The model is set to run 23.5 h using a coupling time of 42.3 s. We compare the numerical results to those experimental data. The hydro-morphodynamic results are presented in figure 3.6.A and the differences between the final bottom profile ψ_f and initial bottom profile ψ_0 are presented in figure 3.6.B. The reference is the experimental curve in dark red.

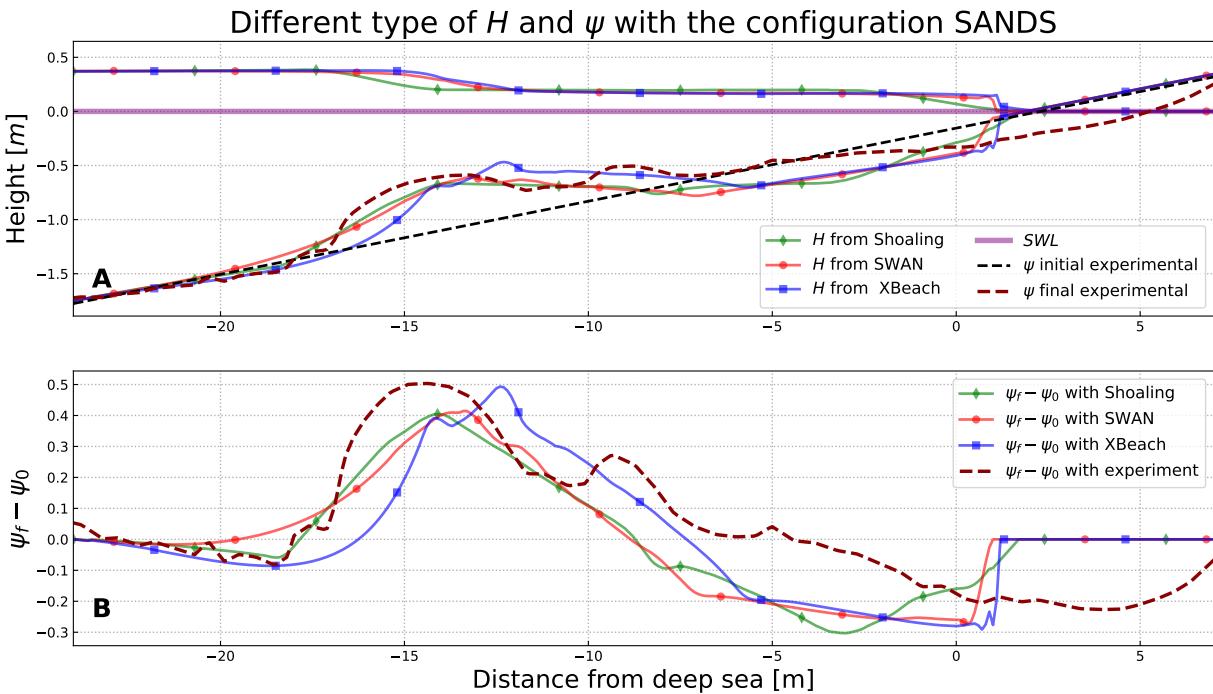


Figure 3.6 – A) Hydro-Morphodynamic results obtained with OptiMorph model using Hadamard strategy with wave models (Shoaling (green), SWAN (red) and XBeach (blue)). Bottom profile configuration from the SANDS erosive experience. Black: bottom profile, green: H and ψ from improved shoaling with Hadamard strategy, red: H and ψ from SWAN with Hadamard strategy, blue: H and ψ from XBeach with Hadamard strategy, dark red: ψ from experience. B) Morphodynamic ecart of $\psi_f - \psi_i$ obtained with the Shoaling, SWAN, XBeach models and experiment. Bottom profile configuration from the SANDS channel experiment. In green: morphodynamic differences from shoaling with Hadamard strategy ($\psi_{RMSE} = 11.7$ cm), red: morphodynamic differences from SWAN with Hadamard strategy ($\psi_{RMSE} = 12.7$ cm), blue: morphodynamic differences from XBeach with Hadamard strategy ($\psi_{RMSE} = 13.5$ cm), dark red: morphodynamic differences from the experiment.

In all three simulations and the experiment, a sedimentary bar is created over time and a trough is formed between the sandbar and the shore. These sedimentary bars are positioned below the breaking point of the wave. The sedimentary bars from the simulations have one main hump, whereas in the experiment there are two. In the simulations, the trough rises once the water has touched the shore ($x = 1$ m), while in the experiment, the trough continues afterwards (up to $x = 7$ m). The three simulations produce relatively similar results.

The next simulation from LIP - 1C flume experiment (Roelvink et al. 1995). In this other case, we set a domain Ω of 180 m in length with a uniform subdivision of 180 cells. For XBeach and SWAN, the incoming wave boundary condition is provided using a JONSWAP wave spectrum (Hasselmann et al. 1973), with a significant wave height of $H_s = 0.6$ m and a peak frequency at $f_p = 8$ s⁻¹. For the extended shoaling model (appendix 3.4), we use directly H_s and a wave period $T_0 = 8$ s. The breaker model of XBeach, SWAN and extended Shoaling model are the same as in the previous simulation (still with $\gamma = 0.4$). The mobility parameter Y of our morphodynamic model has a value of 5×10^{-3} m.s.kg⁻¹. The model is set to run 13 h using a coupling time of 46.8 s.

We compare the numerical results to those experimental data. The hydro-morphodynamic results are presented in figure 3.7.A and the differences between the final bottom profile ψ_f and initial bottom profile ψ_0 are presented in figure 3.7.B. The reference is the experimental curve in dark red.

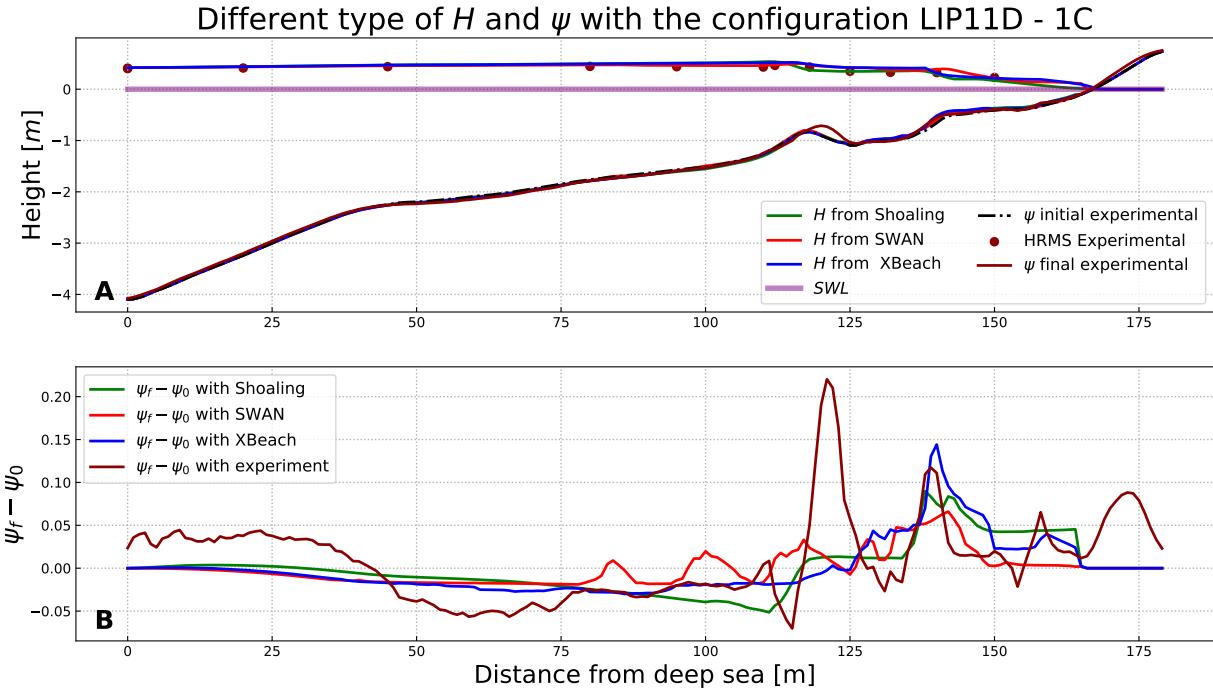


Figure 3.7 – A) Hydro-Morphodynamic results obtained with OptiMorph model using Hadamard strategy with wave models (Shoaling (green), SWAN (red) and XBeach (blue)). Bottom profile configuration from the LIP 1C channel experiment. Red points: measured HRMS, black: bottom profile, green: H and ψ from improved shoaling with Hadamard strategy, red: H and ψ from SWAN with Hadamard strategy, blue: H and ψ from XBeach with Hadamard strategy, dark red: ψ from experience. B) Morphodynamic ecarts of $\psi_f - \psi_i$ obtained with the Shoaling, SWAN, XBeach models and experiment. Bottom profile configuration from the LIP 1C channel experiment. In green: morphodynamic differences from shoaling with Hadamard strategy, red: morphodynamic differences from SWAN with Hadamard strategy, blue: morphodynamic differences from XBeach with Hadamard strategy, dark red: morphodynamic differences from the experiment.

In this experiment, the outer sedimentary bar is moving towards the coast. None of the simulations reproduces this behaviour: the outer bars remain in the same place ($x = 120$ m). In the experiment, the inner sediment bar grows ($x = 140$ m). Simulations show very similar behaviour. The three simulations produce relatively similar results.

4.3 Hydro-Morphodynamic Results on Open-sea Configurations

In this section, we perform simulations in open-sea configurations. The bottom profiles are linear, concave and convex shapes. These shapes are not directly observable in nature but representative of several typical settings (dissipative, reflexive). Forth, they allow to observe if the morphodynamic model is able to reproduce the phenomenology of sedimentary evolution of sand beaches. For these cases, we perform our morphodynamic model using waves from SWAN, XBeach, extended Shoaling and Shallow-Water models.

In this configuration, we use the same model settings as before with the exception of a Ω domain length resized at 600 m with a uniform subdivision in 600 cells. The forcing is no longer uniform but represents a storm event of 3 days with a peak wave height $H_s=2$ m. The waves have a period of $T_0 = 12$ s and the water depth at $x = 0$ is $h_0 = 20$ m. The

coupling time is set to 345 s. For the Shallow-Water model, the resolution is described in (Marche et al. 2007). η_{RMS} starts to be calculated after 10 simulated wave periods (10 T_0 , here 120 s). Then, η_{RMS} (defined in equation (3.7)) is averaged over 8 wave periods ($k = 8$). The results of these simulations are presented in figure 3.8.

**Evolution of ψ with different hydrodynamic models and different bottom profile
 $H_0 = 2 \text{ m}, T_0 = 12 \text{ s}, h_0 = 20 \text{ m}, \Omega = 600 \text{ m}$, Dynamic forcing**

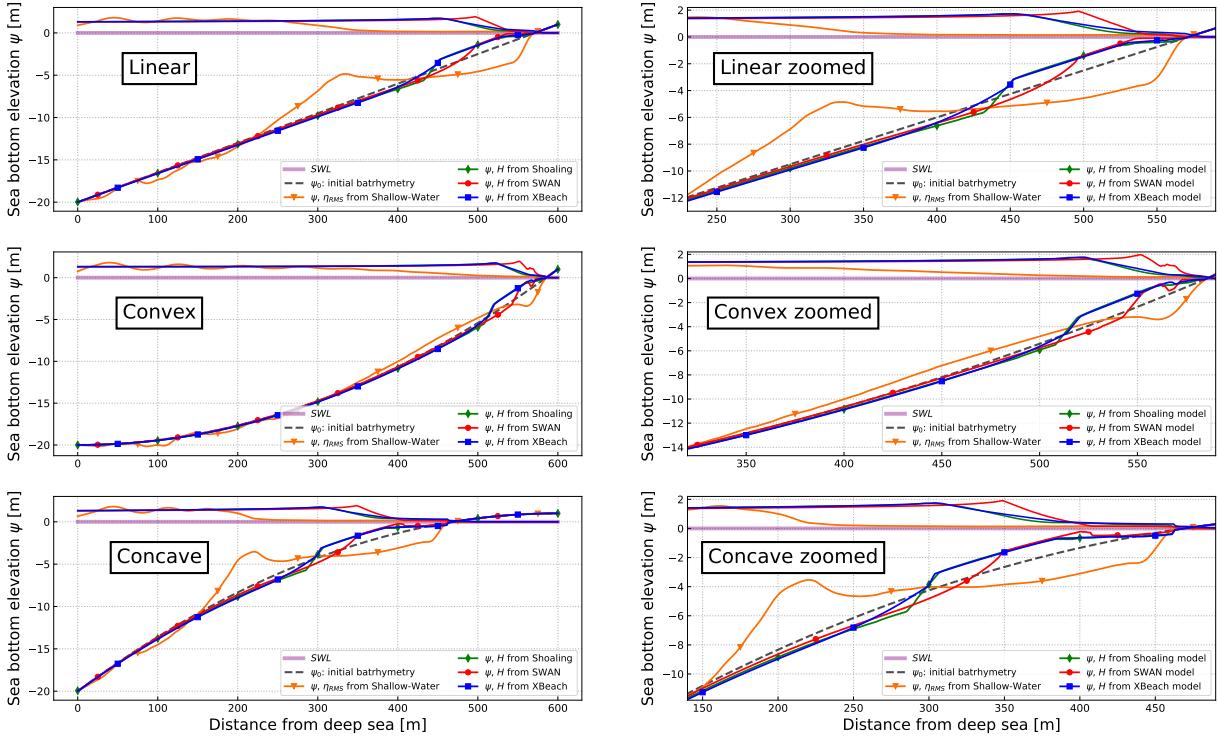


Figure 3.8 – Evolution of ψ using Hadamard strategy with extended Shoaling (green), SWAN (red), XBeach (blue) and Shallow-Water (orange) models. Simulation on open-sea configuration with linear, convex and concave configurations. Simulation parameters of $H_0 = 2 \text{ m}$, $T_0 = 12 \text{ s}$, $h_0 = 20 \text{ m}$, $\Omega = 600 \text{ m}$.

In all three cases of direct H_s calculation, the simulations produce very similar results. Depending on the angle of the slope, a sedimentary bar is observed more or less far from the shore. For a steep angle (convex beach), the sediment bar is very close to the shore; whereas for a slight angle (concave beach), the bar is further from the shore. All these sedimentary bars are all followed by a trough and are positioned below the breaking point of the wave. In the case of a convex profile, the sediment bar and the wave breaking, produced using SWAN, are closer to shore than the other two simulations.

In the case of the Shallow-Water model, the results are quite different: the sedimentary bar is larger, further from the shore and has a larger trough.

5 Discussion

5.1 Computation time

This section is devoted to the analysis of simulation times of the wave models and our morphodynamic calculation. Table 3.1 corresponds to the computation times for the LIP 11D - 1C simulations (section 4.2) with Hadamard strategy using the SWAN, XBeach and extended Shoaling models. XBeach was used for providing wave calculation only but this model also calculates the circulation. Therefore, it was necessary to run it over a longer time than that required by morphodynamics to get the right significant wave height H_s .

Simulation with 180 points	Hydrodynamic			Morphodynamic by gradient descent
	Shoaling	SWAN	XBeach	
Computation time for 1 iteration (s)	0.004	0.278	7.372	0.012
Total computation time for 1000 iterations (mins)	0.26	4.83	123.06	0.2

Table 3.1 – Computation time with 180 points calculated: LIP11D - 1C with different wave models. Simulations made with a 2.4 GHz computer using a single core on an Intel Xeon E5-2680 processor.

We notice that the calculation time of the Shoaling model is very small (direct calculation in python); it is 50 times smaller than that of SWAN and XBeach. XBeach calculation times come from the circulation model, which has the advantage of giving the current u (contrary to SWAN) and could be used for another definition of \mathcal{J} functional. The morphodynamic calculation time is very small and negligible compared to the hydrodynamic (except shoaling). By increasing the mesh size to 1000 points (5 times more), we obtain the table 3.2.

Simulation with 600 points	Hydrodynamic			Morphodynamic by gradient descent
	Shoaling	SWAN	XBeach	
Computation time for 1 iteration (s)	0.013	0.7158	17.243	0.044
Total computation for with 1000 iterations (mins)	0.966	12.67	288.12	0.762

Table 3.2 – Computation time with 600 points calculated with different wave models. Simulations made with a 2.4 GHz computer using a single core on an Intel Xeon E5-2680 processor.

The calculation times are also multiplied by 3. The computation time of the Shallow-Water model (presented in section 4.3) has not been shown as it is necessary to solve the model a very large number of iterations (corresponding to the simulation time of $k T_0$ from equation (3.7)) in order to perform a single descent iteration.

To save computing time, we could use some interpolation strategy between the grid of the wave tool and that of the morphodynamic model. This would allow performing wave computations on grids with fewer points with the same final results.

5.2 Flume Simulation

This section is devoted to the morphodynamic behaviour of our model using the Hadamard strategy on flume configuration ([Roelvink et al. 1995](#); [Eichentopf et al. 2018](#)). The main question is to check whether the numerical model is capable of reproducing the morphodynamic behaviours measured experimentally.

In the SANDS results shown in figure [3.6](#), we can see that a sediment bar is created from a linear beach profile (1/15). Although the simulations do not reproduce the sedimentary bar exactly like the experiment, they show very similar results. The sedimentary bar in the simulations is much shorter (in the sense of x) than in the experiment. In the simulations, the pattern of troughs between the sediment bar and the shoreline is very similar to that in the experiment. However, in the experiment, the trough goes beyond the water level. This result cannot be observed in our model (except with a tide) as there is currently no mechanism to model this erosion beyond the water level. This induces errors in our model, which conserves the quantity of sand. This lack of sand could explain why our sedimentary bar is shorter than the experimental one.

In the LIP 1C results shown in figure [3.7](#), we notice that two main sandbars are observed. The inner one ($x = 140$ m) seems to grow. The outer one ($x = 120$ m) moves to the shore. The 3 simulations based on Hadamard strategy succeeded in reproducing the behaviour of the inner bar ($x = 140$ m). XBeach model coupled to OptiMorph (blue) overestimates this sandbar and SWAN model coupled to OptiMorph (red) underestimates it. On this bar, there is a consequent loss of energy which induces a strong gradient and allows the bar to grow. However, none of the simulations has succeeded in reproducing the behaviour of the outer bar ($x = 120$ m) moving towards the shore.

This is because, the parameterization of the sea bottom ψ and equation [\(5.16\)](#) describing its dynamics only accounts for vertical variations using the gradient of the functional with respect to the sea bottom shape. Therefore, no lateral translation can be predicted by this model. To be able to account for lateral displacements, we need to introduce transport mechanisms, though, for instance, the following modification of the model:

$$\begin{cases} \psi_t = Y \Delta d - V \nabla_s \psi \\ \psi(t=0) = \psi_0 \end{cases}, \quad (3.8)$$

where we have introduced a transport operator in the right-hand side. $\nabla_s \psi$ is the spatial derivative of ψ along the mean slope of the sea bottom and V the velocity along this direction. We show the behaviour of the model using the following expression of V involving, the amplitude of orbital velocity at bottom U_b ([Wiberg et al. 2008](#)), and the

significant wave height H :

$$V = 0.01 U_b \left(\frac{H}{H_{max}} \right)^p \quad \text{with} \quad U_b = \frac{H \pi}{T_0 \sinh(kh)}. \quad (3.9)$$

The dimensionless morphodynamic factor 0.01 has been chosen in order to make the ranges of lateral and vertical variations comparable.

Figure 3.9.A illustrates the behaviour of this model for different p values. The best choice appears to be $p = 1$ and as expected, the bar moves back towards the coast (asymmetrically) under smaller waves which shows that transport mechanisms were necessary. Velocity distribution figure 3.9.B shows that the transport mechanisms are mainly located around the two sandbars.

To go farther, and to make the approach generic, we should express this velocity V using the gradient of the functional as done for the vertical motion using the gradient of the functional with respect to the shape.

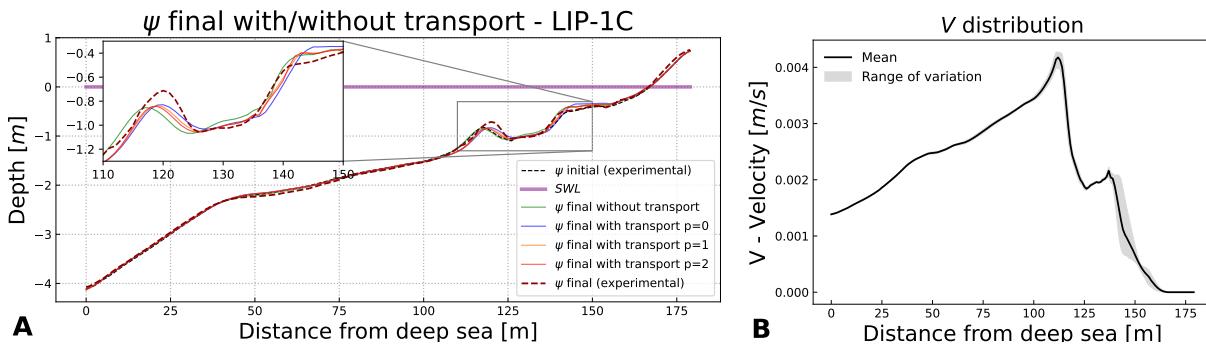


Figure 3.9 – A) Morphodynamic results by the OptiMorph model augmented by the transport mechanisms for $p = 0, 1, 2$ and the XBeach wave model, for the LIP 1C channel experiment. B) Velocity distribution for $p = 1$.

By comparing with hydro-morphodynamic models in the literature such as XBeach (D. J. Roelvink et al. 2009), we notice in figure 3.10 that for similar simulations (from the same benchmark),

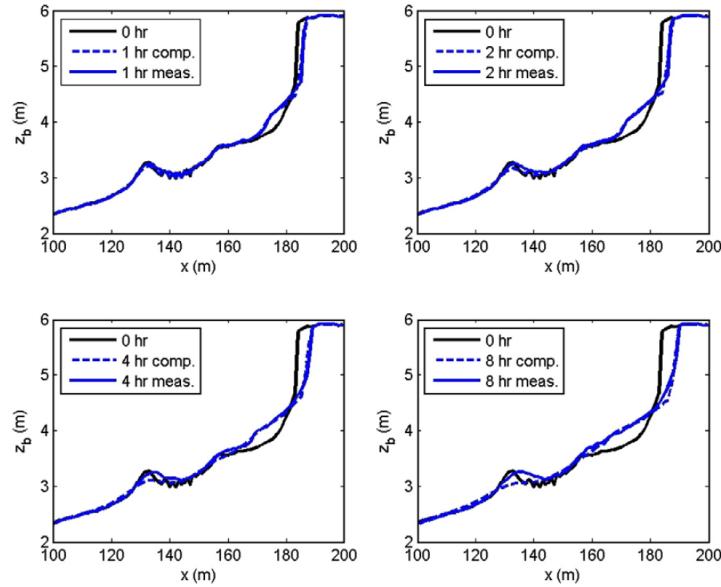


Figure 3.10 – Figure from (D. J. Roelvink et al. 2009): Measured and modeled bed level (with XBeach) after 1, 2, 4 and 8 h of wave action, for a water level of 4.56 m above the flume bottom

XBeach did not reproduce the morphodynamic behavior of the first sandbar. Indeed, this one collapses whereas it should move towards the shore. However, the shoreline displacement are very good reproduced by XBeach. In fact, it is specified that XBeach is not a model that is supposed to reproduce the behavior of the bars but rather that of the troughs at the shoreline.

5.3 Open-Sea Simulation

This section is devoted to the morphodynamic behaviour of our model using the Hadamard strategy on open-sea configuration. The two simulations figures 3.6 and 3.8, shows that there is a creation of the sandbar at the wave breaking point. Figure 3.8 shows that a slight pit is created before the sandbar and a trough one after. These observations are providing because they represent the major morphological features along a typical sand bar profiles. Indeed, it is common to observe sedimentary bars at the wave breaking point. Moreover, the steeper the slope (convex), the later the breaking, the closer the sandbar is to the shore ($x = 550$ m). Conversely, the gentler the slope (concave), the farther the breaking point, the farther the sandbar is from shore ($x = 350$ m). These types of beach profiles are usually observed in nature (Wright et al. 1984). It highlights the fact that even with an unrealistic initial beach profiles, the model can produce a realistic beach profile without any need in pre-nucleation of the bottom perturbation. Forth, whatever the physics behind the wave spectral model, the morphodynamic model is able to produce very similar morphodynamic results. On the other hand, the results produced

by the Shallow-Water model are quite different, but no less realistic.

5.4 Gamma Sensibility

To highlight the creation of sandbars at the wave breaking point, we artificially change the breaking point by varying the [W. Munk \(1949\)](#) criterion γ on hydro-morphodynamic simulations using Hadamard strategy. These simulations are performed with the SWAN wave model ([Booij et al. 1996](#); [Reniers et al. 2022](#)) and the same wave parameters as the simulation [4.3](#) ($T_0 = 12\text{ s}$ and $H_0 = 2\text{ m}$). By taking the [W. Munk \(1949\)](#) criteria at the values 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, we obtain the figure [3.11](#).

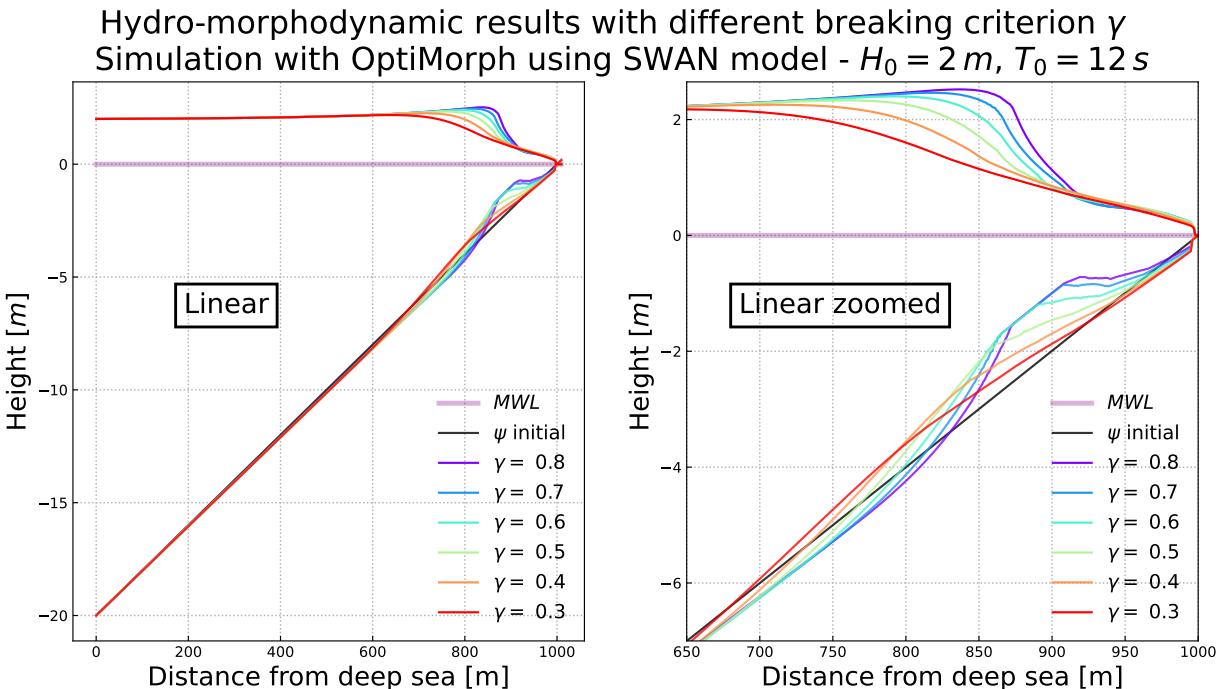


Figure 3.11 – Hydro-morphodynamic results with different breaking criterion γ - Simulation with OptiMorph (Hadamard strategy) using SWAN model - $H_0 = 2\text{ m}$, $T_0 = 12\text{ s}$, $h_0 = 20\text{ m}$.

The figure [3.11](#) shows that the sandbars are formed systematically at the wave breaking point. The higher the coefficient γ , the closer to the coast the waves break and the closer the sandbar is to the coast. Moreover, troughs at the lee side of the sandbar like observed in the nature are systematically nucleated ([Wright et al. 1984](#)).

5.5 Limitation of a Wave Resolving Model

Coupling our morphodynamic model with a wave resolving model has many limitations. The first is that computation times are much longer, as shown in the section [5.1](#). Secondly, it's much harder to get a breaking with this family of models. Under certain

conditions, these models have so much numerical dissipation that they do not mark the breaking. This is the case with our Shallow-Water model, taking a domain of $\Omega = 600$ m and a medium beach slope as shown in figure 3.12.B. Furthermore, we note that η_{RMS} is very far from the other wave models.

Comparison between H and η_{RMS} on the same scale

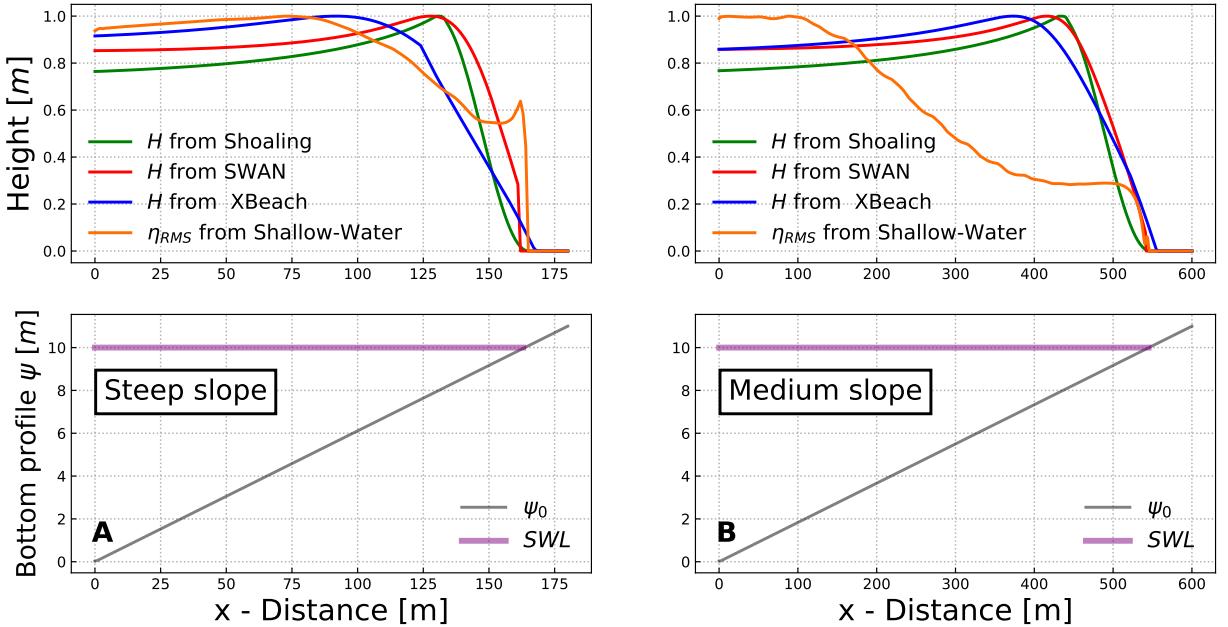


Figure 3.12 – Comparison between H from models Shoaling (green), SWAN (red), XBeach (blue) and η_{RMS} from Shallow-Water (orange) models. H and η_{RMS} are on the same scale. A) Simulation parameters of $H_0 = 1.5$ m, $T_0 = 8$ s, $h_0 = 20$ m, $\Omega = 180$ m. B) Simulation parameters of $H_0 = 1.5$ m, $T_0 = 8$ s, $h_0 = 20$ m, $\Omega = 600$ m.

The previous section 5.4 shows that it is necessary to have a well-marked breaking in order to have a realistic morphodynamic response. In the case of a steep slope (figure 3.12.A), it is possible to have a marked breaking and therefore a coherent morphodynamic response. η_{RMS} is also quite close to the other wave heights. Once again, we notice that our model is conditioned by wave breaking.

6 Conclusion

In this chapter, we have described the Hadamard strategy that we have applied to our model. Using this approach, we calculate an approximation of the gradient $\nabla_\psi \mathcal{J}$ of the functional \mathcal{J} with respect to the shape ψ without any additional wave calculation. This study allows us to differentiate any functional \mathcal{J} according to any input variable and relating to any wave model. The analytical and numerical comparisons performed prove that Hadamard strategy is accurate and robust. We applied this tool to realistic

and idealized hydro-morphodynamic simulations. This strategy is very powerful, as it can be applied to any shape optimization problem. The morphodynamic results with SANDS ([Eichentopf et al. 2018](#)) are very encouraging because they succeed in reproducing the dynamics of the dominant sedimentary bar. However, the initial results on LIP 1C ([Roelvink et al. 1995](#)) failed to reproduce the displacement of the outer sedimentary bar, although the behaviour of the inner was well reproduced. The model still needs to be improved in order to fix the lateral displacement and erosion above the water level. Nevertheless, our model is of low-complexity and reproduces the phenomenology as shown by the open-sea and SANDS results where it creates a bar at the breaking point without the need of a priori nucleation or pre-location of the bar.

Chapter key points

- A morphodynamic model by minimization that can be coupled with any wave model.
- Encouraging results on known morphodynamic benchmarks and open-sea configurations.
- An encouraging lead for lateral displacement.
- Reproduces certain natural coastal mechanisms (sandbars, erosion,), our model is able to create realistic beaches with a bar at the breakpoint, starting from scratch.
- Hadamard is a new approach which allows obtaining easily and robustly $\nabla_{\psi}\mathcal{J}$. It can be used for general shape optimization.

CHAPTER

4

Extending the OptiMorph model in 2D

In the last chapter, we have found a strategy to couple our morphodynamic models with any wave model capable of producing time/spectral averaged wave quantities. We integrated wave calculations from SWAN, XBeach and Shallow-Water into our model, and compared the morphodynamic results with 1D hydro-morphodynamic references from LIP and SANDS, as well as with open-ocean simulations. In this chapter, we develop the 2D model in a straightforward manner. We then carry out morphodynamic simulations reproducing the COPTER 2D basin experiment. Finally, we discuss the sensitivity of the wave model to our morphodynamic results.

Current chapter contents

1	Introduction	118
2	Upgrade OptiMorph Model to 2D	118
2.1	2D Wave Model	118
2.2	2D Morphodynamic Model	119
2.3	2D Hadamard Derivative	119
2.3.1	Mathematical Background	120
2.3.2	Numerical Validation	120
2.4	2D Constraints	121
2.4.1	Slope Constraint	121
2.4.2	Sand Conservation Constraint	122
3	2D Applications	122
3.1	Presentation of the Copter 2D experience	122
3.2	Application on Copter 2D	124
3.3	Linear Seabed with Geotube	124
4	Discussion	126
5	Conclusion	126

1 \mathcal{J} ntroduction

The extension of our morphodynamic model to a higher dimension opens up a much wider range of applications: from simple simulations on a 1D real seabed to simulations on 2D more complex seabeds with structures that attenuate wave effects.

Unlike 1D, 2D takes into account a wider range of wave effects. Aspects such as diffraction, refraction and swell angle must not be neglected. It is therefore important to couple our morphodynamic model with a powerful hydrodynamic model that can handle 2D effects. Many 2D models handle these 2D effects and are capable of generating averaged water levels ([D. J. Roelvink et al. 2010](#); [Booij et al. 1996](#); [James Thornton Kirby et al. 1994](#)). In this chapter, simulations will mainly be carried out using the REF/DIF: refraction/diffraction model ([James Thornton Kirby et al. 1994](#)). However, the user is free to use the morphodynamic model with the hydrodynamic model he prefers.

In numerical models, the transition from a 1D to a 2D model is often a tricky step. In fact, it is often necessary to develop the theory anew and redo the code from scratch. For example, in finite volumes, it is necessary to rewrite the equation model and all the numerical schemes, then implement them. In our case, this step has the advantage of being relatively easy. Indeed, the morphodynamic equation (4.2) has the advantage of being global and therefore always functional in 2D without change.

In this chapter, we will first see how the model has been extended to dimension 2 and its validity in 2D. Then, we will then carry out simulations on the Copter basin configuration in 2D. Finally, we discuss the sensitivity of the wave model to our morphodynamic results.

2 \mathcal{U} pgrade OptiMorph Model to 2D

The OptiMorph 2D model will operate in a similar way to the 1D model. It will retain all its versatility, always being able to be coupled with any 2D hydrodynamic model. In this thesis, calculations will be performed on simple square grids, although implementation on triangular grids may be possible in the future. However, this is not the aim of the thesis.

2.1 2D Wave Model

In the previous chapter [3](#), we demonstrated that the calculation of the gradient $\nabla_{\psi}\mathcal{J}$ was possible in 1D using the Hadamard approach. The transition to 2D is described in section [2.3](#). As a result, any 2D hydrodynamic model can be implemented in our code. The choice of hydrodynamic model is decisive in obtaining interesting morphodynamic results. 2D effects such as diffraction and reflection must therefore be taken into account.

The REF/DIF models (James Thornton Kirby et al. 1994) is very well suited to this type of phenomenon. This model is coupled between a refraction part and a diffraction part. It solves the mild-slope equation of Berkhoff (1972) and a dissipation equation. It should be noted that this model provides accurate results for the wave field on bed slopes ranging from 0 to about 1/3. More details on hydrodynamic models are provided in chapter 2, section 4.4, including a detailed presentation of the REF/DIF models.

2.2 2D Morphodynamic Model

The advantage of a global morphodynamic model makes it usable in any space without any changes. We recall below the definition of the functional \mathcal{J} (J s m^{-1}). For all $t \in [0, T_f]$, we have:

$$\mathcal{J}(\psi, t) = \frac{1}{16} \int_{t-T_{coupl}}^t \int_{\Omega} \rho_w g H^2(\psi, x, y, \tau) dx d\tau, \quad (4.1)$$

where H denotes the height of the waves over the 2D domain Ω (m^2), ρ_w is water density (kg m^{-3}), and g is the gravitational acceleration (m s^{-2}). T_{coupl} (s) defines the coupling time interval between hydrodynamic and morphodynamic models so that we have T_f/T_{coupl} iterations.

As explained above, the morphodynamic equation (4.2) remains the same,

$$\begin{cases} \psi_t = Y \Lambda d \\ \psi(t=0) = \psi_0. \end{cases} \quad (4.2)$$

The initial state ψ_0 is defined on \mathbb{R}^2 , ψ_t is the evolution of the bottom elevation over time (m s^{-1}) (also defined on \mathbb{R}^2), Y is a measure of the sand mobility expressed in m s kg^{-1} (also defined on \mathbb{R}^2), Λ measures the excitation of the seabed by the orbital motion of water waves, and d is the direction of the descent (J s m^{-2}), which indicates the manner in which the sea bottom changes. In unconstrained configurations, there would be $d = -\nabla_{\psi} \mathcal{J}$, which by its definition indicates the direction of a local minimum of \mathcal{J} with respect to ψ as illustrated in figure 1.5 in 1D.

2.3 2D Hadamard Derivative

In this section, we explain the calculation of the gradient with respect to the shape ∇_{ψ} , in 2D, in a similar way to the previous chapter 3. On the one hand, we will look at the mathematical developments. On the other, we will look at the 2D numerical validation, and thus at the limits of this strategy.

2.3.1 Mathematical Background

The Hadamard strategy introduced in the previous chapter 3 has been written in multi-dimensional (section 3). Using the equation (5.21) with $\nabla_X \mathcal{J} = \left(\frac{\partial \mathcal{J}}{\partial x}, \frac{\partial \mathcal{J}}{\partial y}, \frac{\partial \mathcal{J}}{\partial \psi} \right)^\top$ we obtain the new equation:

$$\nabla_\psi \mathcal{J} \approx \frac{\partial \mathcal{J}}{\partial x} n_x + \frac{\partial \mathcal{J}}{\partial y} n_y + \frac{\partial \mathcal{J}}{\partial \psi} n_z \quad (4.3)$$

with n_x , n_y and n_z the x , y and z component of n . The vector n is calculated on each component i, j of the grid as the vector product of two non-collinear vectors of the plane generated by the 3 points associated to $(\psi_{i+1,j}, \psi_{i,j}, \psi_{i,j+1})$ (see figure 4.3).

2.3.2 Numerical Validation

1D numerical validation was carried out in the previous chapter 3 section 3.3. The 2D extension does not change the validity of our Hadamard strategy. Two simple 2D validation cases are shown in figures 4.1 and 4.2. In these cases, we use the simple shoaling model presented in equations (1.2a). This allows us to directly compare the analytical solution $\nabla_\psi H$ with the numerical one. In the first case, we assume a linear seabed. In the second 4.2, we add a bump to induce non-linearity.

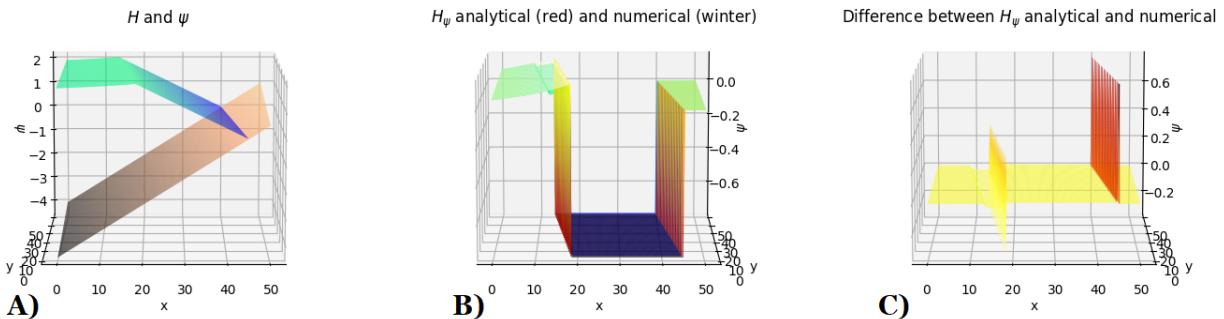


Figure 4.1 – OptiMorph 2D using Hadamard with multi-1D simple shoaling models with linear bottom elevation. A) Seabed and Wave Height B) Superposition of analytical and numerical gradients C) Differences between analytical and numerical gradients.

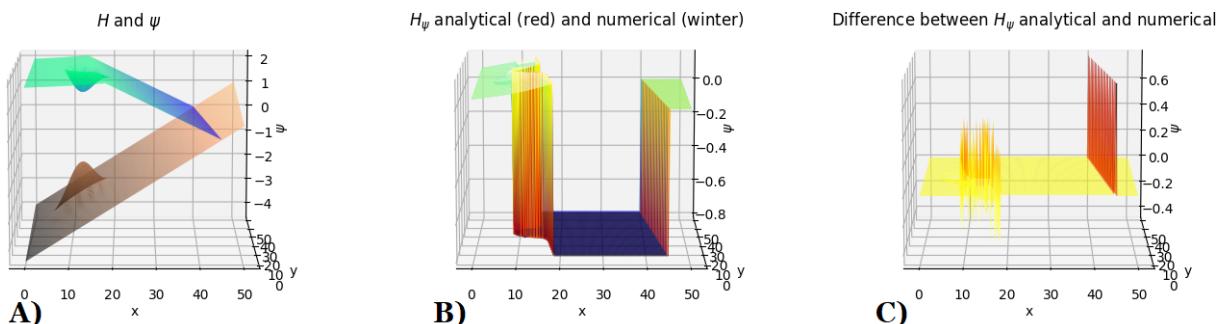


Figure 4.2 – OptiMorph 2D using Hadamard with multi-1D simple shoaling models with linear bottom elevation + geotube.
 A) Seabed and Wave Height B) Superposition of analytical and numerical gradients C) Differences between analytical and numerical gradients.

The results **4.1.C** and **4.2.C** show that the model is very robust over the whole Ω domain. However, some difficulties are encountered on the non-linear parts of ψ and H . This can be explained by the fact that the n normal vectors are often miscalculated in these parts.

2.4 2D Constraints

The two constraints of the model are easily transformed into 2D. The first is the slope constraint, which ensures that there are no unrealistic slopes in the model. The second is the sand conservation constraint, which is useful in experimental configurations, for example.

2.4.1 Slope Constraint

The 2D slope constraint becomes slightly more complex than in 1D (equation (5.23)). In fact, it is necessary to take into account the maximum slope according to \vec{x} (red nodes on figure **4.3**),

$$\left| \frac{\partial \psi}{\partial x} \right| \leq M_{\text{slope}}, \quad (4.4)$$

according to \vec{y} (blue nodes on figure **4.3**),

$$\left| \frac{\partial \psi}{\partial y} \right| \leq M_{\text{slope}}, \quad (4.5)$$

and also on the diagonals along \vec{a} and \vec{b} (green nodes on figure **4.3**),

$$\left| \frac{\partial \psi}{\partial a} \right| \leq M_{\text{slope}}, \quad \text{and} \quad \left| \frac{\partial \psi}{\partial b} \right| \leq M_{\text{slope}}. \quad (4.6)$$

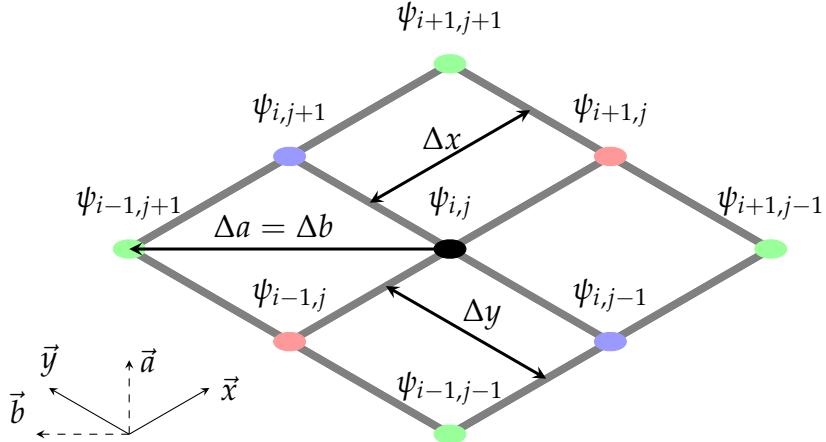


Figure 4.3 – Nodes representation.

2.4.2 Sand Conservation Constraint

The sand conservation constraint remains the same. The integration of the Ω domain simply switches from 1D to 2D as illustrated in the following equation (4.7).

$$\int_{\Omega} \psi(t, x, y) d\Omega = \int_{\Omega} \psi_0(x, y) d\Omega, \quad \forall t \in [0, T_f], \quad (4.7)$$

with ψ the seabed at time t and ψ_0 the initial seabed.

3 2D Applications

Contrary to the previous chapter, 2D morphodynamic data are very rare. Most 2D validations are carried out in 1D: LIP, SANDS, Duck, etc. It is therefore difficult to obtain 2D morphodynamic validation data, so we will use 2D Copter data (Bouchette 2017), whose 1D data were used in the chapter 1. We will then carry out an application with a linear seabed by adding a geotextile tube as was done in the multi-1D section 4 of the chapter 1.

3.1 Presentation of the Copter 2D experience

The 2D Copter experiment (Bouchette 2017) has been conducted in the 30 m x 30m LHF wave basin in Grenoble (see Figure 4.4A) with a length scale of 1/10 and a time scale 1/3 (obtained with Froude scaling).

The sediment parameters that have been used in the present experiments are as follows: the density of 2 650 kg/m³ and median diameter $d_{50} = 0.166$ mm. This choice leads to

fulfill a Rouse scaling for a prototype grain size of $d_{50} \approx 0.3$ mm.

The beach morphology was measured by means of a laser profiler mounted on a motorized trolley located on a sliding rail **4.4A**). This measurement technique required emptying the basin before recording the bed elevation. The seabed elevation was recorded with millimeter accuracy every 10 cm and 1 cm in the alongshore and cross-shore direction, respectively. As shown in the Figure **4.4A**) the bathymetric survey zone was restricted by the sliding rail configuration and covered the area $7.84 \text{ m} < x < 22.84 \text{ m}$ in the cross-shore direction, $3.12 \text{ m} < y < 28.02 \text{ m}$ alongshore.

The free surface elevations were measured by means of 18 capacitance gages.

JONSWAP irregular waves were generated over 20mn sequences, repeated several times. Four typical wave climates are considered:

1. Storm rising: $H_s = 0.17 \text{ m} / T_0 = 2.1 \text{ s}$,
2. Storm apex: $H_s = 0.23 \text{ m} / T_0 = 2.3 \text{ s}$,
3. Storm waning: $H_s = 0.18 \text{ m} / T_0 = 3.5 \text{ s}$,
4. Calm wave conditions: $H_s = 0.11 \text{ m} / T_0 = 2 \text{ s}$.

In part 2 (Storm apex), a geotube was added as shown in figure **4.4**.

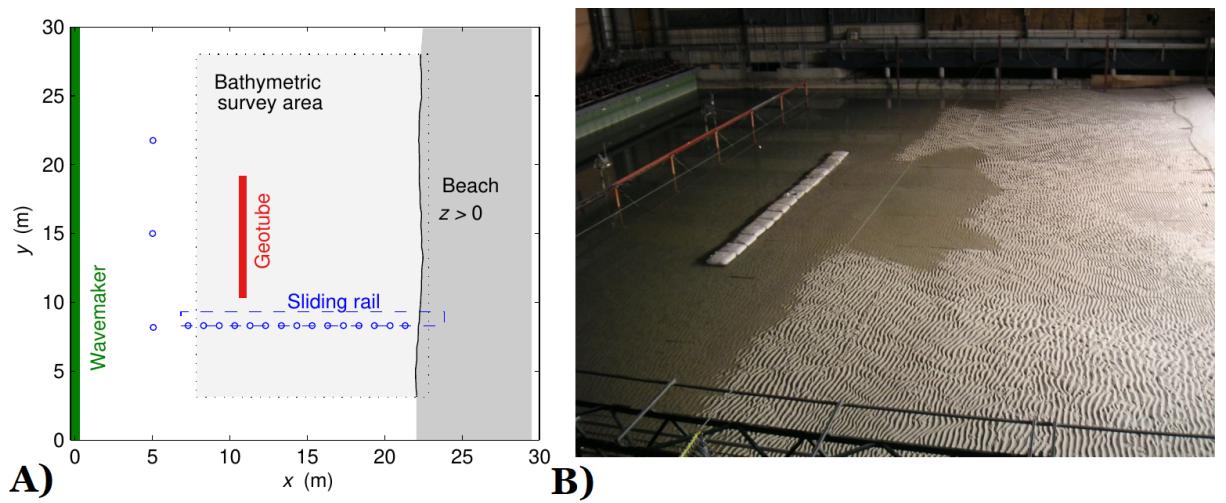


Figure 4.4 – A) Schema of Copter's 2D configuration. **B)** Photo of the 2D Copter experiment.

3.2 Application on Copter 2D

To begin, we perform hydro-morphodynamic simulations with our morphodynamic approach using Hadamard’s calculation of $\nabla_{\psi}\mathcal{J}$. To highlight the phenomenological aspect of our model, we start by performing simulations on Copter 2D seabed.

In this case, we set up the models as follows. We set a domain $\Omega = 24.9 \times 15$ m with a uniform subdivision in \vec{x} of 50 cells and uniform subdivision in \vec{y} of 50 cells. For REF/DIF and Shoaling multi-1D models, the incoming wave boundary condition is $H_s = 0.23$ m, a wave period $T_0 = 2.3$ s and a wave angle of incidence of 0 degrees, but this can be changed in REF/DIF. The breaker model of the extended Shoaling model is simply based on a [W. Munk \(1949\)](#) breaking criterion $\gamma = 0.78$. The mobility parameter Y of our morphodynamic model has a value of 4×10^{-2} m.s.kg⁻¹. The model is set to run 20 mins using a coupling time of 24 s. We compare in figure 4.5 the results obtained using the two wave models multi-1D shoaling, REF/DIF and the experimental result.

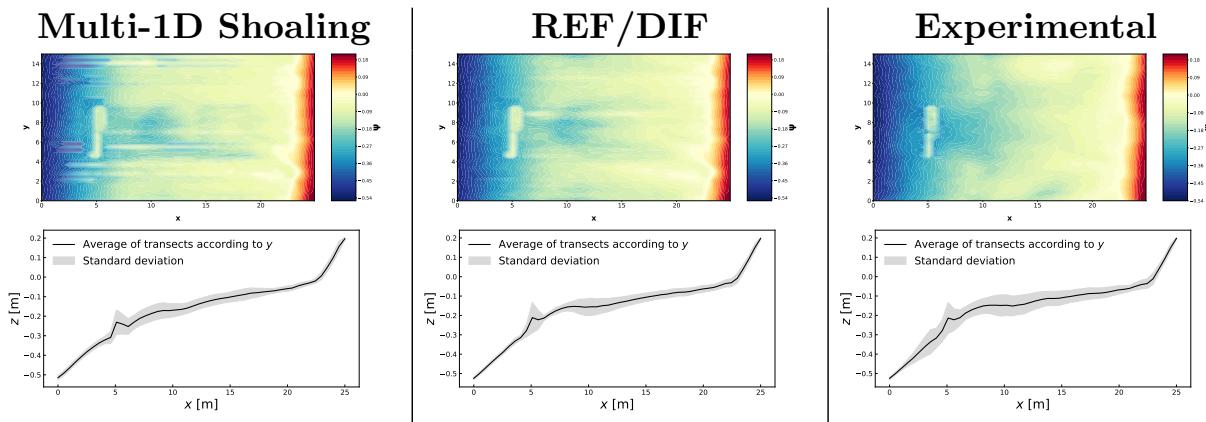


Figure 4.5 – (Top) Morphodynamic results using OptiMorph in 2D using the Multi-1D Shoaling model (left), REF/DIF (center) and comparing with experiments (right). **(Bottom)** Variability of the transects (in \vec{x} direction) with the mean (black) and standard deviation (gray).

In all three figures, a trough is visible at the rear of the geotube. The variability of morphodynamic results with the multi-1D Shoaling model and the experiment is much greater than that of REF/DIF.

3.3 Linear Seabed with Geotube

In this case, we set up the models as follows. We set a domain $\Omega = 600 \times 20$ m with a uniform subdivision in \vec{x} of 300 cells and uniform subdivision in \vec{y} of 60 cells. For REF/DIF and Shoaling multi-1D models, the incoming wave boundary condition is $H_0 = 2$ m, a wave period $T_0 = 6$ s and a wave angle of incidence of 0 degrees. The breaker model is the same as in the previous section. The mobility parameter Y of our

morphodynamic model has a value of 4×10^{-2} m.s.kg $^{-1}$. The model is set to run 3 days using a coupling time of 5184 s.

The initial seabed is linear (see Figure 4.6), with a \vec{x} slope of 1/100, with the addition of a geotextile tube at $x = 150$ m and a height of 2.5 m.

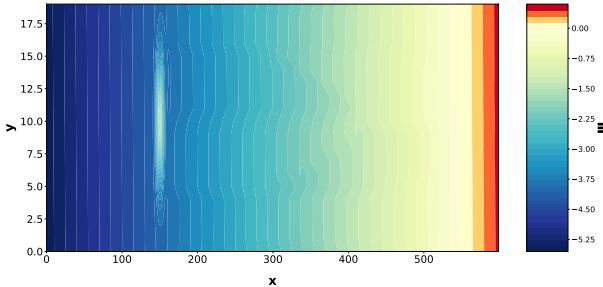


Figure 4.6 – Initial seabed with a linear slope of 1/100 with a geotextile tube at $x = 150$ m.

We compare in figure 4.7 the results obtained using the two hydrodynamic models shoaling / REF/DIF.

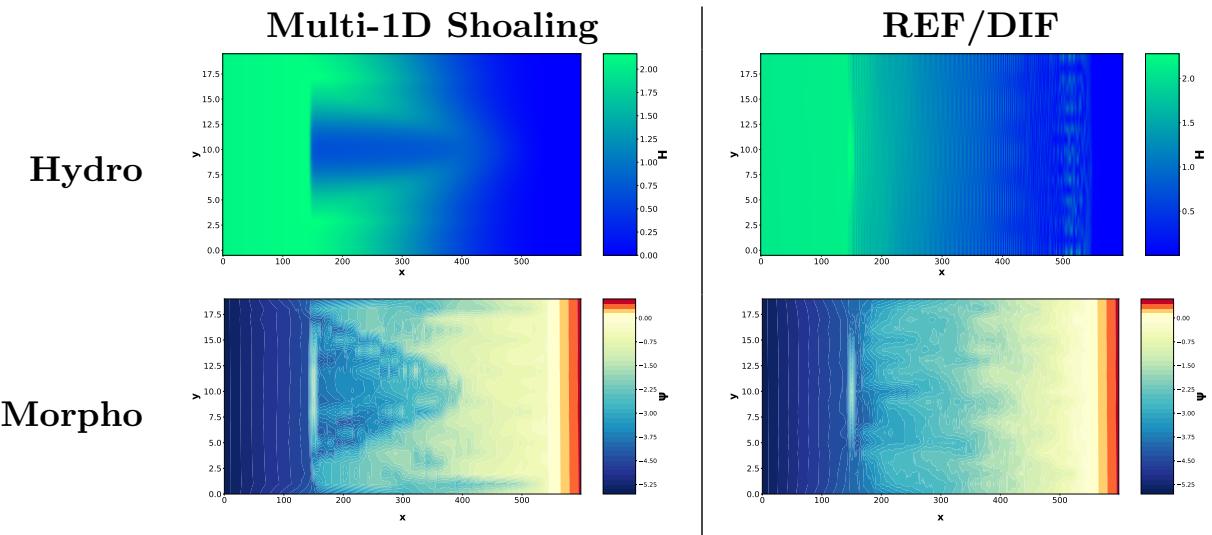


Figure 4.7 – 2D simulation of OptiMorph using Shoaling and REF/DIF models. Linear seabed (1/100) with geotube. Offshore height $H_0 = 2$ m and wave period $T_0 = 6$ s.

Here, the hydro-morphodynamic results produced by the two models are completely different. However, a trough can still be seen behind the geotube. This is much more pronounced using the multi-1D Shoaling model than using the REF/DIF models. Small oscillations are observed on the REF/DIF wave model.

4 Discussion

In Copter's 2D configuration ([Bouchette 2017](#)), the geotextile tube can be seen as a sedimentary bar. This leads to the wave breaking prematurely, creating a trough behind the geotube. The morphodynamic model coupled with multi-1D shoaling or REF/DIF reproduces this behaviour with the break in the slope behind the structure, as shown in figure [4.5](#). This result can also be seen in figure [4.7](#) and it was also illustrated in the chapter [1](#) in the section [4](#), as well as in the results in the chapter [3](#). For morphodynamic results with the multi-1D shoaling models (figure [4.5](#) left), non-linearities are observed between transects. This is due to the variability between transects (figure [4.5](#) left/bottom), as the multi-1D shoaling model calculates the hydrodynamics on each transect without taking 2D effects into account.

To obtain consistent morphodynamic results, it is very important to choose the associated hydrodynamic model carefully. It must be chosen according to the field of study (type of beach, type of experiment, type of waves, etc.) with the help of specialists. Indeed, the results of figure [4.7](#) show that two completely different wave models (multi-1D shoaling and REF/DIF), produce completely different morphodynamics. For example, the breaking of the REF/DIF wave model is very steep and therefore induces localised sediment transport below this wave breaking (justification in the section [5.4](#)). This is in contrast to the multi-1D model, which has a smoother breaking, resulting in greater sediment movement behind the geotube. This strategy is in line with [Murray \(2007\)](#) who explains that there is no universal morphodynamic model, and that it is necessary to choose carefully on a case-by-case basis.

5 Conclusion

In this chapter, we looked at the implementation of the OptiMorph 2D model. Due to the global scale of this model, implementation is very straightforward. The 2D model is validated in a similar way to the 1D model. An application on the Copter 2D configuration ([Bouchette 2017](#)) has been performed. In this case, the geotextile tube acts like a sedimentary bar and as a result, a trough is formed behind this geotube. The model has been able to reproduce this behaviour qualitatively. Dealing with complex problems, this model can therefore be used in coastal engineering involving wave dissipation structures. However, the choice of the wave model must be made with greater knowledge of the study case to avoid obtaining inconsistent morphodynamic results. In the future, it will be very important for users to choose the 2D wave model carefully.

Chapter key points

- Straightforward implementation of the 2D model.
- Several applications using multi-1D Shoaling and REF/DIF wave models.
- A model capable of managing complex structures on the seabed.
- Morphodynamic sensitivity due to wave models.

CHAPTER

5

OptiMorph 2.0 User Guide

In this user guide, we present the new version of the OptiMorph model based on the coastal hydro-morphodynamics by minimization principle. This morphodynamic model can be coupled with any hydrodynamic model. We therefore present how to couple this model with hydrodynamic models such as XBeach or SWAN.

Current chapter contents

1	Introduction	131
1.1	About	131
1.2	Expectation and Objectives	131
1.3	Target Audience	131
2	Processes and Theoretical Formulation	132
2.1	Domain and Definitions	132
2.1.1	Tide	132
2.2	Hydrodynamic Models	133
2.2.1	Extended Shoaling Model	133
2.2.2	XBeach Model	135
2.2.3	SWAN Model	137
2.3	Morphodynamic Model by Wave Energy Minimization	138
2.3.1	Introduction	139
2.3.2	Governing Equation of Seabed Dynamics	139
2.3.3	Parameter Y	139
2.3.4	Parameter Λ	140
2.3.5	Direction of Descent d	141
2.3.6	Choice of Cost Function \mathcal{J}	142
2.3.7	Hadamard Derivative to Compute $\nabla_{\psi}\mathcal{J}$	142
2.3.8	Slope Limiter	143
2.4	Model Constraints	145
2.4.1	Slope Constraint	145
2.4.2	Sand Stock Constraint	146
3	Numerical Model	147
3.1	Presentation	147
3.1.1	Workflow	147
3.1.2	Program Organization	148
3.2	Running OptiMorph	151

3.2.1	Installation of OptiMorph	151
3.2.2	Installation of PAGURE (to install SWAN and XBEACH) . . .	152
3.2.3	Input File	161
4	Applications	163
4.1	1D Linear Seabed Beach Configuration using Hadamard approach with SWAN	164
4.1.1	Setting	164
4.1.2	Input files	164
4.1.3	Load SWAN and Run OptiMorph on Cluster	166
4.1.4	Results	166
4.2	1D Linear Seabed Beach with GeoTube using Hadamard approach with XBeach	169
4.2.1	Setting	169
4.2.2	Input Files	169
4.2.3	Load XBeach and Run OptiMorph on Cluster	171
4.2.4	Results	171
4.3	2D Linear Seabed Beach Configuration using Hadamard approach with Shoaling	174
4.3.1	Setting	174
4.3.2	Input Files	174
4.3.3	Results	176

1 ntroduction

1.1 About

The numerical hydro-morphodynamic model presented here embodies a new approach to coastal morphodynamics, based on optimization theory. The model we present here follows on from the model presented by [Cook et al. \(2021a\)](#) in his user guide. This one is also based on the assumption that a sandy seabed evolves over time in order to minimize a certain wave-related function, the choice of which depends on what is considered the driving force behind coastal morphodynamics. This numerical model was given the name OptiMorph, and has the advantages of being fast, robust, and requires very few input parameters.

The model we present in this user guide is based on the same principles as [Cook et al. \(2021a\)](#) model. However, this model has been recreated with new enhancements, which we present in this user guide.

1.2 Expectation and Objectives

The main goal of OptiMorph was to demonstrate the potential of using optimal control in the modeling of coastal dynamics by designing an adaptable, easy-to-use numerical model. Our model aims to be generic, fast, robust and easy to use. It is intended to act as a morphodynamic module that can be coupled with any hydrodynamic model. This model aims to simulate morphodynamic phenomenology (creation of sedimentary bars) very well. It could eventually be incorporated into morphodynamic models without this phenomenological aspect.

1.3 Target Audience

The OptiMorph model is a tool intended for any person wishing to simulate the natural evolution of the coastal seabed in response to the incoming wave conditions, and/or to study the effect of submerged man-made devices on the sediment transport. It can be used by engineers seeking an opinion on the morphodynamic aspect of a project. By students wishing to understand morphodynamic phenomena. By a morphodynamic developer looking to improve his model...

2 Processes and Theoretical Formulation

2.1 Domain and Definitions

We consider a coordinate system composed of a horizontal axis x and a vertical axis z . We denote $\Omega := [0, x_{\max}]$ the domain of the cross-shore profile of the active coastal zone, where $x = 0$ is a fixed point in deep water where no significant change in bottom elevation can occur, and x_{\max} is an arbitrary point at the shore beyond the shoreline, as shown by Figure 5.1. The elevation of the sea bottom is a one-dimensional positive function, defined by: $\psi : \Omega \times [0, T_f] \times \Psi \rightarrow \mathbb{R}^+$ where $[0, T_f]$ is the duration of the simulation (s) and Ψ is the set of physical parameters describing the characteristics of the beach profile. In order to model the evolution over time of ψ and given the assumption that ψ changes over time in response to the energy of shoaling waves, a description of the surface waves is needed.

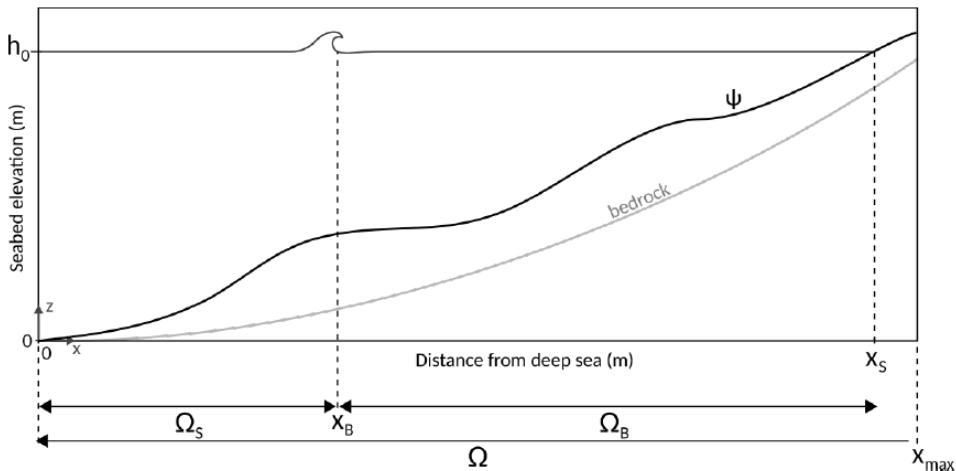


Figure 5.1 – Illustration of the cross-shore profile where breaking occurs once at $x = x_B$.

i Here, we present only part 1D. For the 2D part, the notations are analogous.

2.1.1 Tide

In this model, some parameters are time-variable: $T_0(t)$, $h_0(t)$, $H_0(t), \dots$. The choice of a temporally variable closure depth $h_0(t)$ allows managing the effect of tides. This is defined as follows:

$$h_0(t) = \frac{M_{eff}}{2} \sin \left(\frac{2\pi t}{T_{tide}} \right) \quad \text{with} \quad Ma = \frac{C_{tide} M_{ref}}{100} \quad (5.1)$$

with M_{ref} the reference tidal range (m): for example, in Brest, it is 6.1 m. M_{eff} the effective tidal range (m), C_{tide} the tidal coefficient and T_{tide} the tide duration (s).

2.2 Hydrodynamic Models

Unlike the previous user guide of Optimorph (Cook et al. 2021a), this one will focus solely on 3 hydrodynamic models. Indeed, numerical advances have enabled us to couple hydrodynamic models that are well known in the literature. We no longer need to develop models ourselves. The first is a purely numerical model based on the W. Munk (1949) criterion: this model enables the code to be tested very quickly. The other two models are XBeach (D. J. Roelvink et al. 2009; Zimmermann et al. 2012; Bugajny et al. 2013; Williams et al. 2015) and SWAN (Booij et al. 1996).

2.2.1 Extended Shoaling Model

The Shoaling model (Cook 2021) did not succeed to model wave breaking with wave periods $T_0 > 2\text{ s}$. This model was therefore improved to give birth to the extended model below:

Extended Shoaling model

$$H(x, t) = \begin{cases} H_0(x, t)K_S(x, t) & \text{for } x \in \Omega_S \\ \mathcal{F}(\gamma h(x, t)) & \text{for } x \in \Omega_B \end{cases} \quad (5.2a)$$

$$(5.2b)$$

where \mathcal{F} is a numerical parameterization function of the breaking define below (5.3):

$$\mathcal{F}(\gamma h(x, t)) = H(x_{start}) + [H(x_{stop}) - H(x_{start})] \cdot f\left(\frac{x - x_{start}}{x_{stop} - x_{start}}\right) \cdot g\left(\frac{h_{max} - h}{h_{max} - h_{min}}\right) \quad (5.3)$$

with $x \in \Omega_B = [x_{start}, x_{stop}]$, $h \in [h_{min}, h_{max}]$ and the following notations:

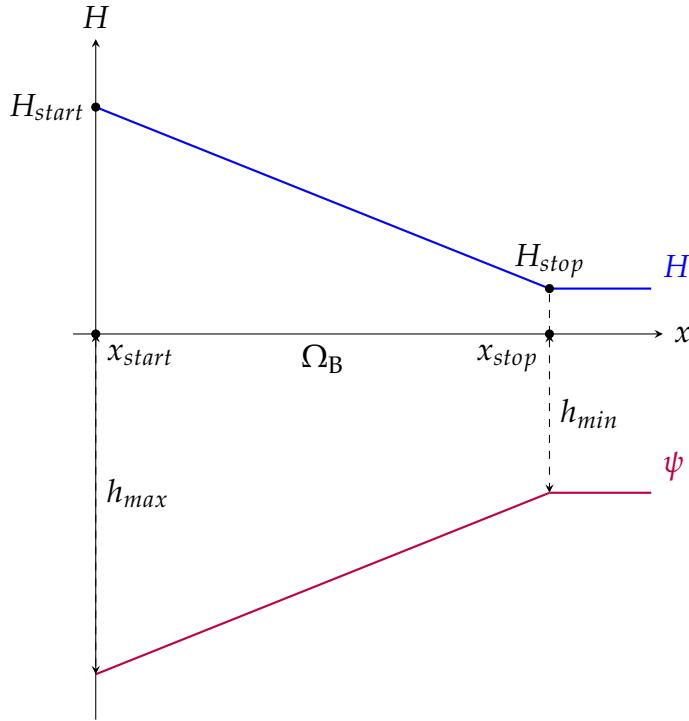


Figure 5.2 – Illustration of notations.

H_{start} and H_{stop} are the water heights at the beginning and end of the breaking on the domain $\Omega_B = [x_{start}, x_{stop}]$. The first function f gives an account of the breaking without taking into account the bed shape. It simply gives the appearance of the breaking. The second function g takes into account the seabed and interacts with it. Note that if f and g are the affine functions $x \mapsto x$, we find the breaking $\gamma h(x, t)$ illustrated on figure 5.2. We can present below (figure 5.3) some of these functions that set the breaking:

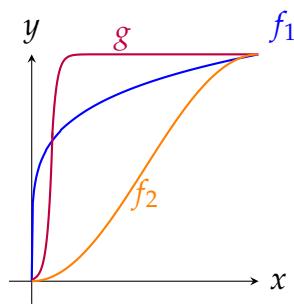


Figure 5.3 – Illustration of f_1 , f_2 and g defined in $[0, 1] \rightarrow [0, 1]$.

These functions were chosen to try to capture a natural breaking. They have no physical meaning.

It is necessary to stipulate that the model will first locate all the Ω_B domains and then apply the equation (5.3) on each of them. The result of this model can be seen with the breaker shown in figure 5.29.

2.2.2 XBeach Model

The XBeach model is a process-based model developed by the Delft University of Technology. It is a two-dimensional, depth-integrated numerical model that simulates the hydrodynamics, sediment transport, and morphological changes of coastal systems. XBeach is a flexible model that can be used to simulate a variety of coastal processes, including wave breaking, bedload transport, and nearshore morphological changes. The model is based on the principles of conservation of mass, momentum, and energy and uses a finite-difference numerical scheme to solve the governing equations. XBeach has been widely used in coastal studies due to its flexibility and accuracy, and it has been applied to a wide range of coastal systems, including estuaries, beaches, and coastal wetlands. The model can be used as a profile model in 1D ([Pender et al. 2013](#)), or as an area model in 2D ([McCall et al. 2010](#)), and today, there are three modes in which the hydrodynamics can be resolved in XBeach, being:

- **Stationary** – All wave group variations, and thereby all infragravity motions, are neglected, and only the mean motions are included. This type can be applied for modeling morphological changes under moderate wave conditions;
- **Surfbeat** – This in-stationary, hydrostatic mode, is wave group resolving, and is hence also applicable when one is interested in the swash zone processes;
- **Non-hydrostatic** – The non-linear shallow water equations are solved, and hence individual short-wave propagation and transformation is resolved.

In our case, we will focus on the **Stationary** mode.

2.2.2.1 Hydrodynamics

The wave action balance is solved to obtain the wave forcing:

$$\frac{\partial A}{\partial t} + \frac{\partial c_x A}{\partial x} + \frac{\partial c_y A}{\partial y} + \frac{\partial c_\theta A}{\partial \theta} = -\frac{D_w}{\sigma} \quad (5.4)$$

Where A is the wave action, C the wave propagation speed (where the subscripts refer to the x - and y -directions, and θ -space), θ is the angle of incidence, D_w the wave energy dissipation per directional bin and σ the intrinsic wave frequency. The wave action as above (5.5) by:

$$A(x, y, t, \theta) = \frac{S_w(x, y, t, \theta)}{\sigma(x, y, t)} \quad (5.5)$$

In which the S_w is the wave energy density per directional bin. The total wave energy E_H is obtained by integration of the wave energy density S_w over all directional bins:

$$E_H = \int_0^{2\pi} S_w(x, y, t, \theta) d\theta \quad (5.6)$$

The distribution of the total wave energy dissipation \bar{D}_w over all directional bins is calculated proportional to the energy density distribution as follows:

$$D_w(x, y, t, \theta) = \frac{S_w(x, y, t, \theta)}{E_w(x, y, t)} \bar{D}_w(x, y, t) \quad (5.7)$$

The total wave energy dissipation is calculated using a method described by [D. J. Roelvink \(1993\)](#) as the product of the dissipation per breaking event and the fraction of broken waves Q_b . The energy dissipation per wave breaking event is assumed to take place over half of the representative wave period T_{rep} , resulting in the following expression for the total, directionally integrated, wave energy dissipation:

$$\bar{D}_w = \alpha \frac{2}{T_{rep}} Q_b E_H \quad (5.8)$$

Where α is a calibration factor and E_w the total wave energy (Equation (5.6)). The fraction of breaking waves Q_b is estimated from a Rayleigh distribution ([Battjes et al. 1978](#)):

$$Q_b = 1 - \exp \left(- \left(\frac{H_{rms}}{H_{max}} \right)^n \right) \quad (5.9)$$

Where the root-mean-square wave height H_{rms} is calculated from the wave energy E_H , and the maximum wave height H_{max} is calculated using the breaker index γ (the ratio between the breaking wave height and the water depth, usually given the value 0.88).

$$E_H \sim \frac{1}{8} \rho g H_{rms}^2 \quad \Rightarrow \quad H_{rms} = \sqrt{\frac{8E_w}{\rho g}}, \quad H_{max} = \gamma_b h \quad (5.10)$$

This closes the set of equations for the wave action balance (Equation (5.4)). From the wave energy, the wave-induced radiation stresses can be determined using linear wave theory. Similar to the wave action balance, a roller balance is solved and coupled to the wave energy balance, where the wave energy dissipation forms a source of energy in the roller balance. The roller-induced radiation stress is calculated and together with the wave-induced radiation stress they are used to calculate the wave forcing: The flows are calculated using a depth-averaged formulation of the shallow water equations, taking into account wave-induced mass flux and return flows. This Generalized Lagrangian Mean

(GLM) formulation uses Lagrangian velocities ([Andrews et al. 1978](#)):

$$\frac{\partial u^L}{\partial t} + u^L \frac{\partial u^L}{\partial x} + v^L \frac{\partial u^L}{\partial y} - fv^L - v_h \left(\frac{\partial^2 u^L}{\partial x^2} + \frac{\partial^2 u^L}{\partial y^2} \right) = \frac{T_{sx}}{\rho h} - \frac{T_{bx}^E}{\rho h} - g \frac{\partial \eta}{\partial x} + \frac{F_x}{\rho h} \quad (5.11a)$$

$$\frac{\partial v^L}{\partial t} + u^L \frac{\partial v^L}{\partial x} + v^L \frac{\partial v^L}{\partial y} + fu^L - v_h \left(\frac{\partial^2 v^L}{\partial x^2} + \frac{\partial^2 v^L}{\partial y^2} \right) = \frac{T_{sy}}{\rho h} - \frac{T_{by}^E}{\rho h} - g \frac{\partial \eta}{\partial y} + \frac{F_y}{\rho h} \quad (5.11b)$$

$$\frac{\partial \eta}{\partial t} + \frac{\partial u^L h}{\partial x} + \frac{\partial v^L h}{\partial y} = 0 \quad (5.11c)$$

Where the Lagrangian velocity components (denoted by the superscript L) are the superposition of the Eulerian velocity and the Stokes' drift velocity:

$$u^L = u^E + u^S \quad \text{and} \quad v^L = v^E + v^S \quad (5.12)$$

2.2.3 SWAN Model

The SWAN model is a spectral numerical model designed to simulate waves evolving in coastal regions, lakes, and estuaries under defined wind, bathymetry, and current conditions. It is based on the Energy Density Balance equation ([5.5](#)) linking the advection term to the source and sink terms. Therefore, the wave energy evolves in both geographic and spectral space and changes its aspect due to the presence of wind at the surface, friction with the bottom, or during the breaking of the waves. The SWAN model is a stable model based on unconditionally stable numerical schemes (implicit schemes). SWAN, in its third version, is in stationary mode (optionally non-stationary) and is formulated in Cartesian or spherical coordinates. The unconditional numerical stability of the SWAN model makes its application more effective in shallow water. In SWAN, the waves are described with the two-dimensional spectrum of the wave action density A ,

$$A(x, y, \sigma, \theta) = \frac{E(x, y, \sigma, \theta)}{\sigma} \quad (5.13)$$

where x and y are the horizontal components of geographic space, σ is the relative frequency, θ is the wave direction, and E is the energy density.

The spectrum considered in the SWAN model is that of the wave action density $A(\sigma, \theta)$ rather than the spectrum of the energy density $E(\sigma, \theta)$. This is because, in the presence of currents for the reasons we mentioned above (non-conservation of E_H) ([G. B. Whitham 2011](#)). Because wave action propagates in both geographic and spectral space under the influence of genesis and dissipation terms, wave characteristics are described in terms of two-dimensional wave action density. The action density spectrum balance equation relating the propagation term to the effects of the source and sink terms, in Cartesian

coordinates, is ([Hasselmann et al. 1973](#))

$$\frac{\partial A}{\partial t} + \frac{\partial (C_x A)}{\partial x} + \frac{\partial (C_y A)}{\partial y} + \frac{\partial (C_\sigma A)}{\partial \sigma} + \frac{\partial (C_\theta A)}{\partial \theta} = \frac{S}{\sigma}. \quad (5.14)$$

On the left-hand side of Equation (5.14), the first term represents the local temporal variation of the wave action density, the second and third terms represent the propagation of wave action in the geographical space of velocities C_x and C_y , the fourth term represents the shifting of the relative frequency due to variations in bathymetry (with propagation velocity C_σ) and currents (with propagation velocity C_θ), and the fifth term represents the refraction induced by the combined effects of depth and currents. $C_x, C_y, C_\sigma, C_\theta$ propagation velocities are obtained from linear wave theory. The term in the right-hand side of Equation (5.14) represents processes that generate, dissipate, or redistribute wave energy, and S can be expressed as ([Lv et al. 2014](#))

$$S = S_{in} + S_{wc} + S_{brk} + S_{bot} + S_{n14} + S_{n13} \quad (5.15)$$

where S_{in} is the wind energy input. The dissipation terms of wave energy is represented by the contribution of three terms: the white capping S_{wc} , bottom friction S_{bot} , and depth induced breaking S_{brk} . S_{n14} and S_{n13} represent quadruplet interaction and triad interactions, respectively.

A finite difference scheme is used for each of the five dimensions: time, geographic space, and spectral space made the numerical implementation in SWAN effective. The following guidelines must be followed in order to obtain the discretization adopted at the SWAN model level for numerical computation:

1. time of a constant and identical time step Δt for the propagation term and the source term,
2. geographical space of a rectangular grid with constant spatial steps Δx and Δy ,
3. spectral space of a constant directional step $\Delta\theta$ and a constant relative frequency step $\Delta\sigma/\sigma$,
4. frequencies between a fixed minimum maximum values of 0.04 Hz and 1 Hz respectively,
5. the direction θ can also be delimited by the minimum and maximum values θ_{\min} and θ_{\max} (as an option).

2.3 Morphodynamic Model by Wave Energy Minimization

2.3.1 Introduction

The fundamental assumption governing OptiMorph states that the seabed evolves over time so as to minimize a certain quantity, named cost function. The choice of cost function depends on what is considered the driving force behind the morphodynamic response to the seabed. The one we've chosen calculates wave energy. In other words, the shape of the seabed varies in an effort to minimize the energy of the surface waves at that given time. At each time, the model indicates the direction to a local minimum of the cost function with regard to the parameterization of the seabed. Two physical parameters limit or encourage seabed mobility depending on the proprieties of the sediment and the depth of the water. This optimization problem is subjected to a limited number of constraints, allowing for a more accurate description of the morphodynamic evolution. The first concerns the maximal slope of the seabed, the second manages the sand stock of the profile in the case of an experimental flume.

The optimization problem that OptiMorph seeks to solve is:
For each $t \in [0, T]$, find the shape ψ of the seabed such that the cost function \mathcal{J} is minimal, while subjected to constraints.

The \mathcal{J} calculation is performed using a hydrodynamic model selected from those presented here.

2.3.2 Governing Equation of Seabed Dynamics

The evolution of the sea bottom is assumed to be driven by the minimization of a cost function \mathcal{J} (J s m^{-1}) with the following gradient descent taking the initial sea bottom ψ_0 ,

$$\begin{cases} \psi_t(., t) = Y \Lambda d(., t) \\ \psi(., 0) = \psi_0(.). \end{cases} \quad (5.16)$$

where ψ_t is the evolution of the bottom elevation over time (m s^{-1}), Y is a measure of the sand mobility expressed in m s kg^{-1} , Λ measures the excitation of the seabed by the orbital motion of water waves, and d is the direction of the descent (J s m^{-2}), which indicates the manner in which the sea bottom changes. This approach uses two parameters and two constraints.

2.3.3 Parameter Y

The first parameter Y takes into account the physical characteristics of the sand and represents the mobility of the sediment. Simulations with varying Y that reflect variations of the d_{50} grain diameter from 0.25 mm to 2 mm were performed. Changes in the beach profile were observed but no significant alteration of the trends in beach profile evolution

through time. The asymptotic behavior of the simulations remains the same although the velocity at which a given profile is reached changes. Further explanation of the nature of the Y parameter will be given at a later stage of the model development. For Y great, as is the case with finer particles, the seabed may be submitted to significant change. For Y close to zero, little mobility is observed.

2.3.4 Parameter Λ

The first constraint Y takes into account the physical characteristics of the sand and represents the mobility of the sediment. The second parameter Λ is a local function which represents the influence of the relative water depth kh on the beach profile dynamics and is defined after the term describing the vertical attenuation of the velocity potential according to linear wave theory ([R.L. Soulsby 1987](#)):

$$\begin{aligned} \varphi : \Omega \times [0, h_0] &\longrightarrow \mathbb{R}^+ \\ (x, z) &\longmapsto \frac{\cosh(k(x)(h(x) - (h_0 - z)))}{\cosh(k(x)h(x))} \end{aligned} \quad (5.17)$$

An illustration of the orbital velocity of the wave particles is given in figure 5.4. This function describes the excitation of the water particles for a given location along the cross-shore profile and a given water depth. However, our interest lies in the excitation of the seabed by the surface waves. Therefore, it is natural to consider the orbital damping function at $z = \psi(x)$. The parameter Λ of equation (5.16) is therefore defined by:

$$\Lambda(x) = \varphi(x, \psi(x)) = \frac{1}{\cosh(k(x)h(x))} \quad (5.18)$$

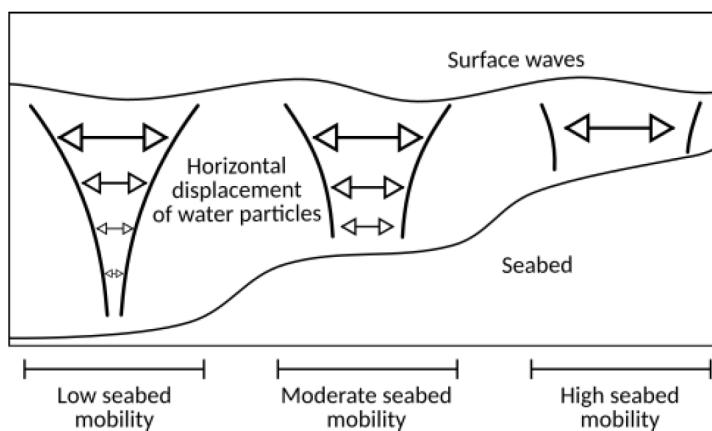


Figure 5.4 – Illustration of the orbital velocity over the cross-shore profile from ([Cook 2021](#))

This parameter governs the manner in which the waves affect the seabed. In deeper waves, the surface waves have little to no effect on the seabed below. No movement should be observed of the seabed, and thus $\Lambda \approx 0$ over this portion of the cross-shore profile. When the waves have a large impact on the seabed, e.g. at the coast, greater movement can be observed and as such we set $\Lambda \approx 1$. An illustration of Λ is given in figure 5.4.

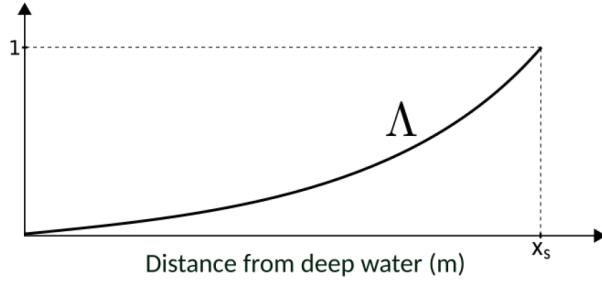


Figure 5.5 – Variation of the parameter Λ over the cross-shore profile from (Cook 2021)

2.3.5 Direction of Descent d

d is the direction of the descent ($J \text{ s m}^{-2}$), which indicates the manner in which the sea bottom changes. In unconstrained configurations, there would be $d = -\nabla_{\psi} J$, which by its definition indicates the direction of a local minimum of J with respect to ψ as illustrated on figure 5.6.

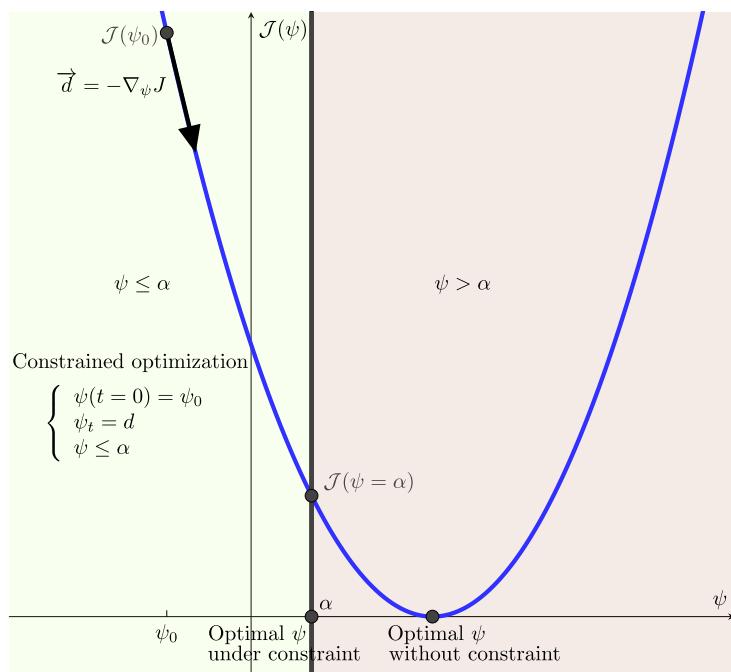


Figure 5.6 – Illustration of gradient descent with $\psi \leq \alpha$. The optimum does not necessarily correspond to the critical point $\nabla_\psi \mathcal{J} = 0$

The tricky step will be to obtain this quantity $\nabla_\psi \mathcal{J}$: this is explained in the section [2.3.7](#) with Hadamard’s derivation.

2.3.6 Choice of Cost Function \mathcal{J}

The shape of the beach profile is determined by the minimization of the potential energy of waves, for all $t \in [0, T_f]$:

$$\mathcal{J}(\psi, t) = \frac{1}{16} \int_{t-T_{coupl}}^t \int_{\Omega_S} \rho_w g H^2(\psi, x, \tau) dx d\tau \quad (5.19)$$

where H denotes the height of the waves over the cross-shore profile (m), ρ_w is water density (kg m^{-3}), and g is the gravitational acceleration (m s^{-2}). T_{coupl} (s) defines the coupling time interval between hydrodynamic and morphodynamic models so that we have T_f/T_{coupl} iterations.

2.3.7 Hadamard Derivative to Compute $\nabla_\psi \mathcal{J}$

We use the approximation described in ([Hadamard 1914](#); [Mohammadi 2007](#); [Mohammadi 2010](#)). We consider $\nabla_\psi \mathcal{J}$ in the sense of Hadamard following the definition:

$$\nabla_\psi \mathcal{J} = \lim_{\varepsilon \rightarrow 0} \frac{\mathcal{J}(\psi + \varepsilon n) - \mathcal{J}(\psi)}{\varepsilon}, \quad (5.20)$$

where n is the normal to the shape ψ . This can be seen as applying a [Gâteaux \(1913\)](#) derivation in the direction normal to the shape. The principle is illustrated in figure [5.7](#).

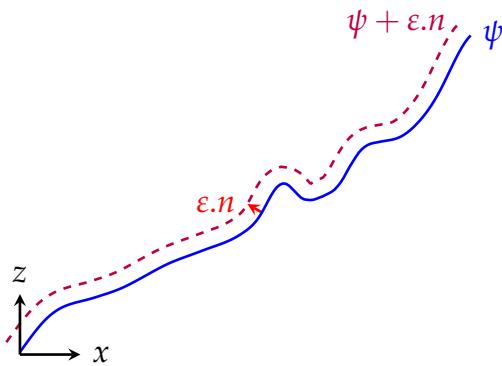


Figure 5.7 – Representation of two sea bottom profiles ψ and $\psi + \varepsilon n$. To calculate the gradient, we need to calculate at all points the associated normal vector n .

Using the Taylor-Young formula at order 1, we consider the following approximation:

$$\begin{aligned}\nabla_\psi \mathcal{J} &= \lim_{\varepsilon \rightarrow 0} \frac{\mathcal{J}(\psi) + \varepsilon \nabla_X \mathcal{J} \cdot n - \mathcal{J}(\psi)}{\varepsilon}, \\ &\approx (\nabla_X \mathcal{J}) \cdot n,\end{aligned}\tag{5.21}$$

with $X = (x, z)^\top$. To implement this approach practically, we simply need to use the equation (5.21) with: $\nabla_X \mathcal{J} = \begin{pmatrix} \frac{\partial \mathcal{J}}{\partial x} \\ \frac{\partial \mathcal{J}}{\partial z} \end{pmatrix}$ and $n = \frac{1}{\sqrt{d\psi^2 + dx^2}} \begin{pmatrix} -d\psi \\ dx \end{pmatrix}$ and we obtain:

$$\nabla_\psi \mathcal{J} \approx \frac{\partial \mathcal{J}}{\partial x} n_x + \frac{\partial \mathcal{J}}{\partial z} n_z,\tag{5.22}$$

with n_x and n_z the x and z component of n . $\frac{\partial \mathcal{J}}{\partial x}$ and $\frac{\partial \mathcal{J}}{\partial z}$ are calculated using finite differences. The $\Delta\psi$ quantity can sometimes be almost zero, depending on the configuration of the sea bottom. To avoid code explosions, we'll use a slope limiter.

2.3.8 Slope Limiter

A slope limiter was introduced in Hadamard differentiation, and helps maintain stable code. This is based on the following algorithm:

This limiter is applied every time differentiation is calculated in Hadamard. This limiter is very effective, as shown in the figure 5.8.

Algorithm 1 A slope limiter

Input: y is the vector to limit, n the size of y , n_x the maximum limitation window, often $n_x=20$

Output: y without degeneration

```
1: error ← 1
2: for Δx=1,nx do
3:   y0 ← y
4:   for i=Δx,n - Δx do
5:     ymin ← min(y0[i - Δx], y0[i + Δx])
6:     ymax ← max(y0[i - Δx], y0[i + Δx])
7:     y[i] ← max[min(y0[i], ymax), ymin]
8:   end for
9:   error0 ← error
10:  error ← ||y - y0||
11:  if Δx = 1 then
12:    e0 ← error
13:  end if
14:  error ←  $\frac{\text{error}}{e_0}$ 
15:  if Δx > 1 and error > error0 then
16:    y ← y0
17:    break
18:  end if
19: end for
```

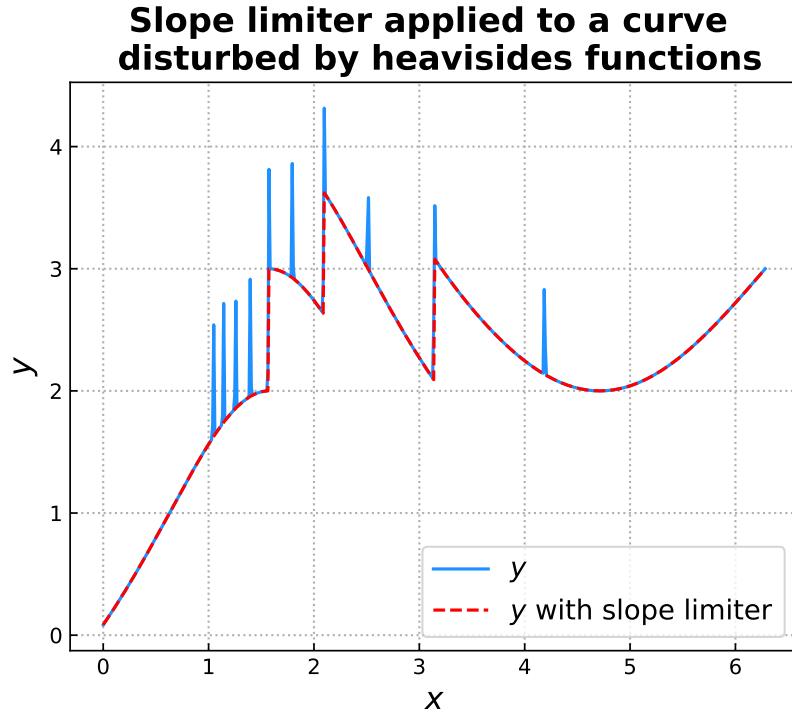


Figure 5.8 – Slope limiter applied to a curve disturbed by heavisides functions

2.4 Model Constraints

In the interest of simplicity, we have adopted two physical constraints though more can be introduced if necessary.

2.4.1 Slope Constraint

The first concerns the local slope of the bottom. Depending on the composition of the sediment, the bottom slope is bounded by a grain-dependent threshold M_{slope} (Dean et al. 2004). This is conveyed by the following constraint on the local bottom slope illustrated by 5.9:

$$\left| \frac{\partial \psi}{\partial x} \right| \leq M_{\text{slope}} \quad (5.23)$$

The dimensionless parameter M_{slope} represents the critical angle of repose of the sediment. This angle is based on observed angles in natural beach environments, which are often between 0.01 and 0.2 (Bascom 1951; Vos et al. 2020; Short 1996). We have considered the observed critical angle of 0.2.

2.4.2 Sand Stock Constraint

A second constraint concerns the sand stock in the case of an experimental flume. In a flume, the quantity of sand must be constant over time, as given by (5.24), contrarily to an open-sea configuration where sand can be transported between the nearshore zone and a domain beyond the closure water depth where sediment is definitely lost for beach morphodynamics (Hattori et al. 1980; Quick 1991). This constraint can be written as :

$$\int_{\Omega} \psi(t, x) dx = \int_{\Omega} \psi_0(x) dx \quad \forall t \in [0, T_f] \quad (5.24)$$

This constraint is necessary for verifying and validating the numerical model with the wave flume experimental data.

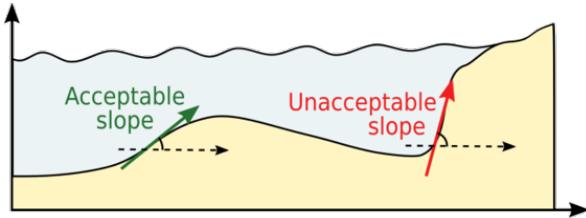


Figure 5.9 – Slope constraint (5.23) from (Cook 2021)

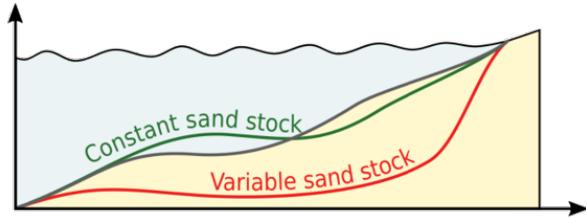


Figure 5.10 – Sand conservation (5.24) from (Cook 2021)

2.4.2.1 Numerical implementation of projections

For the numerical implementation, we introduce a new quantity C_{sand} . For a given time $t \in [0, T_f]$, $C_{\text{sand}}(t)$ is a difference between the current and the initial sand stock, weighted by ψ :

$$C_{\text{sand}}(t) = \int_{\Omega} (\psi - \psi_0)^2 d\Omega. \quad (5.25)$$

To have a sand conservation (equation (5.24) and figure 5.10), we must have $C_{\text{sand}}(t) = 0$. The minimization problem then becomes:

For each $t \in [0, T]$, find the shape ψ of the seabed such that the cost function \mathcal{J} is minimal, while maintaining $C_{\text{sand}}(t) = 0$.

The projection method chosen to satisfy this constraint is the same as for (Cook 2021).

Since $C_{\text{sand}}(0) = 0$, we wish to minimize \mathcal{J} while keeping C_{sand} constant. This equates to following the direction $\nabla_{\psi} \mathcal{J}$ while keeping $\nabla_{\psi} C_{\text{sand}} = 0$. In order to do so, we project the direction $\nabla_{\psi} \mathcal{J}$ onto the orthogonal of $\nabla_{\psi} C_{\text{sand}}$. Hence, the direction of

the descent d becomes:

$$d = \nabla_\psi \mathcal{J} - \left\langle \nabla_\psi \mathcal{J}, \frac{\nabla_\psi C_{\text{sand}}}{\|\nabla_\psi C_{\text{sand}}\|} \right\rangle \frac{\nabla_\psi C_{\text{sand}}}{\|\nabla_\psi C_{\text{sand}}\|}, \quad (5.26)$$

with

$$\nabla_\psi C_{\text{sand}}(t) = 2\psi \int_\Omega (\psi - \psi_0) dx. \quad (5.27)$$

This new direction of the descent, illustrated by Figure 5.11, describes a less optimal path to the minimum of \mathcal{J} , but ensures that $\nabla_\psi C_{\text{sand}}(t) = 0$, i.e. $C_{\text{sand}}(t) = 0$, for all $t \in [0, T]$. We can easily show that the new direction d and $\nabla_\psi C_{\text{sand}}$ are now orthogonal:

$$\langle d, \nabla_\psi C_{\text{sand}} \rangle = \left\langle \nabla_\psi \mathcal{J} - \left\langle \nabla_\psi \mathcal{J}, \frac{\nabla_\psi C_{\text{sand}}}{\|\nabla_\psi C_{\text{sand}}\|} \right\rangle \frac{\nabla_\psi C_{\text{sand}}}{\|\nabla_\psi C_{\text{sand}}\|}, \nabla_\psi C_{\text{sand}} \right\rangle = 0 \quad (5.28)$$

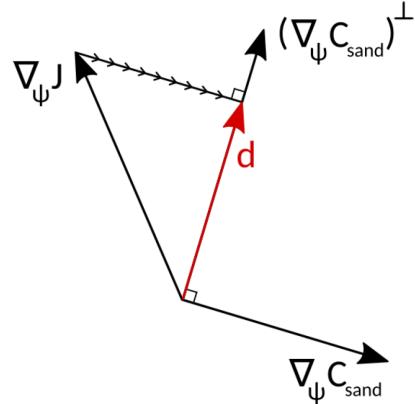


Figure 5.11 – Illustration of the new direction of the descent in \mathbb{R}^2 : the direction $\nabla_\psi \mathcal{J}$ is projected onto the orthogonal of C_{sand} to yield d .

3 Numerical Model

3.1 Presentation

In this section, we present the model OptiMorph, how to install and use it.

3.1.1 Workflow

Figure 5.12 illustrates the workflow of the OptiMorph model, with the associated hydrodynamic model. Prior to initiating the model, the user is required to establish

the initial configuration for the simulation. This includes the forcing data, the choice of hydrodynamic model, the seabed elevation data, and the constraints.

During each discrete time step, the forcing data is provided to the hydrodynamic model. This model then calculates the wave height over the cross-shore profile and thus provides the cost function \mathcal{J} (or direction of the descent d) used by OptiMorph's morphodynamic module. Using the imported sand characteristics, the new shape of the seabed is determined by minimizing the cost function \mathcal{J} (or following the direction of the descent d). Constraints are applied to the seabed either before or after the minimization takes place, and the new seabed is retained. At the next time step, the hydrodynamic model is fed a new forcing condition as well as the new seabed. This cycle continues over the course of the simulation, and illustrates the intricate interaction between the hydrodynamic and morphodynamic processes.

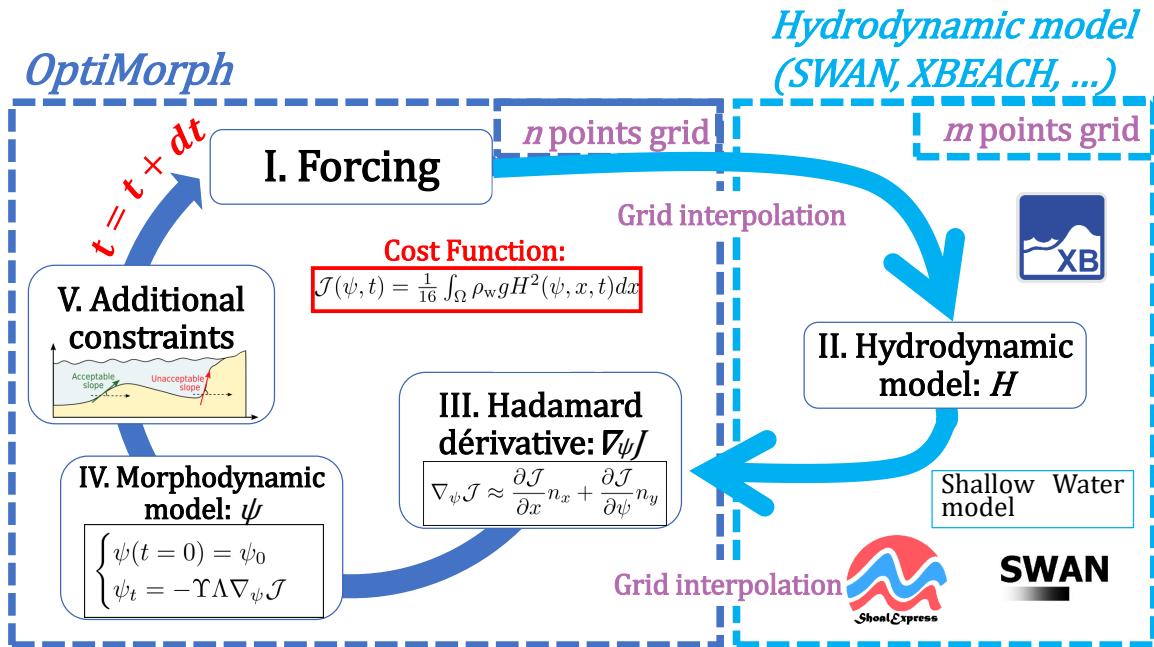


Figure 5.12 – OptiMorph workflow coupled with hydrodynamic model

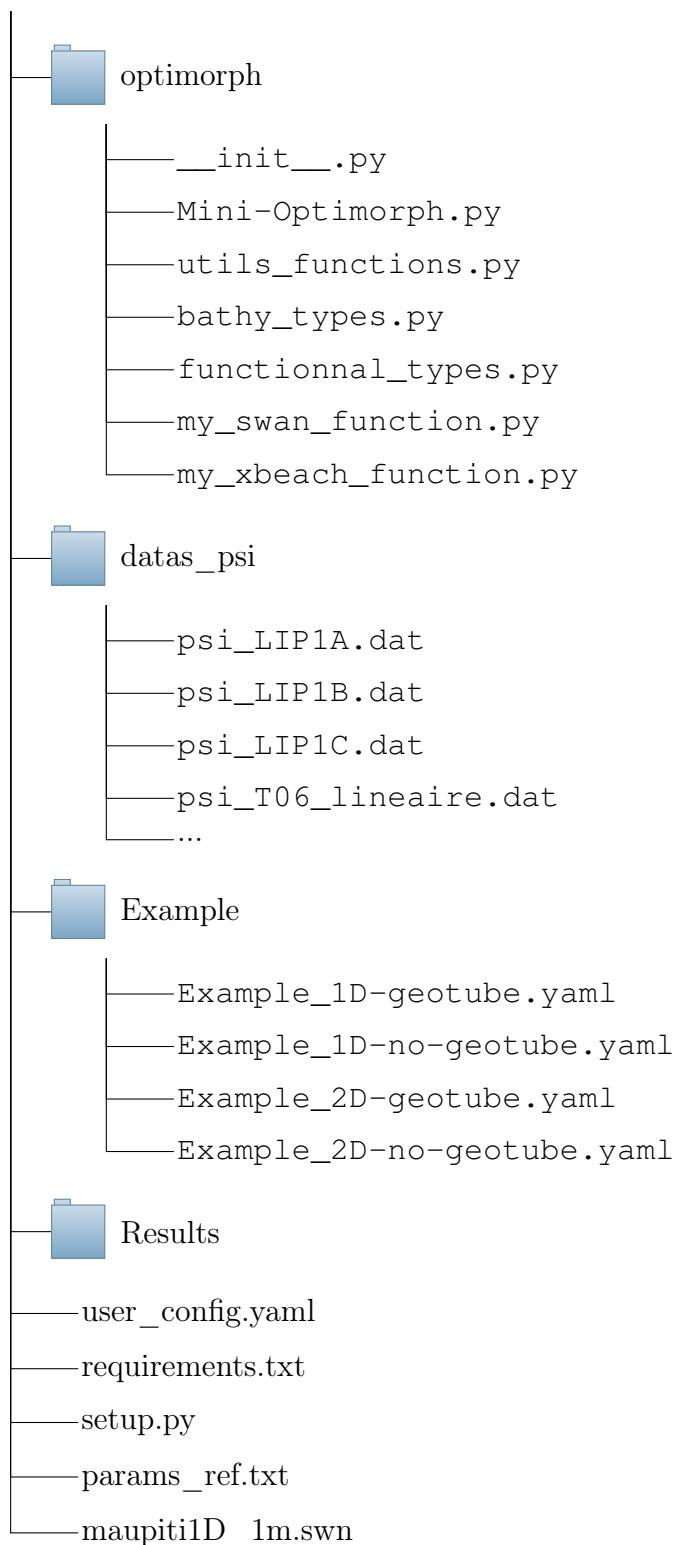
3.1.2 Program Organization

The OptiMorph program is broken down into the following tree structure. There are 4 main folders:

- a folder `optimorph` which is the heart of the program, all the code architecture (functions, governing equation, ...) is contained in it, the `Mini-Optimorph.py` program must be run to start the calculation;

- a `datas_psi` folder containing several types of bathymetry in `.dat` format;
- a `Results` folder where files and figures will be stored;
- a folder `Example` with files `user_config.yaml` already preconfigured.

To run a calculation, you need to change the parameters in the `user_config.yaml` file.



3.2 Running OptiMorph

OptiMorph is installed in 3 stages. First, the basic OptiMorph code is installed. Next, we need numerical models such as SWAN or XBeach to couple our code to them. For these installations, we'll use Pagure, which makes them easier.

3.2.1 Installation of OptiMorph

The code we provide uses python version >3.7 . To install it, please, download the last 1D version using the command line on terminal:

```
\$ git clone https://oauth2:github_pat_11A7K63PA01ZnQdd@GwmZyy_KREFJ3NVsX2QZyR3oswTmNG99sFbEKMT2wUWYCVN5yLJKWOQKGQzqa1wtvn@github.com/rupont/OptiMorph-1-2D.git
```

If you don't have a terminal, you can always download the latest version from the following link: <https://ronan-dupont.github.io/files/teaching/OptiMorph-1-2D.zip>.

If you don't have git installed, you can install with the line:

```
\$ sudo apt install git
```

Then, you need to install the following modules on your computers (if you don't already have them):

- numpy
- matplotlib
- scipy
- pandas
- xarray
- pyyaml
- imageio
- Pillow

You can do it in one line with on unix / cmd (windows) terminal.

```
\$ pip install -r requirements.txt
```

With just this setup, we can run calculations using the Shoaling model. However, if we want to couple our OptiMorph model with SWAN, XBeach or other software, we need to install these programs.

If you have all the module installed and you'd like to make the first launch, you can run the following command:

```
\$ python3 optimorph/Mini-Optimorph.py
```

3.2.2 Installation of PAGURE (to install SWAN and XBEACH)

The first step is to install a software program called PAGURE¹, developed by a former GLADYS doctoral student and postdoctoral fellow (Fabien Rétif²). This software will collect all the libraries needed to run the digital tools, and link them in their correct version with the compiler chosen to compile on the cluster.

First, we connect to the cluster:

```
\$ ssh e_gcl-XX@muse-login01.hpc-lr.univ-montp2.fr
```

From now on, all the commands presented in this section are to be entered on the cluster, not on your local machine. We use git – which we'll see in detail later – to retrieve pagure.

1. <https://github.com/fretif/pagure>
2. <https://www.fabienretif.com>

```
\$ cd
\$ mkdir install-softs
\$ cd install-softs
\$ mkdir pagure.git
\$ cd pagure.git
\$ git clone https://github.com/fretif/pagure.git .
Cloning into '.'...
remote: Enumerating objects: 2000, done.
remote: Counting objects: 100% (334/334), done.
remote: Compressing objects: 100% (231/231), done.
remote: Total 2000 (delta 229), reused 200 (delta 103), pack-reused
→ 1666
Receiving objects: 100% (2000/2000), 4.67 MiB | 0 bytes/s, done.
Resolving deltas: 100% (1475/1475), done.
\$ ./pagure.sh # affiche un message de PAGURE par défaut
```

PAGURE is now installed in the directory /install-softs/pagure.git.

3.2.2.1 Configuring your work environment

Before using PAGURE, it's important to configure your working environment. To do this, we're going to use an environment manager called `module` (yes, that's a funny name!). To illustrate the usefulness of an environment manager, I suggest the following scenario:

Imagine you've installed version 2.18 of QGis on your computer and your colleague shares a file with you that works with version 3.20. What do you do? You can update QGis to version 3.20. But you run the risk that your files won't open with the new version, or that some features have disappeared. This is where the environment manager comes into play, allowing you to have both versions co-exist on your system at the same time. You can then choose to load version 2.18 into your working environment and switch to another version at any time.

An environment manager is useful when you need complete control over your working environment and the flexibility to run multiple versions of the same software. This is perfectly justified in the case of a cluster or a workstation dedicated to numerical computation.

In most computers, such as MESO, the `module` tool is supplied directly by the cluster administrators. On your local workstation, the `module` tool will probably not be installed, but PAGURE will detect it and install it automatically.

Without further ado, let's start using the `module` tool to configure our working en-

vironment on MESO.

```
\$ module avail

----- /usr/share/Modules/modulefiles
→ -----
dot           module-git   module-info modules      null          use.own

----- /trinity/shared/modulefiles/modulegroups
→ -----
cv-admin     cv-advanced  cv-local    cv-standard local

----- /trinity/shared/modulefiles/local
→ -----
.....
-----
→ /trinity/shared/modulefiles/cv-standard -----
....
intel/itac/64/2020.4.912      intel/mkl/32/2017.1.132
intel/compiler/32/2016.3.210   intel/compiler/64/2016.3.210
intel/compiler/32/2017.1.132   intel/compiler/64/2017.1.132
intel/compiler/32/2020.4.912   intel/compiler/64/2020.4.912
gcc/4.9.3(default)            intel/mkl/64/2020.4.912
gcc/6.1.0                      intel/omnipath/64/libpsm2-10.3.8-3
gcc/7.5.0                      git/2.9.3
gcc/8.5.0                      intel/mpi/64/2017.1.132
gdb/7.11                       intel/mpi/64/2020.4.912
....
```

This command lists all the libraries/software available on the cluster, with their versions and sometimes the name of the compiler editor used (GCC or Intel).

Let's start by resetting the working environment with the command :

```
\$ module purge
```

Then we'll load the modules `cv-standard use.own` and `intel/compiler/64/2017.1.132` with the command line:

```
\$ module load cv-standard use.own intel/compiler/64/2017.1.132
```

Let's check that the libraries have been loaded in our working environment:

```
\$ module list
Currently Loaded Modulefiles:
1) cv-standard      2) use.own      3) intel/compiler/64/2017.1.132
\$ ifort --version
ifort (IFORT) 17.0.1 20161005
Copyright (C) 1985-2016 Intel Corporation. All rights reserved.
```

We've loaded the 2017 version of the Intel brand compiler. Our environment is ready!

3.2.2.2 Installing the SWAN model

To install the SWAN template, you must first configure your working environment with the 2017 version of the INTEL compiler (see section 3.2.2.1). Next, we'll run PAGURE with a set of arguments corresponding to this model.

```
\$ ./pagure.sh --prefix=/home/e_gcl-XX/softs --system=cluster
--compiler=intel --filter=SWAN
↪ --module-dir=/home/e_gcl-XX/privatemodules
```

This command will compile and install a set of modules (in the sense of module software) in the privatemodules directory of your personal environment, download the SWAN software and compile and link it to these libraries, then install it in the soft directory of your personal environment.

On startup, PAGURE summarizes information about the working environment it has detected and the software it will install:

```
[ INFO ] system is set to cluster
[ INFO ] prefix is set to /home/e_gcl-XX/softs
[ INFO ] module dir is set to /home/e_gcl-XX/privatemodules
[ INFO ] Installation mode is set to auto
[ INFO ] Force to download is set to 0
[ INFO ] Force to reinstall is set to 0
[ INFO ] Auto-remove is set to 1
[ INFO ] Automatic installation of mandatory libraries is set to 1
[ INFO ] When using a filter, show old version is set to 1
[ INFO ] Python interpreter is set to python2.7
.....
[ INFO ] compiler is set to INTEL 17
[ INFO ] MPI library is set to mpich321
.....
[ OK ] Make dir prefix
.....
[ INFO ] The following libraries are pre-selected to be installed :
[ INFO ] mpich 3.2.1
[ INFO ] zlib 1.2.11 (needed by HDF5)
[ INFO ] parallel-netcdf 1.12.1 (needed by Netcdf 4.8.0)
[ INFO ] hdf5 1.10.5 (with parallel I/O)
[ INFO ] netcdf 4.8.0 (version C - need HDF 1.10.5 and
→ Parallel-Netcdf 1.12.1)
[ INFO ] netcdf 4.5.3 (version Fortran - need Netcdf-C 4.8.0, HDF
→ 1.10.5 and Parallel-Netcdf 1.12.1)
[ INFO ] swan 41.31
.....
[ OK ] We are now ready to install. Please check the information
→ above
.....
Everything is OK ? Press Enter to continue or press q to quit
```

At this point, PAGURE asks whether the information detected is correct before continuing its execution. You should therefore check that :

- the system detected is indeed that of a cluster

```
[ INFO ] system is set to cluster
```

- the prefix path, i.e. where all libraries useful to SWAN will be installed, is /home/e_gcl-XX/softs.

```
[ INFO ] prefix is set to /home/e_gcl-XX/softs
```

- the path of the modules directory, i.e. where the modules will be installed to load the SWAN libraries in your work environment, is /home/e_gcl-XX/privatemodules.

```
[ INFO ] module dir is set to /home/e_gcl-XX/privatemodules
```

- the detected compiler is the 2017 version of Intel.

```
[ INFO ] compiler is set to INTEL 17
```

- the parallel computing library (MPI) is Mpich

```
[ INFO ] MPI library is set to mpich321
```

If everything is OK, you can press Enter on your keyboard and wait for PAGURE to finish installing SWAN.

```
# un temps loooooooooooooonnnnnnnnnng! (sans doute 1h la
↪ première fois), puis:
[ INFO ] Removing archive file and source files
[ OK ] Install swan 41.31
[ OK ] Congratulation, you did it
```

If all has gone well, you can check the installation by listing the libraries with the module tool.

```
\$ module avail
...
-----/home/e_gcl-XX/privatemodules
↪ -----
netcdf-fortran/hdf5.110/mpich321/icc17/4.5.3
↪ hdf5/mpich321/icc17/1.10.5
netcdf-c/hdf5.110/mpich321/icc17/4.8.0
↪ parallel-netcdf/mpich321/icc17/1.12.1
zlib/icc17/1.2.11                               mpich/icc17/3.2.1
swan/mpich321/icc17/41.31
```

So you have a new module swan/mpich321/icc17/41.31

3.2.2.3 Installing the XBEACH model

To install the XBEACH template, we need to start by loading a new working environment.

Let's start by resetting the working environment with the command :

```
\$ module purge
```

Then we'll load the modules cv-standard use.own and gcc/7.5.0 with the following command:

```
\$ module load cv-standard use.own gcc/7.5.0
```

Let's check that the libraries have been loaded in our working environment:

```
\$ module list
Currently Loaded Modulefiles:
1) cv-standard      2) use.own      3) gcc/7.5.0
\$ gcc --version
gcc (GCC) 7.5.0
Copyright © 2017 Free Software Foundation, Inc.
Ce logiciel est un logiciel libre; voir les sources pour les
→ conditions de copie.
Il n'y a AUCUNE GARANTIE, pas même pour la COMMERCIALISATION ni
→ L'ADÉQUATION À UNE TÂCHE PARTICULIÈRE.
```

We've loaded the GNU/GCC brand compiler in version 7.5. Our environment is ready!

To install the XBEACH model in its sequential version, we'll run PAGURE with a set of arguments corresponding to this model.

```
\$ ./pagure.sh --prefix=/home/e_gcl-XX/softs --system=cluster
--compiler=gnu --filter=XBEACH
→ --module-dir=/home/e_gcl-XX/privatemodules
```

This command will compile and install a set of modules (in the sense of module software) in the privatemodules directory of your personal environment, download the XBEACH software and compile and link it to these libraries, then install it in the soft directory of your personal environment.

On startup, PAGURE summarizes the information about the working environment it has detected and the software it will install:

```
[ INFO ] system is set to cluster
[ INFO ] prefix is set to /home/e_gcl-XX/softs
[ INFO ] module dir is set to /home/e_gcl-XX/privatemodules
[ INFO ] Installation mode is set to auto
[ INFO ] Force to download is set to 0
[ INFO ] Force to reinstall is set to 0
[ INFO ] Auto-remove is set to 1
[ INFO ] Automatic installation of mandatory libraries is set to 1
[ INFO ] When using a filter, show old version is set to 1
[ INFO ] Python interpreter 3.7 will be installed
.....
[ INFO ] compiler is set to GNU 7.5
[ WARNING ] No MPI library
.....
[ OK ] Make dir prefix
.....
[ INFO ] The following libraries are pre-selected to be installed :
[ INFO ] python 3.7
[ INFO ] setuptools 57.0.0 (Python module)
[ INFO ] mako 1.2.0 (Python module)
[ INFO ] zlib 1.2.11 (needed by HDF5)
[ INFO ] hdf5 1.10.5
[ INFO ] netcdf 4.8.0 (version C - need HDF 1.10.5)
[ INFO ] netcdf 4.5.3 (version Fortran - need Netcdf-C 4.8.0 and
→ HDF 1.10.5)
[ INFO ] xbeach rev5920 (sequential version)
.....
[ OK ] We are now ready to install. Please check the information
→ above
.....
Everything is OK ? Press Enter to continue or press q to quit
```

At this point, PAGURE asks whether the information detected is correct before continuing its execution. You should therefore check that :

- the system detected is indeed that of a cluster

```
[ INFO ] system is set to cluster
```

- the prefix path, i.e. where all libraries useful to XBEACH will be installed, is /home/e_gcl-XX/softs

```
[ INFO ] prefix is set to /home/e_gcl-XX/softs
```

- the path of the modules directory, i.e. where the modules will be installed to load the XBEACH libraries in your work environment, is /home/e_gcl-XX/privatemodules.

```
[ INFO ] module dir is set to /home/e_gcl-XX/privatemodules
```

- the detected compiler is the GNU/GCC compiler in version 7.5

```
[ INFO ] compiler is set to GNU 7.5
```

If everything is OK, you can press Enter on your keyboard and wait for PAGURE to finish installing XBEACH. Compilation may take some time.

```
# un temps loooooooooooooonnnnnnnng! (sans doute 1h la
→ première fois), puis:
.....
Type the absolute path of the archive file 'xbeach-rev5920.zip' :
```

At this stage, you need to specify the path of the 'xbeach-rev5920.zip' file distributed to you (which can also be found at <https://drive.google.com/file/d/19Ngh9vfnkCzLdVgXjcCLview?usp=sharing>). For example, if you copied it to your home directory, you would specify: /home/e_gcl-XX

```
# un temps loooooooooooooonnnnnnnng! (sans doute 1h la
→ première fois), puis:
[ INFO ] Removing archive file and source files
[ OK ] Install xbeam rev5920 (sequential version)
[ OK ] Congratulation, you did it
```

If all has gone well, you can check the installation by listing the libraries with the module tool.

```
\$ module avail
...
-----/home/e_gcl-XX/privatemodules
→ -----
netcdf-fortran/hdf5.110/mpich321/icc17/4.5.3
→ hdf5/mpich321/icc17/1.10.5
netcdf-c/hdf5.110/mpich321/icc17/4.8.0
→ parallel-netcdf/mpich321/icc17/1.12.1
zlib/icc17/1.2.11                               mpich/icc17/3.2.1
swan/mpich321/icc17/41.31                      xbeach/gcc75/rev5920
```

So you have a new module xbeach/gcc75/rev5920 corresponding to the sequential version of XBEACH.

3.2.3 Input File

The input file for the OptiMorph code is the `user_config.yaml`. It is presented as follows:

user_config.yaml

```

dirname: Example_1D-convexe_tide # dirname to save figs
  ↳ and data
figname: Example Simulation 1D of storm in convexe
  ↳ bathymetry with tide # simulation name appear on figs
debug: False # this mode plot some interesting values
makeGifs: True # make gifs
T0: 6 # wave period
H0: 1.5 # offshore wave height
h0: 10
nwater: 600 # in the water
nsand: 140 # in the sand
n_iteration: 1000
ifre: 50 # save and plot every ifre iteration
mobility: 0.004
bathy_type: 1 # [0:18]
slope_max: 0.2 #
id_cost_fct: 1 # cost function [1:12]
hydro_mode: 1 # : 0 = shoaling, 1 = swan, 2 = XBeach
dynamic: False
gamma: 0.55
coef_maree: 60 # tide coefficient
u_maree: 6.1 # valeur moyenne du marnage
maree_duration: 12.5 # periode de marez
geotube:
  state: False
  position_x: 110 # geotube position x [m]
  position_y: 10 # geotube position y [m] ONLY FOR 2D MODE
  length: 6 # geotube length [m]
  height: 2 # geotube height [m]
two_dimension:
  state: False
  n_i: 300
  n_j: 60
  L_x: 600
  L_y: 20

```

with the description in the table **5.1** below.

Parameter	Type	Name	Description	Unit
dirname	String	-	Directory name	-
figname	String	-	Figure name	-
debug	Boolean	Debug mode	This mode plot some interesting values	-
makeGifs	Boolean	Make Gifs	This make a gif	-
T0	Float	T_0	Wave period	m
H0	Float	H_0	Offshore wave height	m
h0	Float	h_0	Depth of closure	m
Omega	Integer	Ω	Domain size	m
n_iteration	Integer	$n_{iteration}$	Number of iterations	-
ifre	Integer	-	Save/Plot every ifre	-
mobility	Float	Y	Mobility parameter	$m.s.kg^{-1}$
bathy_type	Integer	-	Sea Bottom type ranging [0,18]	-
slope_max	Float	M_{slope}	Maximum slope	-
id_cost_fct	Integer	-	Cost Function type ranging [1,12]	-
hydro_mode	Integer	-	Hydrodynamic mode, 0 = shoaling, 1 = swan, 2 = XBeach	-
dynamic	Boolean	-	Static (0) or Dynamic (1) forcing	-
gamma	Float	γ	Breaking criterion	-
coef_maree	Integer	C_{tide}	Tidal coefficient	-
u_maree	Float	M_{ref}	The reference tidal range	m
maree_duration	Float	T_{tide}	Tide duration	h
geotube:				
state	Boolean	-	Activate (True) or not (False)	-
position_x	Integer	-	Position x of geotube	m
position_y	Integer	-	Position y of geotube	m
length	Integer	-	Geotube length	m
height	Integer	-	Geotube height	m
two_dimension:				
state	Boolean	-	Activate (True) or not (False)	-
n_i	Integer	-	Number of points on x axis	-
n_j	Integer	-	Number of points on y axis	-
L_x	Integer	L_x	Length of x axis	m
L_y	Integer	L_y	Length of y axis	m

Table 5.1 – Input parameters

This file was created as described below. However, you can easily add parameters by opening the file `optimorph/Mini-Optimorph.py`.

4 Applications

In this section, the seabed is described as a simple linear function over the cross-shore profile. First, we simulate the results over a homogeneous sandy seabed, then we look at introducing submerged structures designed to limit wave activity at the coast. Finally, we study this last case in 2D on a linear seabed also inclined along y .

4.1 1D Linear Seabed Beach Configuration using Hadamard approach with SWAN

The applications focus on 3 different cases. These 3 cases aim to show how to use the code with the 3 different hydrodynamics as well as on 3 different configurations.

4.1.1 Setting

The initial cross-shore configuration is given in Figure 5.13: the domain measures 740 m, the mean water level is set at 10 m and we apply a storm profile to the seabed, given by the top left graph of Figure 5.13 . Here we consider a homogeneous sandy seabed, and therefore the mobility of the seabed Y and the maximal slope parameter M_{slope} slope are assumed constant over the cross-shore profile Ω .

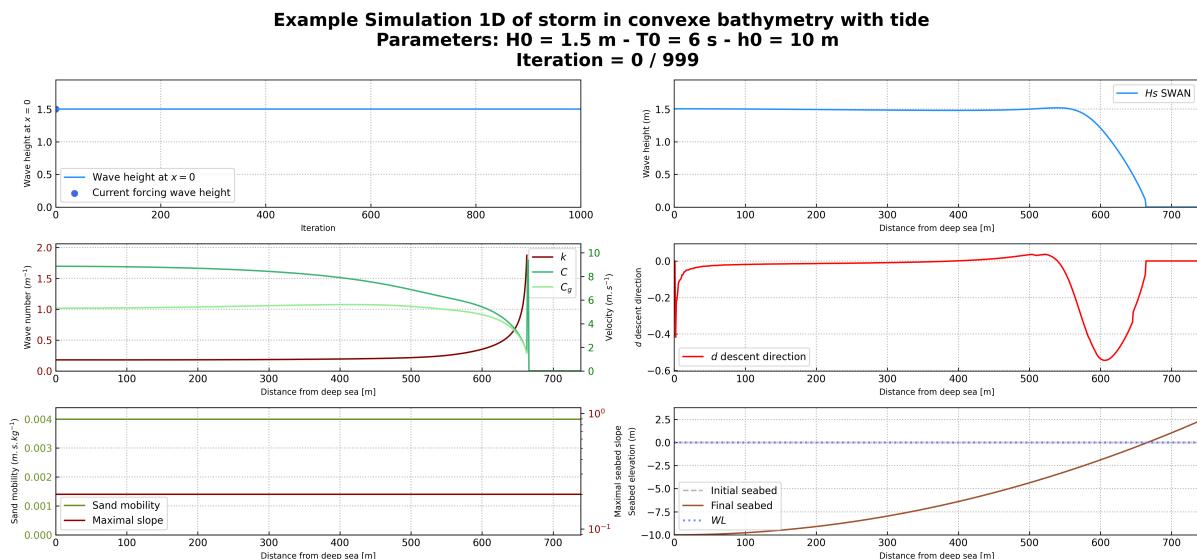


Figure 5.13 – Initial sandy beach configuration

4.1.2 Input files

The input file is present in the `user_config.yaml` file and is configured as follows. It can also be found at the location `Example/Example_1D-no-geotube_tide.yaml`.

user_config.yaml

```

dirname: Example_1D-convexe_tide # dirname to save figs
    ↳ and data
figname: Example Simulation 1D of storm in convexe
    ↳ bathymetry with tide # simulation name appear on figs
debug: False # this mode plot some interesting values
makeGifs: True # make gifs
T0: 6 # wave period
H0: 1.5 # offshore wave height
h0: 10
nwater: 600 # in the water
nsand: 140 # in the sand
n_iteration: 1000
ifre: 50 # save and plot every ifre iteration
mobility: 0.004
bathy_type: 1 # [0:18]
slope_max: 0.2 #
id_cost_fct: 1 # cost function [1:12]
hydro_mode: 1 # : 0 = shoaling, 1 = swan, 2 = XBeach
dynamic: False
gamma: 0.55
coef_maree: 60 # tide coefficient
u_maree: 6.1 # valeur moyenne du marnage
maree_duration: 12.5 # periode de maree
geotube:
    state: False
    position_x: 110 # geotube position x [m]
    position_y: 10 # geotube position y [m] ONLY FOR 2D MODE
    length: 6 # geotube length [m]
    height: 2 # geotube height [m]
two_dimension:
    state: False
    n_i: 300
    n_j: 60
    L_x: 600
    L_y: 20

```

4.1.3 Load SWAN and Run OptiMorph on Cluster

To launch OptiMorph with SWAN, you need to load SWAN into your modules. Here are a few commands that will enable you to run OptiMorph on the cluster without any problems:

```
\$ module purge
\$ module load use.own swan/mpich321/icc17/41.31
\$ module load python/3.7.2
\$ pip install --upgrade pip
\$ pip install -e .
\$ pip install -U matplotlib
\$ python3 optimorph/Mini-Optimorph.py
```

You can also create a bash to launch the file on slurm with the file below:

run.cmd

```
#!/bin/bash

# Example of running python script with a job array

#SBATCH -J Run_test
#SBATCH -p gm_gladys
#SBATCH --account=shoremotion
#SBATCH -c 1                                # one CPU core per task
#SBATCH -o console.out
#SBATCH -e erreur.out
#SBATCH -N 1
#SBATCH -n 1
#SBATCH --ntasks-per-node 1
#SBATCH --ntasks-per-core 1

# Run python script with a command line argument
srun python optimorph/Mini-Optimorph.py
```

then run it via the command:

```
\$ sbash run.cmd
```

4.1.4 Results

At the end of the simulation, we get the following results of Figure 5.13, 5.14, 5.15, 5.16, 5.17, 5.18, 5.19 and 5.20.

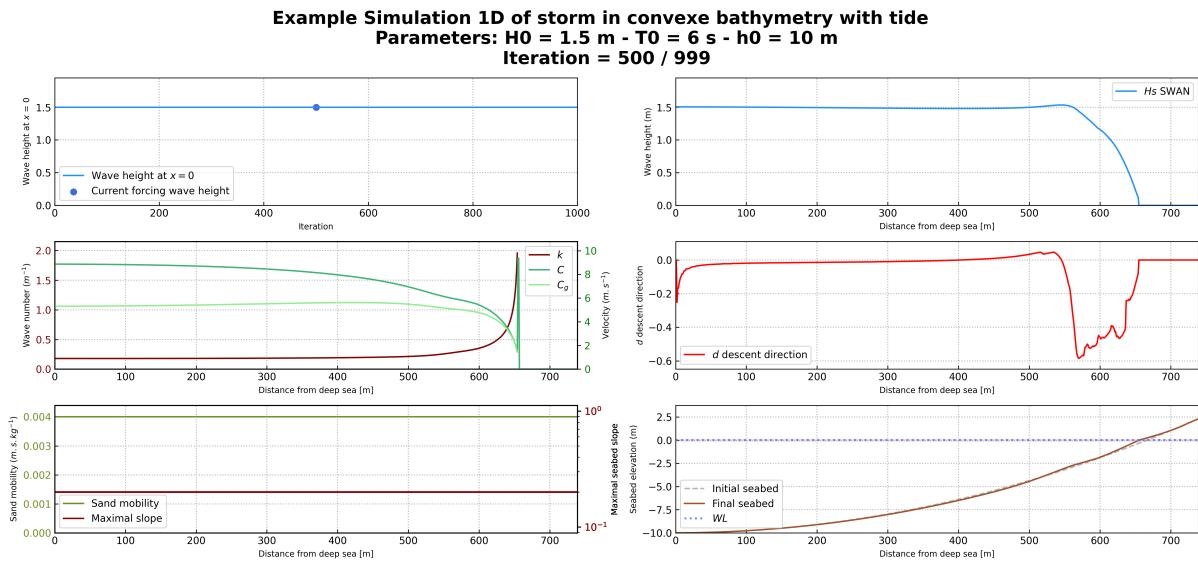


Figure 5.14 – Results halfway through the simulation

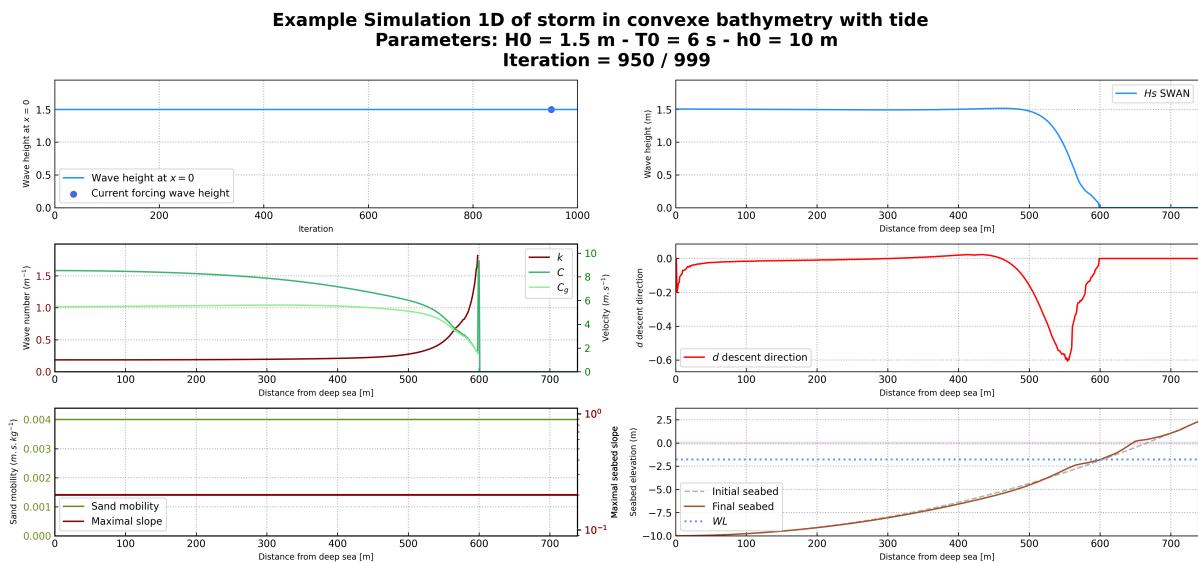


Figure 5.15 – Results at the end of the simulation

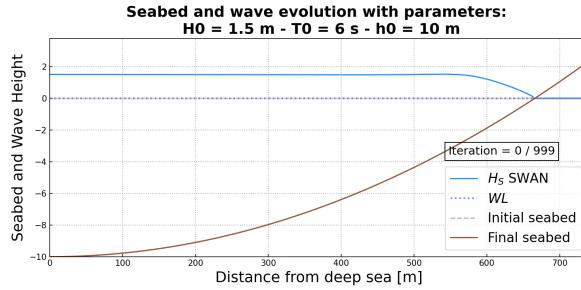


Figure 5.16 – Initial seabed at the beginning of the simulation

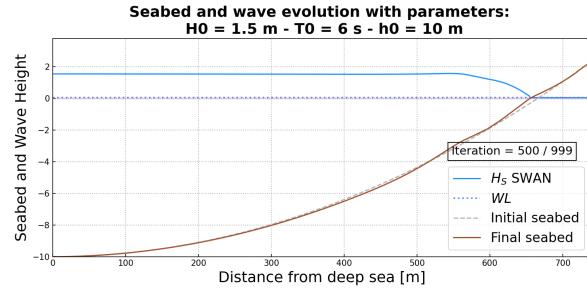


Figure 5.17 – Seabed halfway through the simulation

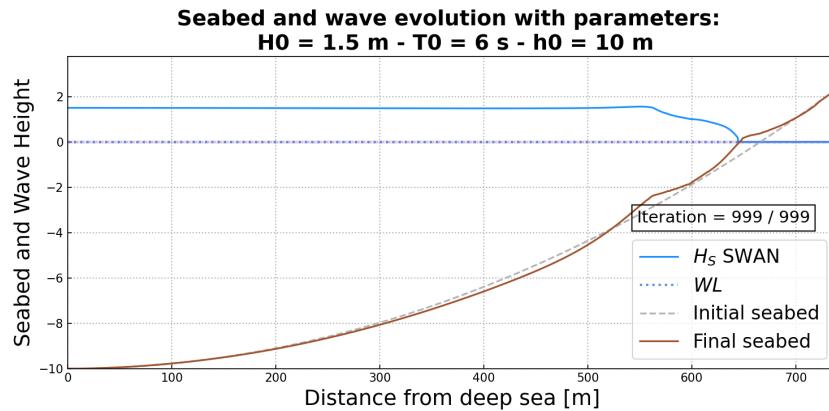


Figure 5.18 – Final seabed at the end of the simulation

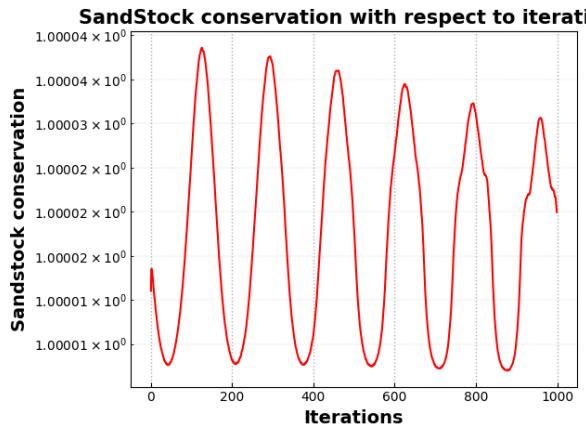


Figure 5.19 – Variation of the sand stock over time

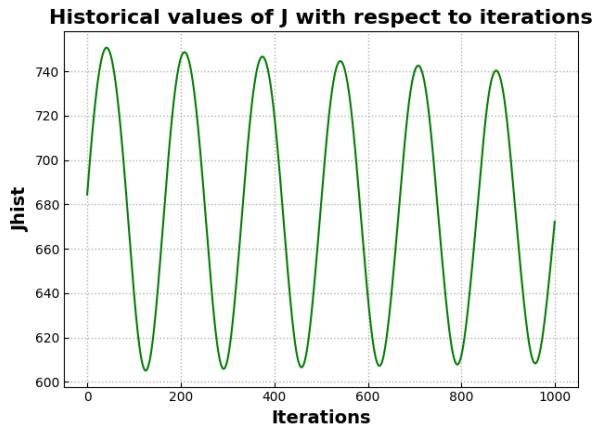


Figure 5.20 – Variation of d over time

A thorough analysis of the results of OptiMorph can be found in (Ronan Dupont et al. 2023) for an experimental flume configuration and (Ronan Dupont et al. 2022) for a linear seabed. Sand conservation has an error of the order of 10^{-5} .

4.2 1D Linear Seabed Beach with GeoTube using Hadamard approach with XBeach

4.2.1 Setting

In this simulation, we introduce a submerged solid structure. To do this, we modify the seabed profile, as well as the sand mobility parameter Y and the maximal slope parameter M_{slope} , which are no longer constant over the cross-shore profile. In the case of the mobility parameter, no movement can occur at the location of the structures, i.e. $Y = 0$ where the breakwater is positioned. Similarly, the maximal slope parameter has also been modified to locally deactivate the slope constraint over the structure. Figure 5.21 shows the new initial configuration incorporating a submerged breakwater located at $x = 600$ m.

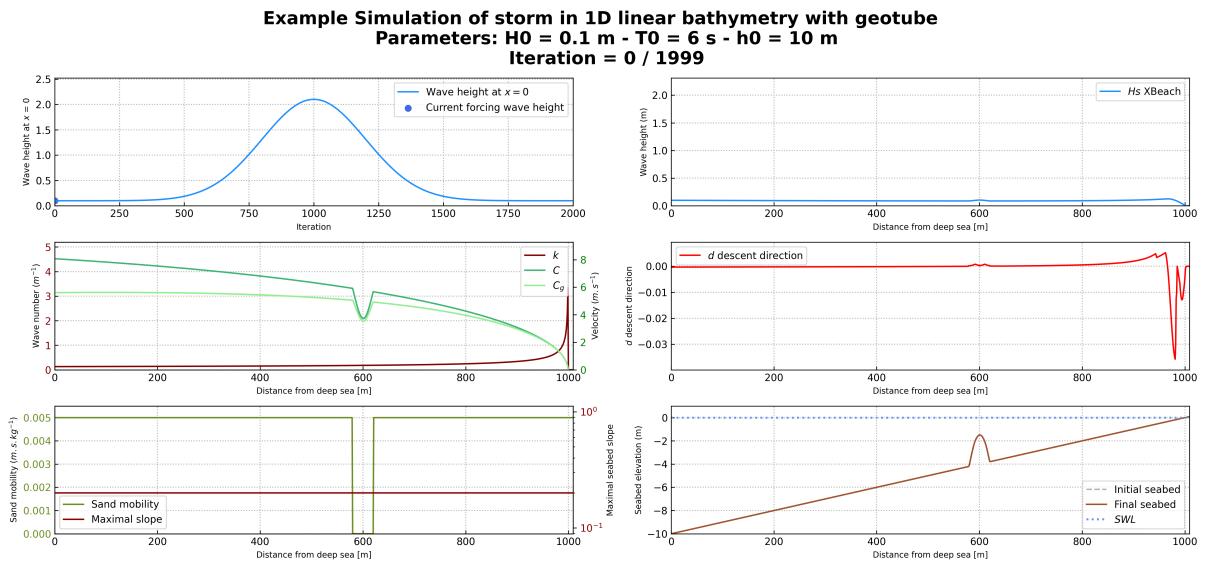


Figure 5.21 – Initial sandy beach configuration with a submerged breakwater located at $x = 600$ m

4.2.2 Input Files

The input file is present in the `user_config.yaml` file and is configured as follows. It can also be found at the location `Example/Example_1D-geotube.yaml`.

user_config.yaml

```

dirname: 1D-geotube_example # dirname to save figs
figname: Example Simulation of storm in 1D linear
    ↳ bathymetry with geotube # simulation name
debug: False # this mode plot some interesting values
makeGifs: True # make gifs
T0: 6 # wave period
H0: 2 # offshore wave height
h0: 10
Omega: 1000
n_iteration: 2000
ifre: 50 # save and plot every ifre iteration
mobility: 0.005
bathy_type: 0 # [0:18]
slope_max: 0.2 #
id_cost_fct: 1 # cost function [1:12]
hydro_mode: 2 # : 0 = shoaling, 1 = swan, 2 = XBeach
dynamic: True
gamma: 0.55
coef_maree: 0 # tide coefficient
u_maree: 6.1 # valeur moyenne du marnage
maree_duration: 12.5 # periode de maree
geotube:
    state: True
    position_x: 600 # geotube position x [m]
    position_y: 10 # geotube position y [m] ONLY FOR 2D MODE
    length: 40 # geotube length [m]
    height: 2.5 # geotube height [m]
two_dimension:
    state: False
    n_i: 300
    n_j: 60
    L_x: 600
    L_y: 20

```

4.2.3 Load XBeach and Run OptiMorph on Cluster

To launch OptiMorph with XBeach, you need to load XBeach into your modules. Here are a few commands that will enable you to run OptiMorph on the cluster without any problems:

```
\$ module purge
\$ module load use.own xbeach/gcc75/rev5920
\$ module load python/3.7.2
\$ pip install --upgrade pip
\$ pip install -e .
\$ pip install -U matplotlib
\$ python3 optimorph/Mini-Optimorph.py
```

You can also create a bash to launch the file on slurm with the file below:

run.cmd

```
#!/bin/bash

# Example of running python script with a job array

#SBATCH -J Run_test
#SBATCH -p gm_gladys
#SBATCH --account=shoremotion
#SBATCH -c 1                               # one CPU core per task
#SBATCH -o console.out
#SBATCH -e erreur.out
#SBATCH -N 1
#SBATCH -n 1
#SBATCH --ntasks-per-node 1
#SBATCH --ntasks-per-core 1

# Run python script with a command line argument
srun python optimorph/Mini-Optimorph.py
```

then run it via the command:

```
\$ sbash run.cmd
```

4.2.4 Results

At the end of the simulation, we get the following results of Figure 5.21, 5.22, 5.23, 5.24, 5.25, 5.26, 5.27 and 5.28.

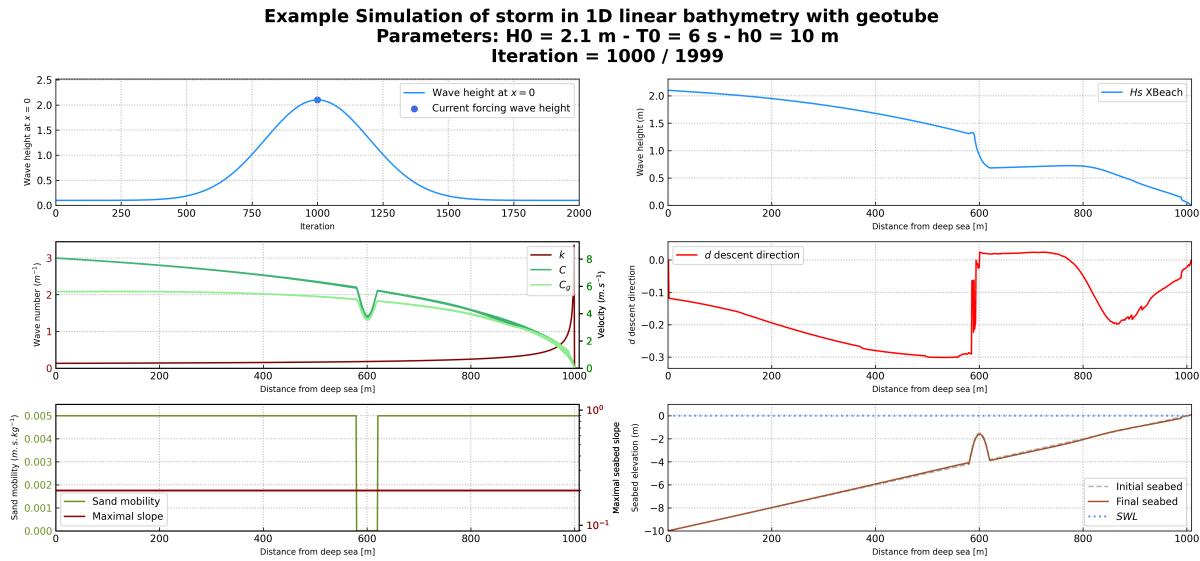


Figure 5.22 – Results halfway through the simulation

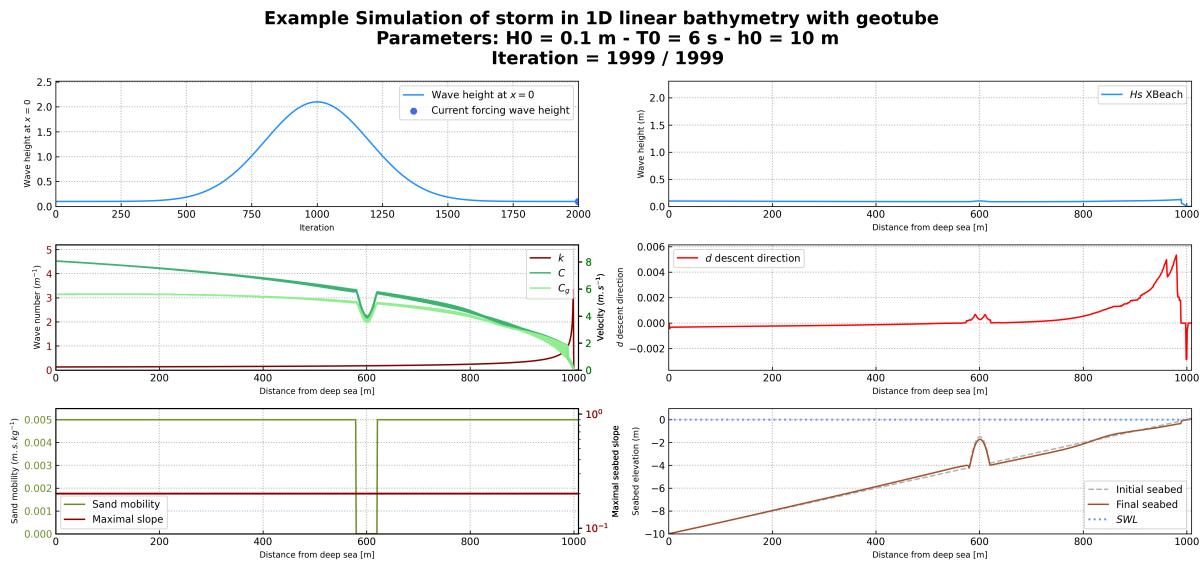


Figure 5.23 – Results at the end of the simulation

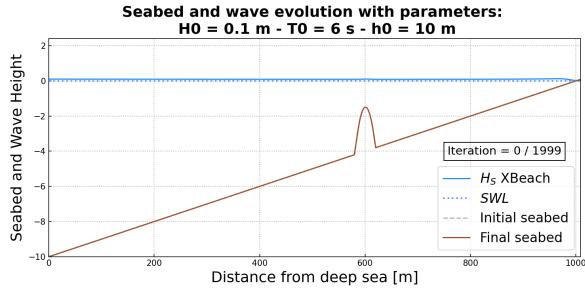


Figure 5.24 – Initial seabed at the beginning of the simulation

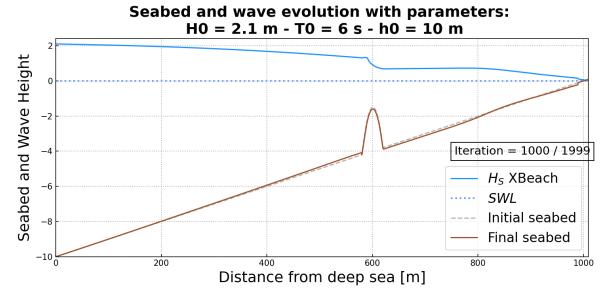


Figure 5.25 – Seabed halfway through the simulation

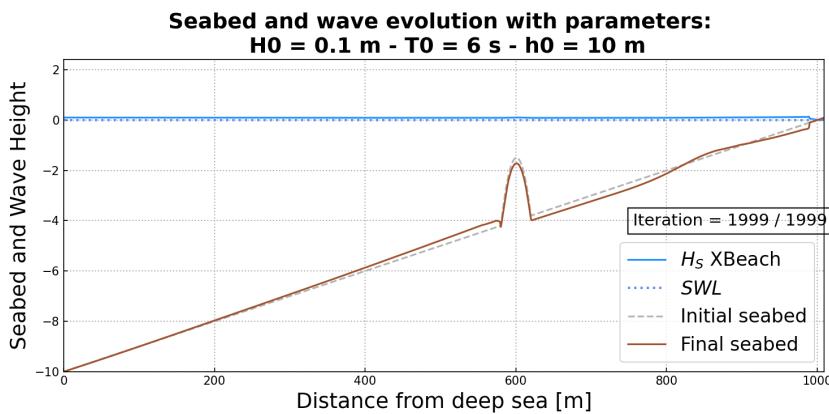


Figure 5.26 – Final seabed at the end of the simulation

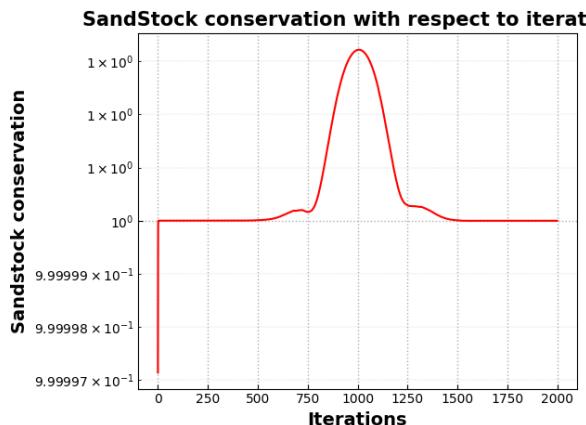


Figure 5.27 – Variation of the sand stock over time

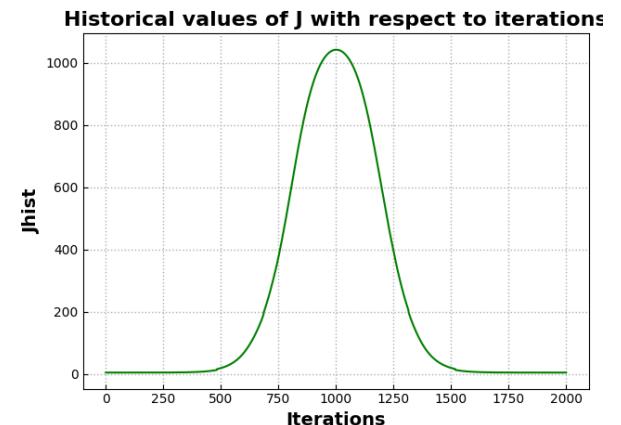


Figure 5.28 – Variation of d over time

We observe that in the case where there is a geotube (figure 5.26), there is less morphodynamic displacement than without a geotube (figure 5.26). Sand conservation has an error of the order of 10^{-5} .

4.3 2D Linear Seabed Beach Configuration using Hadamard approach with Shoaling

4.3.1 Setting

In this simulation, we continue with a submerged solid structure but in 2D. To do this, we modify the seabed profile, as well as the sand mobility parameter Y and the maximal slope parameter M_{slope} , which are no longer constant over the cross-shore profile. In the case of the mobility parameter, no movement can occur at the location of the structures, i.e. $Y = 0$ where the breakwater is positioned. Similarly, the maximal slope parameter has also been modified to locally deactivate the slope constraint over the structure. Figure 5.29 shows the new initial configuration incorporating a submerged breakwater located at $x = 600$ m.

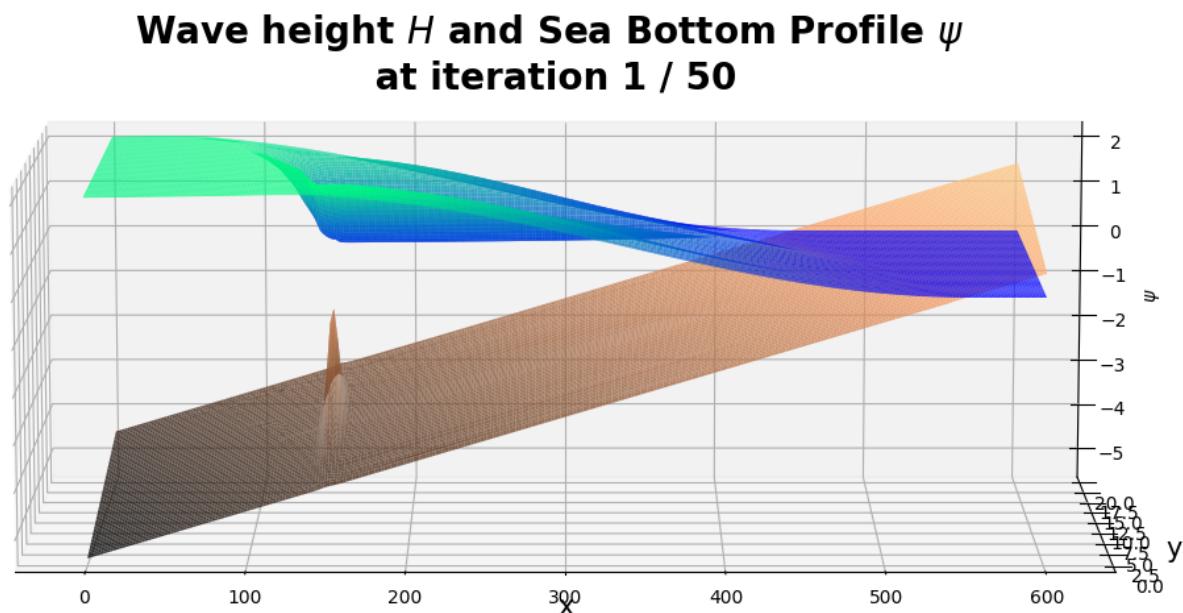


Figure 5.29 – Initial sandy beach configuration with a submerged breakwater located at $x = 150$ m

4.3.2 Input Files

The input file is present in the `user_config.yaml` file and is configured as follows. It can also be found at the location `Example/Example_2D-geotube.yaml`.

user_config.yaml

```

dirname: plot_test_2D # dirname to save figs
figname: Simulation of storm in linear bathymetry with
    ↳ geotube # simulation name
debug: False # this mode plot some interesting values
makeGifs: True # make gifs
T0: 6 # wave period
H0: 2 # offshore wave height
h0: 5.5
Omega: 600
n_iteration: 50
ifre: 5 # save and plot every ifre iteration
mobility: 0.05
bathy_type: 0 # [0:18]
slope_max: 0.2 #
id_cost_fct: 1 # cost function [1:12]
hydro_mode: 0 # : 0 = shoaling, 1 = swan, 2 = XBeach
dynamic: True
gamma: 0.55
coef_maree: 0 # tide coefficient
u_maree: 6.1 # valeur moyenne du marnage
maree_duration: 12.5 # periode de mareex
geotube:
    state: True
    position_x: 150 # geotube position x [m]
    position_y: 10 # geotube position y [m] ONLY FOR 2D MODE
    length: 40 # geotube length [m]
    height: 2.5 # geotube height [m]
two_dimension:
    state: True
    n_i: 300
    n_j: 60
    L_x: 600
    L_y: 20

```

4.3.3 Results

At the end of the simulation, we get the following results of Figure 5.31, 5.32, 5.29, 5.33 and 5.34.

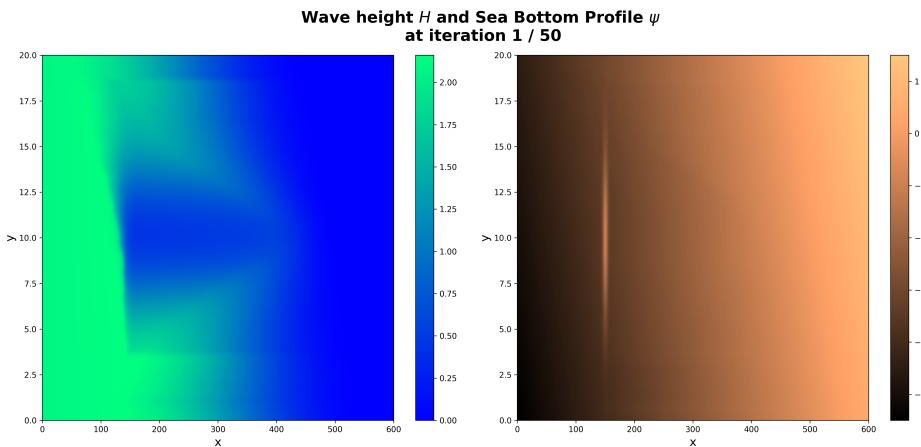


Figure 5.30 – Initial sandy beach configuration with a submerged breakwater located at $x = 150$ m

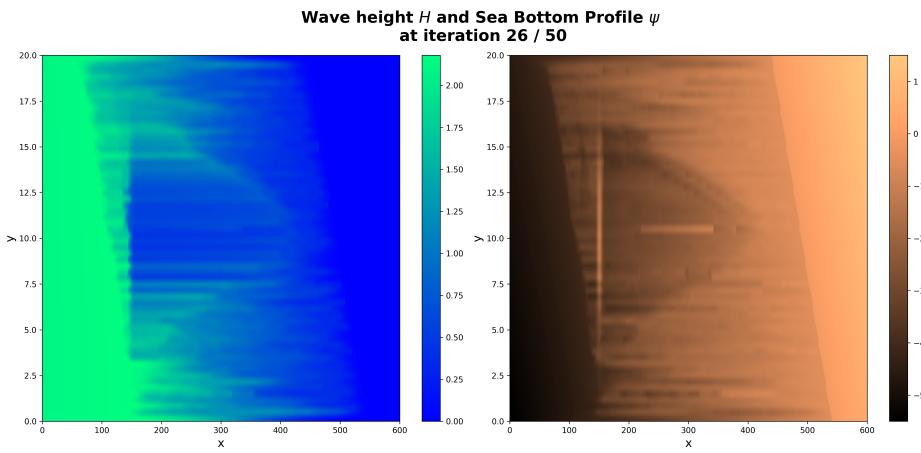


Figure 5.31 – Seabed halfway through the simulation

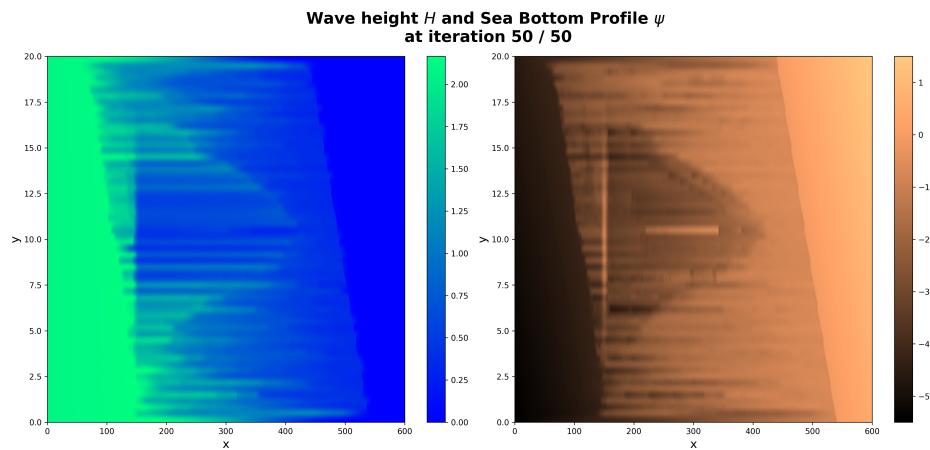


Figure 5.32 – Final seabed at the end of the simulation

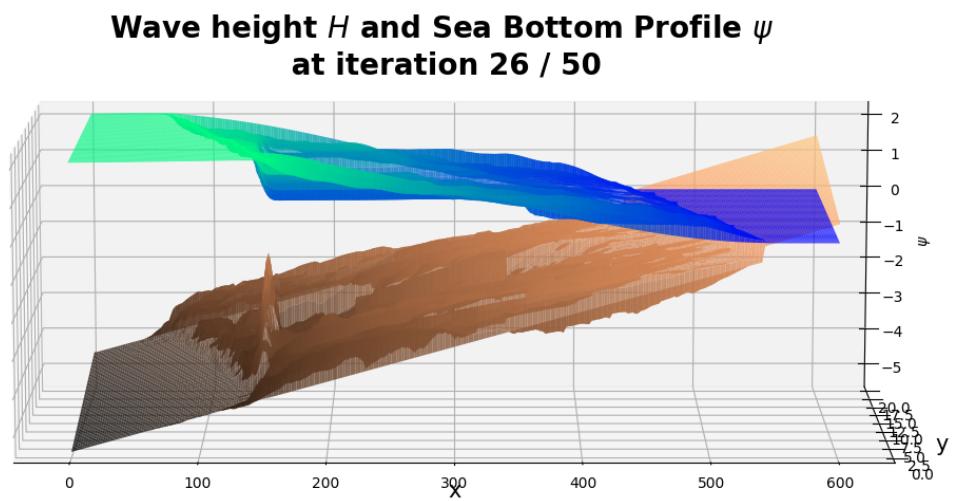


Figure 5.33 – Seabed halfway through the simulation

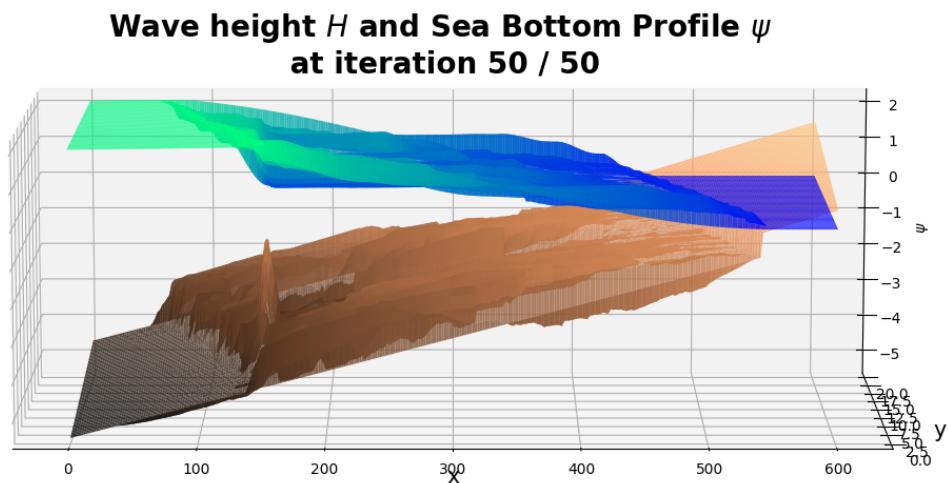


Figure 5.34 – Final seabed at the end of the simulation

A thorough analysis of the results of OptiMorph can be found in chapter 4 of the thesis for the 2D configuration with a geotube.

CONCLUSION ET PERSPECTIVES

1 Conclusion

Les travaux de cette thèse se sont principalement concentrés dans le développement théorique et numérique du modèle morphodynamique OptiMorph. Ce modèle morphodynamique, fonctionnant par le principe de minimisation, rejoint la vision de Murray (2007) en tant qu'élément essentiel à la modélisation, à savoir, un modèle exploratoire reproduisant certains phénomènes naturels. Il a su faire ses preuves sur les aspects numériques, mais également physiques. En effet, celui-ci produit des résultats physiques très réalistes de manière très rapide, sans nucléation ni hyperparamètres. Ce modèle est particulièrement performant sur les phénomènes de créations de barres sédimentaires, il a pu être validé sur les cas de benchmark de SANDS, COPTER et LIP 1C. Grâce à l'approche mathématique de la dérivée par rapport à la forme au sens d'Hadamard, nous avons pu rendre notre modèle générique et donc, couplable avec n'importe quel modèle de vagues. Ceci rend notre modèle polyvalent et opérationnel, pouvant ainsi être déployé facilement pour résoudre des problèmes d'ingénierie du littoral.

Dans l'état de l'art 0, nous avons retracé les travaux initiés en 2004 par B. MOHAMMADI, F. BOUCHETTE et P. AZERAD. La description de ces travaux commence par des problématiques d'optimisation de formes ou de position de structures de défense côtière. La finalité de ces travaux concerne la modélisation morphodynamique par optimisation. Ces derniers travaux ont présenté quelques limites. C'est à la suite de cet état de l'art que les travaux de cette thèse ont pu commencer.

Ensuite, le chapitre 1 s'est concentré sur l'approche historique décrivant la morphodynamique côtière par la théorie de l'optimisation. Plus spécifiquement, le modèle fonctionne sur l'hypothèse qu'un profil de plage sableuse évolue afin de minimiser une fonctionnelle liée aux vagues, dont le choix dépend de ce qui est considéré comme la force motrice derrière les processus morphodynamiques côtiers considérés. Des résultats numériques ont été présentés attestant que notre modèle est stable: celui-ci est bien consistant en temps et en espace. Ensuite, les résultats comparant OptiMorph, XBeach et une des données

expérimentales se sont révélés très bons. De plus, les temps de calcul de notre modèle sont très faibles en comparaison aux modèles classiques. Le modèle a ensuite été étendu en Multi-1D. Des applications effectuées au large de Montpellier ont donné des résultats semblables à des expériences en bassin (COPTER 2D ([Bouchette 2017](#))).

Puis, le chapitre [2](#) commence par les équations fondatrices des mouvements fluides pour dérouler à la théorie linéaire ainsi qu’aux deux grandes familles de modèles hydrodynamiques: les modèles à phase résolue et les modèles spectraux. Nos travaux nécessitant principalement un modèle hydrodynamique produisant une hauteur significative, nous avons choisi les modèles SWAN XBeach, un modèle hydrodynamique que nous avons développé ainsi que le modèle Shallow-Water. En nous basant sur des résultats expérimentaux (LIP 1C ([Roelvink et al. 1995](#))), nous avons montré que les résultats de ces modèles sont relativement bons et que nous pouvons les utiliser dans notre modèle morphodynamique.

Ensuite, le chapitre [3](#) exhibe une nouvelle approche mathématique. En utilisant la dérivée à la manière d’Hadamard, nous avons réussi à rendre notre modèle générique. Grâce à ces avancées, celui-ci peut être couplé à n’importe quel modèle hydrodynamique. Nous avons utilisé les modèles hydrodynamiques de SWAN, XBeach et Shallow-Water dans notre modèle, et nous avons comparé les résultats morphodynamiques au benchmark hydro-morphodynamique LIP11D ainsi qu’à des simulations en pleine mer. Les résultats sont très encourageants dans la mesure où notre modèle est capable de créer une barre sédimentaire au point de déferlement de la vague: ce qui est très proche des conditions réelles.

Enfin, dans le chapitre [4](#), nous étendons l’approche du chapitre précédent à la dimension 2D. Nous développons le formalisme en 2D de notre modèle morphodynamique et nous analysons les résultats qu’il produit sur la configuration expérimentale Copter 2D. Cette configuration peut être utilisée dans l’ingénierie côtière. L’utilisation des modèles REF/DIF et Shoaling en multi-1D mettent en évidence la forte sensibilité au modèle de vague utilisé. Pour obtenir des résultats morphodynamiques cohérents, il est très important de bien choisir le modèle hydrodynamique associé, il doit être choisi en fonction du site d’étude (type de plage, type d’expérience, type de vagues, etc.).

2 Perspectives

Cette thèse s’étant consacrée principalement au développement du modèle OptiMorph, a réussi à restreindre les limitations du modèle évoquées par [Cook \(2021\)](#) dans sa conclusion. Cependant, il reste tout de même un grand nombre de travaux pouvant être effectués afin d’améliorer le modèle.

Un premier point concerne les développements théoriques du modèle. Celui-ci pourra

toujours être amélioré en incorporant toujours plus de physique. Suite aux échecs décrits dans l'annexe B.2, la notion de courant n'a pas été implémentée dans le modèle. Cependant, l'approche du transport décrite dans la section 5.2 du chapitre 3 et en annexe B.3, est un point qui pourrait s'avérer particulièrement intéressant. En effet, en paramétrant une vitesse comme un problème d'optimisation, il doit être possible de gérer un transport vers la côte ou vers le large.

Un second point concerne la validation du modèle. Bien que des travaux aient été effectués sur des cas expérimentaux dans les chapitres 1, 3 et 4. Il est toujours possible de valider le modèle sur des cas différents comme des cas d'accrétion ou bien d'autres cas plus connus comme celui présenté dans l'expérience DynaRev ([Blenkinsopp et al. 2021](#); [Schimmels et al. 2020](#); [Martins et al. 2020](#)), mettant une nouvelle fois en évidence le phénomène de création de barres sédimentaires : ce qui est très bien reproduit par notre modèle. Par ailleurs, le modèle a été validé sur des cas expérimentaux de canal / bassin, il pourrait être intéressant de le valider sur des conditions réelles en pleine mer. Cependant, ces données sont souvent très rares.

Un troisième point, serait d'intégrer notre modèle OptiMorph comme un module d'extension aux modèles de vagues actuels pouvant ainsi produire un résultat morphodynamique. Il pourrait également être ajouté aux modèles morphodynamiques actuels : en résolvant, à la fois, la morphodynamique classique et celle d'OptiMorph, ceci pourrait permettre à ces modèles de pallier certaines limites, notamment sur l'aspect de la création de barres sédimentaires.

Un dernier point pourrait être de coupler notre modèle avec un plus grand nombre de modèles de vagues. Ceci pourrait permettre à l'utilisateur de choisir le modèle qu'il souhaite, sans avoir à se plonger dans le code.

Enfin, une perspective numérique pourrait être d'utiliser l'approche Hadamard décrite dans le chapitre 3 afin de l'utiliser sur des problèmes d'optimisation de forme. Cette approche ne se limite pas seulement à la modélisation morphodynamique des plages, elle pourrait être utilisée dans de nombreux autres domaines : l'ingénierie côtière, l'aéronautique, l'acoustique, la thermodynamique, ...

APPENDIX

A \mathfrak{A} Genetic Algorithm to Solve the Optimization Problem (I.1)

In this section, we describe in more detail how the dual genetic algorithm presented in the introduction works.

A.1 Population Creation

First, a population $\psi_{i=0,\dots,N_{pop}}$ of N_{pop} individuals is created (Figure I.2.1)). Initially, all these individuals are identical to the mother individual ψ_M . For each of these individuals, a random number of "bumps" (represented by figure A.1) is added to ψ_M . The parameters of these "bumps" (number, amplitude A , position, width L) are determined randomly: this ensures diversity in the initial population, as illustrated in the figure I.2.1).

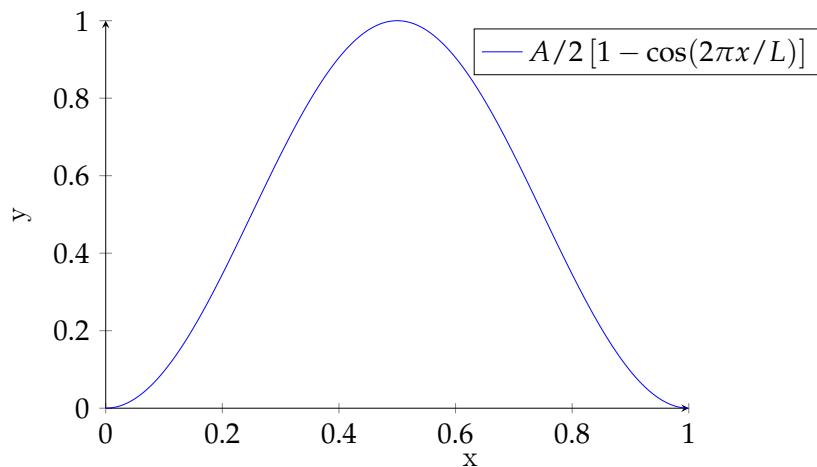


Figure A.1 – Illustration of a sinusoidal half-wrap of amplitude $A = 1$ and length $L = 1$.

The choice of representing our population with sinusoidal evolution comes from FOURIER

who assures that it's possible to represent any function with an infinite number of sinusoids.

A.2 Selection

For each generation (and therefore iteration), 2 individuals ψ_q , ψ_p (with $q, p \in [0, \dots, N_{pop}]$) are randomly selected. These two individuals from the population N_{pop} are then compared in the form of a "duel". In the case of minimization, the individual with the highest \mathcal{J} cost function is mutated. The other individual is retained. Here, the cost function $\mathcal{J} = \frac{1}{16}\rho_w g \int_{\Omega} H^2 d\Omega$ is the wave energy integrated over the whole domain. This selection is represented by Figure I.2.2) and is algorithmically translated by the following pseudo-code 2.

Algorithm 2 Selection and mutation

Input: N_{pop} the number of individuals in the population, $\psi_{i=0, \dots, N_{pop}}$ 0 the population, `randint` the function returning a random integer, `calc_J` the calculation of the \mathcal{J} cost function, `mutation` the function performing a mutation.

Output: The new ψ_i population with a mutation.

```
1:  $p \leftarrow \text{randint}(0, N_{pop})$      $\triangleright$  Random selection of an individual from the population.  
2:  $q \leftarrow \text{randint}(0, N_{pop})$   
3:  $\mathcal{J}_p \leftarrow \text{calc\_J}(\psi_p)$                        $\triangleright$  Calculation of cost function  
4:  $\mathcal{J}_q \leftarrow \text{calc\_J}(\psi_q)$   
5: if  $\mathcal{J}_p > \mathcal{J}_q$  then  
6:    $\mathcal{J}_p \leftarrow \text{mutation}(\mathcal{J}_q)$                    $\triangleright$  Cloning + Mutation  
7: else  
8:    $\mathcal{J}_q \leftarrow \text{mutation}(\mathcal{J}_p)$   
9: end if
```

A.3 Mutation

As previously indicated in the 2 algorithm, the individual undergoing mutation becomes an evolution of the individual with the lowest \mathcal{J} cost function. Mutation acts in a very similar way to population creation. One or more of the individual's bumps will be selected. They will then be mutated by changing their parameters (number, amplitude A , position, width L). They can thus be made to move slightly, have a larger (or smaller) amplitude and a slightly different length. Unlike the creation of the population, the parameters are not entirely regenerated; the old parameters evolve by a small amount ϵ in order to ensure the convergence of the population. This small ϵ shift can be seen on the I.2.3) where there is not much variability between the two individuals (the old and the

mutated). Moreover, the mutated individual has a slightly lower-cost function than the original individual.

A.4 Final Population

By cleverly choosing a convergence criterion, we decide to stop mutations within the population. If the criterion is well chosen, we should obtain a population that has entirely converged at a single point, as shown in the figure I.2.4). In our case, the algorithm stops when \mathcal{J} hardly evolves over a certain number of iterations. This gives us the convergence curve figure A.2.

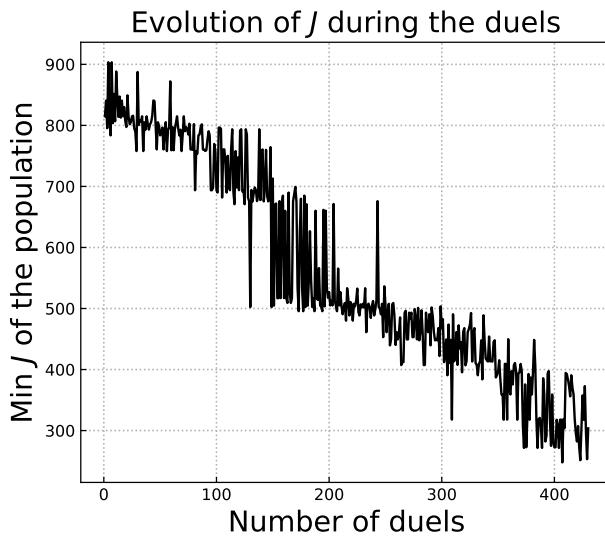


Figure A.2 – Evolution of \mathcal{J} during duels.

Note that the population tends towards a min value. Then, to select the best candidate, we simply select the candidate with the lowest cost function.

B A few Attempts to Improve the Model

Throughout the thesis, we have considered a single \mathcal{J} functional, the wave energy integral. This choice was made deliberately, as we feel that this functional is the most realistic and produces the most realistic results. However, a multitude of different functionals and approaches have been tested in order to improve our model. In this section, we present some of the non-concurrent tests that have been carried out to improve our model. We have essentially focused on two cases: a case with linear bathymetry and an experimental case.

B.1 Reference Cases

In order to test our new functional as well as possible, we will carry out numerous simulations based essentially on the two following test cases. These two cases were presented in Chapter 1, so we know where their limits lie and what results we can expect.

B.1.1 Case 1: Simulation of a One-Week Storm on a Linear Beach

This first benchmark simulation is presented in Chapter 3 of [Cook \(2021\)](#). This simulation is described as highly morphogenic in that it simulates a storm over a few days with the following parameters:

Physics	Simulation parameters			Hydrodynamic		Morphodynamic		Domain		
Parameters	Δx	Δt	T_f	H_{max}	T_0	Y	M_{slope}	L	h_0	slope α
Values	1 m	400 s	1 week	2 m	2 s	4.25e-5 m.s.kg ⁻¹	20%	600 m	7 m	11%

Table B.1 – Parameter of the storm simulation.

which are, the spatial step Δx , the time step Δt , the duration of the simulation T_f , the maximum water height in forcing H_{max} , the wave period T_0 , the sand abrasion Y , the maximum sand slope M_{slope} , the length of the simulation domain L , the closing depth h_0 . This is represented by the following forcing and domain figure B.1.

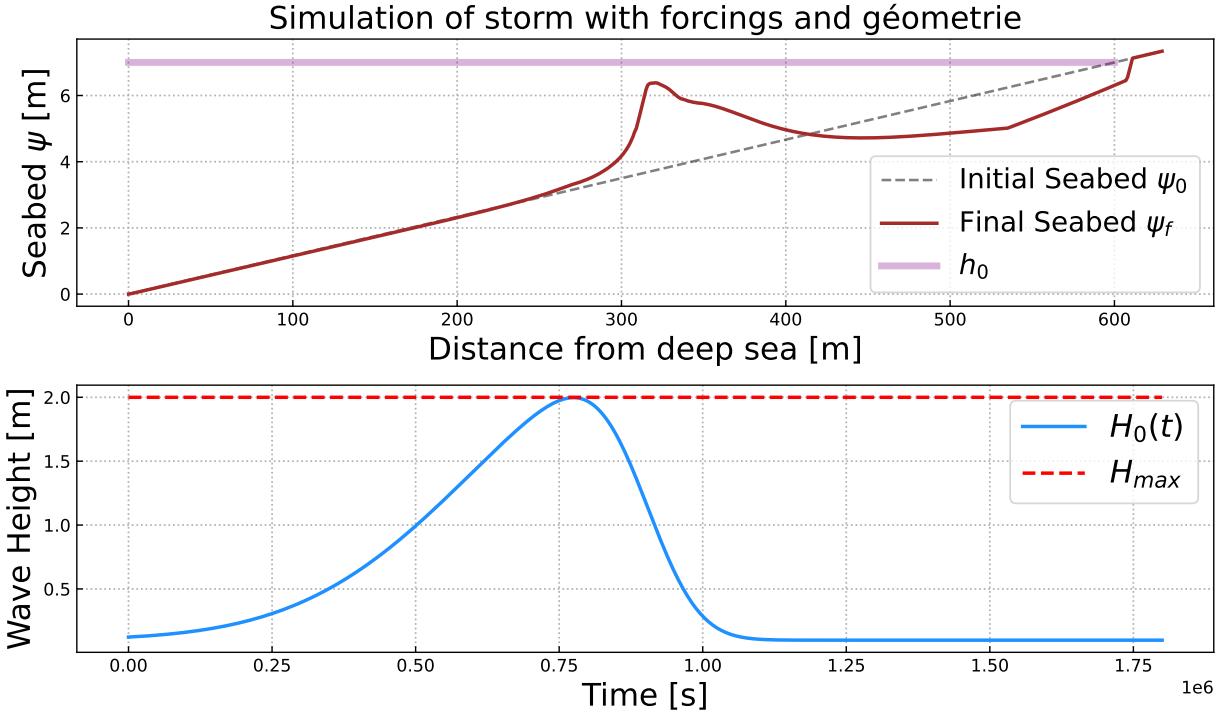


Figure B.1 – Forcing for the one-week storm simulation.

The result of this simulation shows the formation of a realistic sand bar. We use this simulation to observe how the model behaves when the physics are changed.

B.1.2 Case 2: Simulation of a Flume Experiment: COPTER

The second reference simulation is present in the article by [Cook \(2021\)](#). This reference case, called COPTER compares simulations with real data from basin tests. Hydro-morphodynamic simulations are carried out using our OptiMorph model and the well-known XBeach model. The results of these simulations are shown in the figure **B.2**.

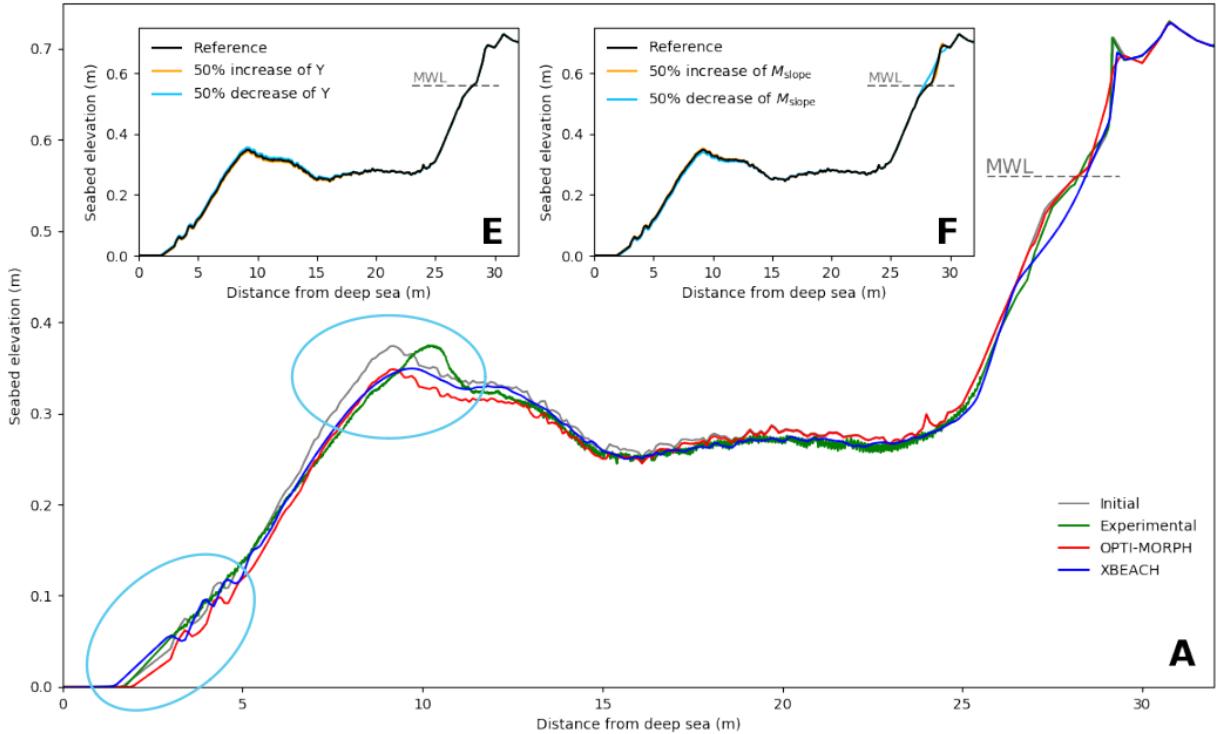


Figure B.2 – Weaknesses (circled in turquoise) of the Copter simulation at the morphodynamic level.

Analysis of these simulations revealed a number of major areas for improvement. These points have been circled in turquoise on the Figure **B.2**. 1) The movement of sand upstream of the simulation. The sandy movements at this point are too great with the OptiMorph model. 2) The sedimentary bar is supposed to be moving to the coast. However, it is sagging.

For the first point, it is possible that the experimental surveys present too many uncertainties at this level. Indeed, it is very difficult to perceive precisely the morphodynamic displacements at the level of the beater. The second point leads us to believe that some-

thing is missing in our model, such as a current that would allow us to move this bar that we have. The purpose of our functional approach will be to try to physically represent this current or other quantities.

B.2 Improvement by Functional Approach

The first idea that comes to mind improving the model in a rather naive way would be to change what governs the morphodynamics, namely the cost function \mathcal{J} . A very large number of functional \mathcal{J} was tested in order to take into account more physics. In the work of [Cook et al. \(2021c\)](#), a number of functional has been tested, as shown in the table [B.2](#).

Keyword	Definition	Commentary
CF0	$d = -\nabla_\psi \mathcal{E}_S \chi_{\Omega_S}$	Recommended
CF1	$d = -\frac{x_B}{x_S} \nabla_\psi \mathcal{E}_S \chi_{\Omega_S}$	
CF2	$d = -\frac{x_B}{x_S} \nabla_\psi \mathcal{E}_S \chi_{\Omega_S}$	
CF3	$d = -x_B \nabla_\psi \mathcal{E}_S \chi_{\Omega_S}$	
CF4	$d = -\frac{x_B}{x_S} \int_{\Omega_S} \nabla_\psi \mathcal{E}_S \chi_{\Omega_S}$	
CF5	$d = (1 - \Lambda) CF2 + \Lambda CF4$	where Λ is the excitation of the seabed
CF6	$d = (1 - \Lambda) CF3 + \Lambda CF4$	where Λ is the excitation of the seabed

Table B.2 – Old cost functions \mathcal{J} .

The results of these tests showed that the most physical and relevant functional was the wave energy functional $\mathcal{J} = \mathcal{E}_H$. This one showed the most relevant results present in ([Cook 2021](#)).

In a similar way, we tried out a large number of functional. Some of the functional we tested are shown in the table [B.3](#).

Keyword	Définition	Commentaire
CF8	$d = -\nabla_\psi (\mathcal{E}_S - \varepsilon \rho U_{orb}^2) \chi_{\Omega_S}$	Kinetic energy removal
CF9	$d = -\nabla_\psi (\rho U_{orb}^2) \chi_{\Omega_S}$	Kinetic energy
CF10	$d = -\nabla_\psi (\varepsilon C_g H^2) \chi_{\Omega_S}$	
CF11	$d = -\nabla_\psi (C_g H) \chi_{\Omega_S}$	
CF12	$d = -\nabla_\psi (\varepsilon (C_g H)^2) \chi_{\Omega_S}$	
CF13	$d = -\nabla_\psi (S_{xx}) \chi_{\Omega_S}$	Radiation stress
CF14	$d = -\nabla_\psi (\nabla S_{xx}) \chi_{\Omega_S}$	Gradient of adiation stress
CF15	$J = \frac{1}{8} \rho_w g \int_{\Omega_S} H^2 dx + \rho_s g \int_{\Omega_S} (\psi(t) - \psi_0(\tau - t))^2 dx$	Sand displacement memory

Table B.3 – Nouvelles fonctions de coût \mathcal{J} .

these functions have been found by relying on basic physics: calculation of forces, work, kinetic energy, ... but also by relying on the balance of moments ([Sous et al. 2020](#)):

$$\frac{\partial}{\partial x} \left(\rho U^2 \right) = -g\rho \frac{\partial \bar{\eta}}{\partial x} - \frac{1}{(\bar{\eta} + h)} \frac{\partial S_{xx}}{\partial x} - \frac{1}{(\bar{\eta} + h)} \bar{\tau}_b \quad [Pa \cdot m^{-1}], \quad (\text{B.1})$$

where U is the depth-averaged velocity, g the gravitational acceleration, ρ the water density, $\bar{\eta}$ the wave setup, h the still water depth, S_{xx} the radiation stress, and $\bar{\tau}_b$ the bed shear stress.

B.2.1 Functional with Kinetic Energy Dissipation (CF8)

The first functional aims at introducing a term taking into account the kinetics of a wave. Indeed, if we take into account the kinetics of a wave, it could be that it could artificially simulate a "current" which would allow us to obtain a displacement of the bar.

$$\mathcal{J} = \frac{1}{16} \int_{\Omega_s} (\rho g H^2 - \varepsilon \rho U_{orb}^2) dx \quad (\text{B.2})$$

with ε in m^{-1} that we will fix for the moment arbitrarily. In order to implement this functional, it is necessary to differentiate it according to ψ (calculation of $\nabla_\psi \mathcal{J}$). The term \mathcal{J}_H has already been differentiated [2.1](#), it only remains to differentiate the term $\int_{\Omega_s} \varepsilon \rho U_{orb}^2 dx$. This is easily done via the following expression for the orbital velocity:

$$U_{orb} : \Omega \times [0, h_0] \longrightarrow \mathbb{R}^+ \\ (x, \psi) \longmapsto \frac{\cosh(k(x)(h(x) - (h_0 - \psi)))}{\cosh(k(x)h(x))}. \quad (\text{B.3})$$

We start again from the calculation of the gradient which allows us to obtain the new expression:

$$\nabla_\psi \mathcal{J} = \frac{1}{4} \rho (g H \partial_\psi H + \varepsilon \partial_\psi U_{orb}) \quad (\text{B.4})$$

with the only term here that we don't know: $\partial_\psi U_{orb}$. With the equation [\(B.3\)](#), we obtain :

$$\boxed{\partial_\psi U_{orb} = (\partial_\psi u v - u \partial_\psi v) v^{-2}}$$

with:

$$\begin{aligned}
 u &= \cosh(k(h - (h_0 - \psi))) \\
 v &= \cosh(kh) \\
 \partial_\psi u &= \sinh(k(h - (h_0 - \psi)))(h + \psi - h_0)k_\psi \\
 \partial_\psi v &= \sinh(kh)(k_\psi h - k)
 \end{aligned}$$

which is easy to calculate because we know the values of h_ψ and k_ψ which were specified in the part 2.

We now try to run simulations on Case 1 ([B.1.1](#)) and Case 2 ([B.1.2](#)) with different values of ϵ from 0.001 to 0.009. The results are shown in the figures [B.3](#) and [B.4](#).

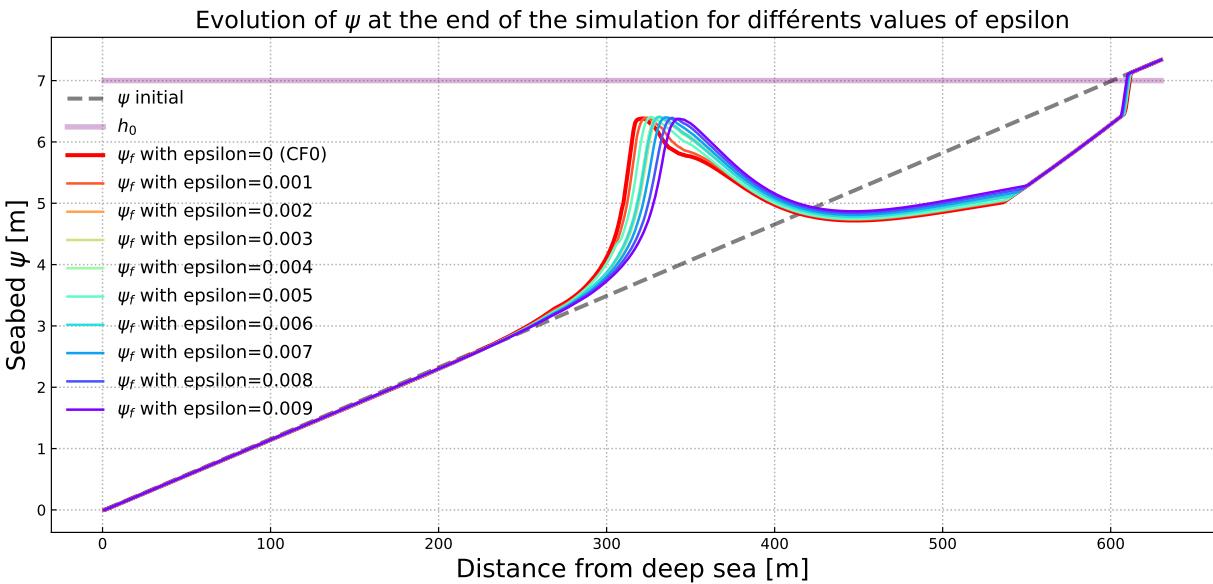


Figure B.3 – Results for different ϵ with the dissipation functional for simulation [B.1.1](#).

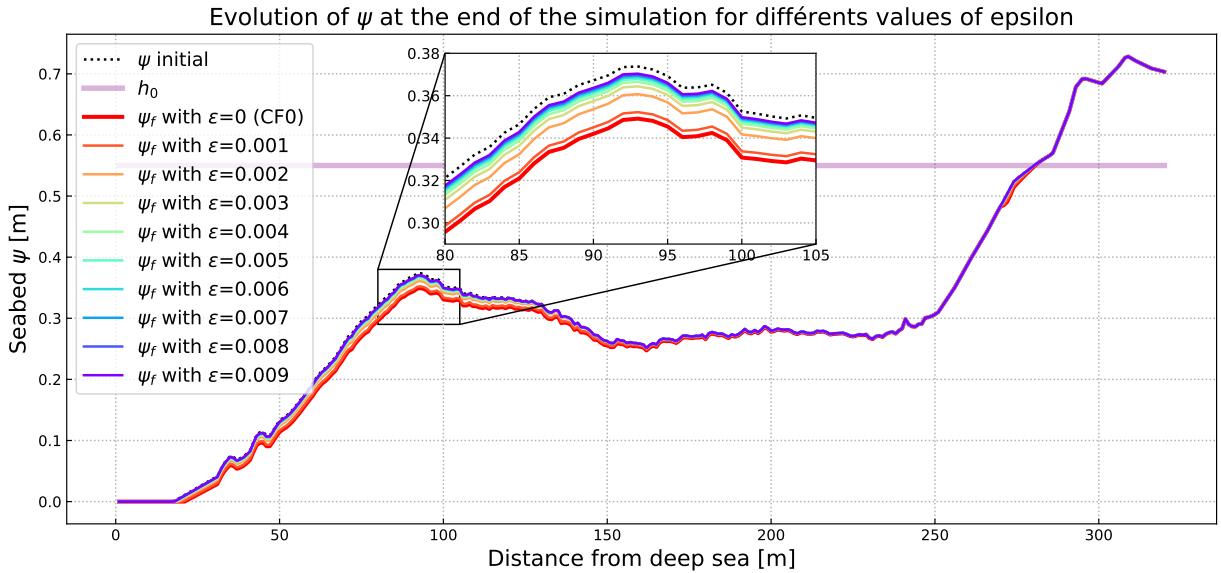


Figure B.4 – Results for different ϵ with the dissipation functional for simulation [B.1.2](#).

In these cases, the CF0 functional is represented in red. Results from figure [B.3](#) suggest that it is possible to move the bar with this functional. However, looking at the figure [B.4](#) we notice that there is no displacement of the sedimentary bar. In fact, the effect of attenuation is merely to attenuate sediment mobility. The conclusion on this functional from a numerical point of view is that it simply attenuates the descent in the gradient descent method. The results with ϵ of the first case are very similar to other results of ψ at 90-100% of the simulation time.

To understand what the model does with this functional, it may be interesting to look at the distribution of each term on the domain. That is to say, where does the term $\mathcal{J}_{cinetique}$ acts and where does the term $\mathcal{J}_{\mathcal{H}}$ act. We display for the first case on figure [B.5](#), the rapports $\frac{J_c}{J_{tot}}$ and $\frac{J_{\mathcal{H}}}{J_{tot}}$ at different moments of the simulation: at the beginning, at the peak and at the end.

APPENDIX

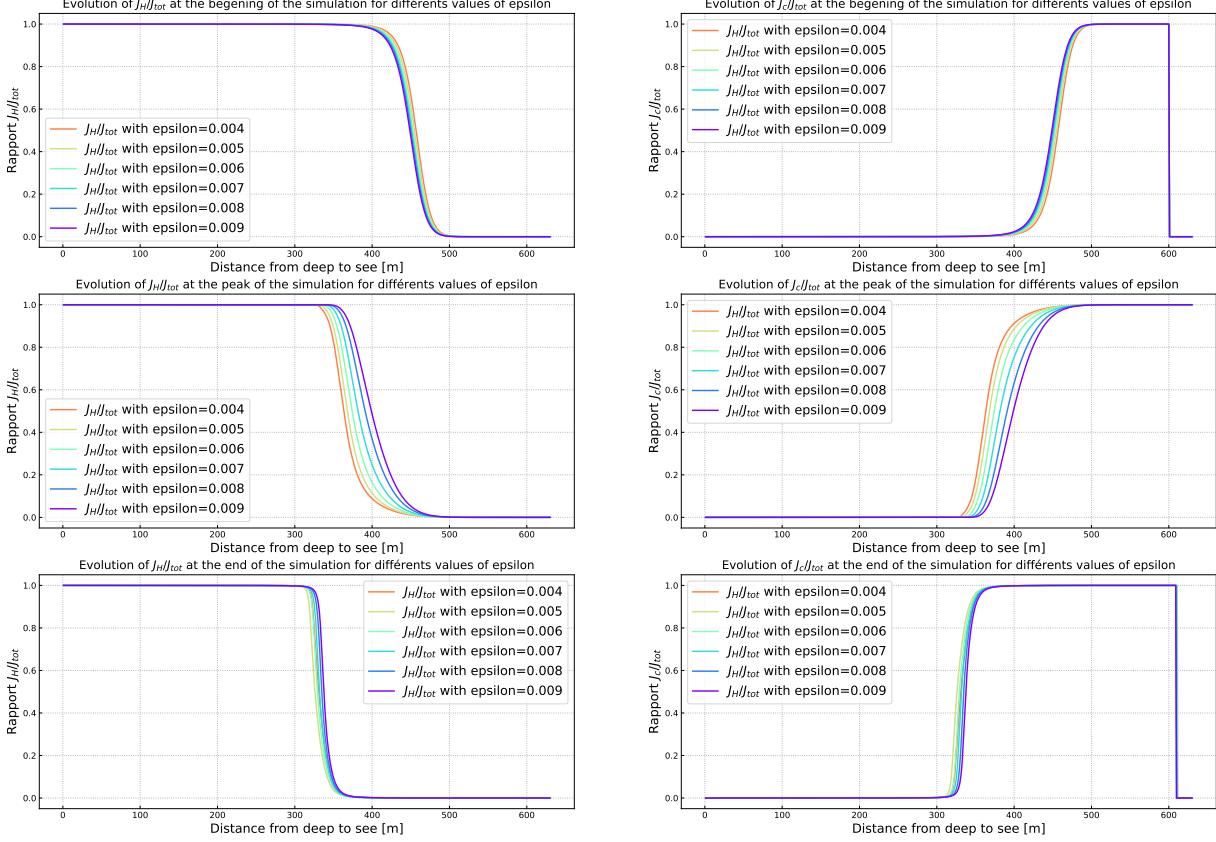


Figure B.5 – Rapports $\frac{J_c}{J_{tot}}$ and $\frac{J_H}{J_{tot}}$ at different times of the simulation: at the beginning, at the peak and at the end: according to different values of ϵ .

Not surprisingly, we notice that in all cases, the \mathcal{J}_H cost function is predominant in far from the coast and then the \mathcal{J}_c cost function takes over for the break-up. This can be explained because there are changes in velocities when the wave breaks and therefore the gradient is more likely to evolve. Although this work is interesting in understanding the physics behind the model despite this functional does not help us to address the weaknesses of our model. We therefore reject this functional in order to move towards other functions.

B.2.2 Functional in Terms of Representing Work

Other functions were tested in the same way at the B.2.1 part. The idea here is to consider a functional that would represent the notion of work. This approach aims to "artificially" adds the notion of current thank to the group velocity. The functionals tested are the following:

$$\mathcal{J}_{CF10} = \varepsilon C_g H^2 \quad (B.5a)$$

$$\mathcal{J}_{CF11} = C_g H \quad (B.5b)$$

$$\mathcal{J}_{CF12} = \varepsilon (C_g H)^2. \quad (B.5c)$$

The differentiation of these functional has been done in a similar way to the previous part [B.2.1](#). The results coming from these functionals are very similar to those produced by the \mathcal{J}_H functional. These functions don't add significant value to the model, so we've decided not to keep them.

B.2.3 Functional with Radiation Stress S_{xx}

In the momentum balance on a wave (equation [\(B.1\)](#)), there is a radiation stress term. This one represents an excess of flow is present as a result of the orbital movement of a wave. This can be observed on the figure [B.6](#). This excess dissipates mainly on the bottom friction.

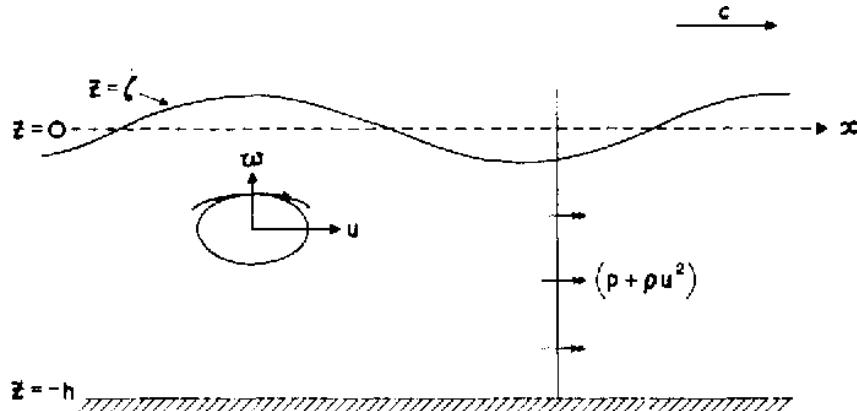


Figure B.6 – Diagram of momentum balance on a wave.

[Longuet-Higgins et al. \(1962\)](#) have sought to quantify this excess flow name the radiation stress S_{xx} . To quantify this variable, the following expression (equation [\(B.6\)](#)) was established.

$$S_{xx} = \overline{\int_{-h}^{\eta} (p + \rho \tilde{u}^2) dz} - \int_{-h}^0 p_0 dz \quad (B.6)$$

The first term of this expression is the total flux of momentum of a wave averaged. It is then subtracted from it the average flow in the absence of a wave. This quantity S_{xx} is thus to be seen as the difference between the time-averaged flux of momentum and

the average flux in absence of a wave. The work of Longuet-Higgins et al. (1962) was to simplify the expression of S_{xx} by a simpler expression namely:

$$S_{xx} = \left(2\frac{C_g}{C} - \frac{1}{2} \right) E = \frac{1}{8}\rho g \left(\frac{2kh}{\sinh 2kh} + \frac{1}{2} \right) H^2 \quad [J.m^{-1}] \quad (\text{B.7})$$

with C_g the group velocity [m], ρ the water density [$kg.m^{-3}$], g the gravitational constant [$m.s^{-2}$], H the significant height and E the wave energy. The new approach would be to postulate that *the system will try to minimize its energy in the sense of momentum: minimize the slopes and currents of it*. The idea transmitted through this is to suppose that the system tries to *minimize the mechanisms by which this energy is transmitted to it*. This would mean in our case to reduce the spatial gradient of the radiation stress. To be clearer, we would try to minimize the following functional:

$$\mathcal{J} = \varepsilon \nabla_x (S_{xx}) \quad [J.m^{-2}] \quad (\text{B.8})$$

with ε in [m] chosen arbitrarily. The differentiation of these functions will be done in a similar way to the previous part B.2.1. Comparing the results obtained with this functional with the others, we notice strong oscillations on the morphodynamics. This can be explained by the fact that the gradient is calculated numerically: this can induce many numerical biases. Moreover, calculating the gradient will necessarily lead to oscillations since the values of S_{xx} (equation (B.8)) are terms in \sinh which oscillate a lot. We can try for example to calculate a functional based only the radiation stress of the following form (equation (B.9)):

$$\mathcal{J} = S_{xx} \quad [J.m^{-2}]. \quad (\text{B.9})$$

Like the previous part, this functional gives results very close to those obtained by the functional \mathcal{J}_H . This is because the expression equation (B.8) shows that this is simply a slightly more complex form of energy function. It is therefore normal to obtain results very close to those obtained by the \mathcal{J}_H . We also reject these functional. All previous work on functional amounts to performing calculations from the same linear theory of physics with variables often very close to the calculation of \mathcal{J}_H .

B.2.4 Functional with Memory Term

A new approach, quite different from the previous ones, would be to incorporate "memory" into the functional. This approach was inspired by (Mohammadi et al. 2014) where we add a constraint on the movement of the sand requiring a minimum of bathymetric changes over the time interval $[t - \tau, t]$ with τ chosen so that $\tau \gg T_0$ and T_0 is the wave

period. This gives us the following functional:

$$\mathcal{J} = \frac{1}{8} \rho_w g \int_{\Omega_S} H^2 dx + \rho_s g \int_{\Omega_S} (\psi(t) - \psi_0(\tau - t))^2 dx, \quad (\text{B.10})$$

with ρ_s the density of sand [$kg.m^{-3}$], ρ_w water density [$kg.m^{-3}$], H significant water height [m], g the gravitational constant [$m.s^{-2}$], ψ bathymetry [m] and ψ_0 initial bathymetry [m].

By differentiating the equation (B.10), we obtain:

$$\nabla_\psi \mathcal{J} = \nabla_\psi J_H + 2\rho_s g (\psi^n - \psi^{n-1}) \quad (\text{B.11})$$

and so we can expand using the descent equation (5.16) to arrive at:

$$\psi^{n+1} = \psi^n - dt \Lambda \Upsilon \frac{\nabla_\psi \mathcal{J}_H}{1 - dt \Lambda \Upsilon \rho_s g}, \quad (\text{B.12})$$

in an unconstrained configuration. This equation is very similar to the equation (B.10). The results obtained with this functional are still very similar to those obtained with \mathcal{J}_H . They are almost identical when we use $\tau = dt$ presented in B.1.1. The idea of acting on the numerical scheme can be an interesting approach. We could add more physics, add this notion of transport by adding a transport term in the numerical descent scheme as we will see in the next part.

B.3 Adding Transport in the Descent Scheme

By discretizing the descent equation (5.16), we obtain the following basic descent equation without constraints:

$$\psi_i^{n+1} = \psi_i^n - \Upsilon \Lambda \nabla_\psi \mathcal{J}. \quad (\text{B.13})$$

We can easily add a term representing a horizontal transport according to a speed V [$m.s^{-1}$] which would transform the equation (B.13) into a new equation:

$$\psi^{n+1} = \psi^n + \Upsilon \Lambda \nabla_\psi \mathcal{J}(\psi^n) - \rho V^n \nabla_x \psi^n. \quad (\text{B.14})$$

This equation can be easily implemented by calculating $\nabla_x \psi^n$ by finite differences. For our tests, we take a constant speed $V = 0.001 m.s^{-1}$ and the orbital velocity $V = U_{orb}$. For the case 1 (B.1.1), we obtain the following final bathymetry results ψ_f on figure B.7.

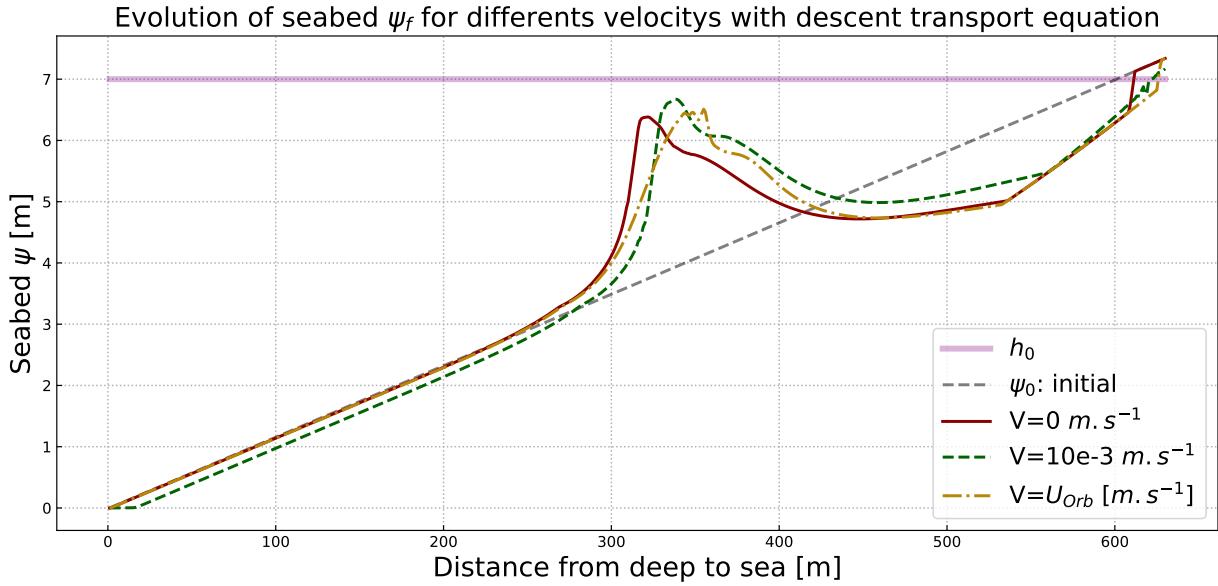


Figure B.7 – Evolution of ψ_f bathymetry for different speeds with the descent transport equation.

The case where $V = 0.001 \text{ m.s}^{-1}$ (green) shows us that transport works well. However, it makes no physical sense. This is not realistic because we should not have any sand displacement in deep-water. The case where $V = U_{orb}$ seems quite realistic. Moreover, it shows a displacement of the bar without velocity towards the side. This could possibly lead to better results on case 2 [B.1.2](#). Performing analogous simulations, we obtain the following results figure [B.8](#).

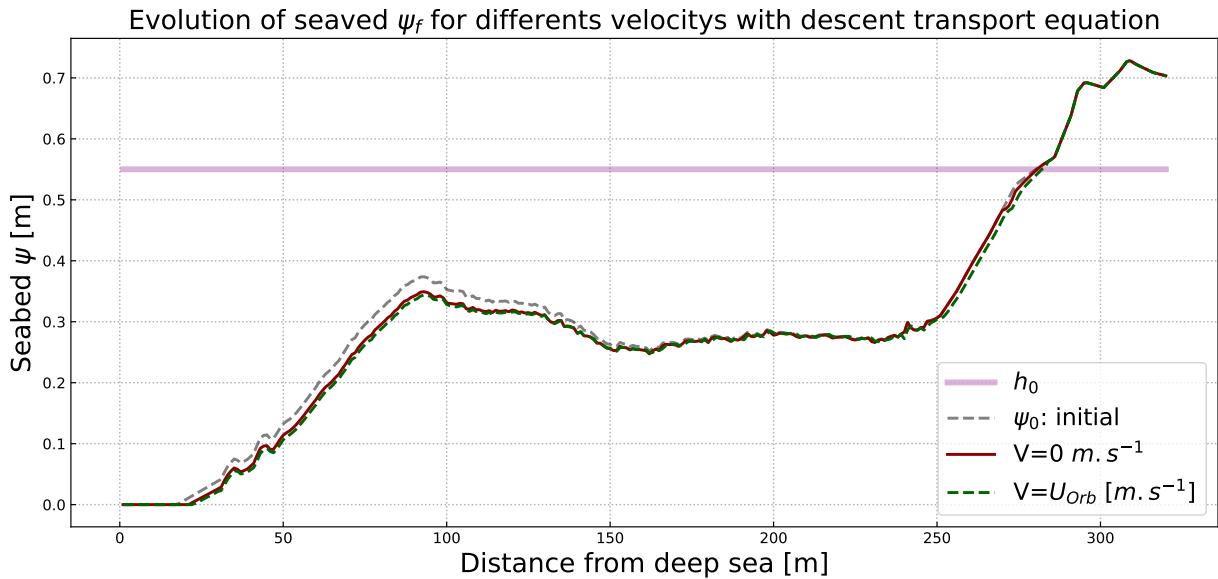


Figure B.8 – Evolution of the seabed ψ_f for different speeds with the descent transport equation.

In this case, we don't see the displacement of the sedimentary bar we'd hoped for. Although the results are not promising, this approach is still very interesting for defining a current in our model. Assuming we have a wave-to-wave resolution model, we could obtain the actual current u . This would be much more relevant than orbital velocity.

B.4 Conclusion

This part focused on a functional approach to account for a better physics by trying to solve the limitations stated by case 1 ([B.1.1](#)) and 2 ([B.1.2](#)). Many other functional have been tested, such as some with currants (from SWAN / XBeach models), bottom stress, Some were interesting but none of them was conclusive enough to lift the limits with this functional approach. However, as indicated in the chapter [3](#) in the section [5.2](#), the velocity remains to be defined cleanly in order to be robust on this transport.

C Configuration File of XBeach

params.txt

```
%%%%% XBeach parameter settings input file %%%
%%% case DISCOVER1 %%%
%%% Bed composition parameters %%%
rhos      = 2650
D90       = 0.0002

%%% Grid parameters %%%
depfile = bathy_psi.dep
posdwn   = -1
nx = 179
ny       = 0
alfa     = 0
vardx    = 1
xfile = x.grd
yfile = y.grd
xori     = 0
yori     = 0
thetamin = -180
thetamax = 180
```

APPENDIX

```

dtheta      = 360

%%% Model time %%%%%% %%%%%% %%%%%% %%%%%% %%%%%% %%%%%% %%%%%% %%%%%% %%%%%%
tstop       = 151
CFL         = 0.900000

%%% Morphology parameters %%%%%% %%%%%% %%%%%% %%%%%% %%%%%% %%%%%% %%%%%% %%%%%%
%ne_layer   = bathy_D1_b.dep

%%% Waves %%%%%% %%%%%% %%%%%% %%%%%% %%%%%% %%%%%% %%%%%% %%%%%% %%%%%% %%%%%%
instat     = 0
Hrms       = 0.42
Trep       = 8.0
dir0       = 270

%%% Output %%%%%% %%%%%% %%%%%% %%%%%% %%%%%% %%%%%% %%%%%% %%%%%% %%%%%% %%%%%%
tint       = 1
tstart     = 150

nglobalvar = 6
zs
zb
H
k
u
taubx

```

D Configuration File of SWAN

params.txt

```
PROJECT 'maupiti1D' 'HOE'  
  
$ Setting the configuration  
$ #####
```

```

SET LEVEL 0.0 NOR 90 INRHOG 1
MODE STATIONARY ONEDIMENSIONAL
COORDINATES CARTESIAN

$ Definition of the grid/ bottom condition
$#####
$      REGULAR [xpc] [ypc] [alpc] [xlenc] [ylenc] [mxc] [myc]           [mdc]   [flow]  [
CGRID REGULAR 0 0 0 180 0 179 0 CIRCLE 12 0.02 0.4 36

$ INPGRID BOTTOM REGULAR  [xpinp] [ypinp] [alpinp] [mxinlp] [myinlp] [dxinlp] [dyinlp]
INPGRID BOTTOM REGULAR 0 0 0 179 0 1 0

READINP BOTTOM -1 'psi.dat' 3 0 FREE

$ Setting physical quantities for the simulation
$#####
$ DIFFRACTION 1 0.2
OFF QUAD
GEN3
BREAKING BKD 1 0.73 7.59 -8.06 8.09
TRIAD

$ Definition of forcing conditions
$#####

BOUND SHAPESPEC JONSWAP 3.30 PEAK DSPr DEGREES

$                               PAR [hs]   [per]  [dir]  [dd]
BOUNDSPEC SIDE West CON PAR 0.42 8.0 0 20

$ Si on met 20, on réduit le nombre de basse fréquence qui est à Nan -999.

$ Definition of output features
$#####

CURVE 'profil' 0 0 179 179 0
TABLE 'profil' HEADER 'swan_output_HSIG.dat' HSIGN
TABLE 'profil' HEADER 'swan_output_T0.dat' TM01

```

APPENDIX

\$ Calculating waves
\$ #####

COMPUTE
STOP
\$
\$ Fin du calcul
\$ #####
STOP

LIST OF FIGURES

Figure I.1	1) Expérience LIP 1C avec H généré par XBeach. 2) Expérience DynaRev avec H mesuré par LIDAR. a) Fond marin et Hauteur d'eau moyennée au début de l'expérience (gris), Fond marin et Hauteur d'eau moyennée à la fin de l'expérience (marron). b) Énergies des vagues associées aux hauteurs d'eaux. L'énergie est calculée sur le rectangle gris.	21
Figure I.2	Workflow d'un algorithme génétique basé sur les duels pour résoudre le problème de minimisation (I.1). Plus d'explications dans l'annexe A.	24
Figure 0.1	Optimal shape minimizing a cost function \mathcal{J} for a given parameterization	29
Figure 0.2	Absolute value of ξ resulting from reflection (a) on rectangular structures perpendicular to the wall, (b) on optimized structures without feasibility constraints, (c) on structures with the same angle as the optimized structures but straight (NW incoming waves with $T=2$ s and $a=0.5$ m).	29
Figure 0.3	Possible configurations for the port of La Turballe	30
Figure 0.4	Weight function \mathcal{P} on Ω	31
Figure 0.5	Optimal pair of (α, β) for the configuration of the port of La Turballe.	31
Figure 0.6	Left: the initial bathymetry of region D for the cost function calculation; right: the modified bathymetry with the optimized protection structure.	33
Figure 0.7	The water height H in the whole domain: (left) for the initial configuration, (right) for the optimized configuration.	33
Figure 0.8	Configuration of the geotube simulation.	34
Figure 0.9	Final seafloor profiles produced by the Opti-Morph morphodynamic model with variable geotube positions on the cross-shore profile.	34
Figure 0.10	Agitation on the left and final bathymetry on the right (initial bathymetry: linear)	35
Figure 1.1	Diagram of a cross-shore profile in the case of an experimental flume.	41
Figure 1.2	Workflow of Optimorph 2021	41
Figure 1.3	Wave height $H(x)$ with the simple Shoaling model for a configuration with a linear bathymetry of slopes about 0.11, wave period $T_0 = 2$ s and offshore water height of $H_0 = 2$ m.	44
Figure 1.4	Wave height $H(x)$ with the improved Shoaling model for a configuration with a linear bathymetry of slopes about 0.11, wave period $T_0 = 2$ s and offshore water height of $H_0 = 2$ m.	45
Figure 1.5	Illustration of gradient descent with $\psi \leq \alpha$. The optimum does not necessarily correspond to the critical point $\nabla_\psi \mathcal{J} = 0$	46
Figure 1.6	Slope constraint (5.23) from (Cook 2021)	51

LIST OF FIGURES

Figure 1.7 Sand conservation (5.24) from (Cook 2021)	51
Figure 1.8 Comparison of mean wave height over a storm simulation. The green points correspond to the mean wave height provided by the gauges of the flume experiment. The mean wave height determined by OptiMorph (red) and XBeach (blue) also appears. The non-zero wave height beyond the shoreline as presented by XBeach is due to wave set-up, which OptiMorph does not handle.	53
Figure 1.9 A. Results of the numerical simulation calculated over the initial seabed (gray) using the XBeach morphodynamic module (blue) and the OptiMorph model (red). These are compared with the experimental data acquired during the COPTER project (green). The mean water level is denoted MWL and is set at 0.56 m. B. Zoomed in view of the sandbar, located between 6 m and 16 m. C. Zoomed in view of the plateau, located between 16 m and 24 m. D. Zoomed in view at the shoreline, located between 24 m and 32 m. E. Robustness analysis of the mobility parameter Y . The reference profile is depicted in black. The orange (resp. light blue) profile is the result of a 50% increase (resp. decrease) in mobility, with all other parameters remaining the same. F. Robustness analysis of the maximal sand slope parameter M_{slope} . The reference profile is depicted in black. The orange (resp. light blue) profile is the result of a 50% increase (resp. decrease) of M_{slope} , with all other parameters remaining the same.	54
Figure 1.10 Functional diagram of the multi-1D	55
Figure 1.11 Geography of the simulation near Montpellier	56
Figure 1.12 a) Initial sea bottom on the Gulf of Aigues-Mortes by adding a Gaussian geotube with a maximum height of 3m. b) Final bathymetry after a simulation of a storm of several days.	57
Figure 1.13 a) Errors \mathcal{E}_{L^2} (green) obtained by simulations of 10 different time steps compared to the reference simulation corresponding to a coupling time of 0.05 s. First order convergence (yellow). b) Errors \mathcal{E}_{L^2} (red) obtained by simulations of 10 different spatial steps compared to the reference simulation corresponding to a spatial step of 0.0002 m. First order convergence (yellow).	58
Figure 1.14 a) Different final bottom profiles for domains of sizes [600, 1200, 1800, 3000]m corresponding to offshore water heights of $h = [7, 14, 21, 35]$ m. The representation is truncated at 7 m depth. b) L^2 error comparing each simulation to the reference solution corresponding to the 3000 m domain (the longest, represents deep-water conditions).	59
Figure 1.15 Mid-term simulation of OptiMorph. A. Forcing wave height for scenario 1, composed of several mid-term events over a 2-day period. B. Forcing wave height for scenario 2, composed of numerous short-term events over a 2-day period. C. Forcing wave height for scenario 3, composed of several mid-term events over a 6-day period. D. Forcing wave height for scenario 4, composed of numerous short-term events over a 6-day period. E. Forcing wave height for scenario 5, composed of few mid-term events over a 6-day period. F. Seabeds resulting from the different forcing scenarios produced by OptiMorph. Two points of interest have been identified: P1 located at $x = 9.3$ m and P2 located at $x = 20.1$ m. G. Evolution of the distance, devoid of sediment, between the wave-maker (located at $x = 0$ m) and the seabed (WM-S), regarding forcing scenarios 3, 4, and 5. H. Vertical evolution of seabed elevation at P1, driven by the 6-day forcing scenarios 3, 4, and 5. I. Vertical evolution of seabed elevation at P2, driven by the 6-day forcing scenarios 3, 4, and 5. J. Evolution of shoreline position, driven by the 6-day forcing scenarios 3, 4, and 5.	62

Figure 2.1	Diagram of the linear theory	67
Figure 2.2	Illustration of notations.	74
Figure 2.3	Illustration of f_1 , f_2 and g defined in $[0, 1] \rightarrow [0, 1]$	75
Figure 2.4	Wave height $H(x)$ with the Shoaling model with the last improve (using f and g) for a configuration with a linear sea bottom of slopes about 0.11, wave period $T_0 = 10\text{s}$ and offshore water height of $H_0 = 2\text{m}$	75
Figure 2.5	Shallow-Water diagram for a 1D flow.	77
Figure 2.6	Shallow-Water diagram for a 1D flow and variable bottom.	79
Figure 2.7	Rayleigh density function $f(H, 1.9)$. The red area is the third-highest observed waves in a certain time series. The dashed line represents $H_{1/3}$, the mean of the third-highest observed waves also assimilated to the significant wave height H_s	81
Figure 2.8	Hydrodynamic results obtained with the Shoaling, SWAN, XBeach and Shallow-Water models. Bathymetric configurations from the LIP 1C channel experiment. Black points, measured HRMS, black bathymetry, green H from extended shoaling ($H_{RMSE} = 0.045\text{ m}$), red H from SWAN ($H_{RMSE} = 0.033\text{ m}$), blue H from XBeach ($H_{RMSE} = 0.028\text{ m}$) and orange from Shallow-Water model.	90
Figure 2.9	Hydrodynamic results obtained with the Shoaling, SWAN, XBeach and Shallow-Water models. Bathymetric configurations from a linear interpolation of the LIP 1C channel experiment. Black bathymetry, green H from extended shoaling ($H_{RMSE} = 0.059\text{ m}$), red H from SWAN ($H_{RMSE} = 0.024\text{ m}$), blue H from XBeach ($H_{RMSE} = 0.043\text{ m}$) and orange from Shallow-Water model.	90
Figure 2.10	Hydrodynamic results obtained with the Shoaling, SWAN, XBeach and Shallow-Water models. Bathymetric configurations from a linear, convex and concave open-sea configurations. Black bathymetry, green H from extended shoaling, red H from SWAN, blue H from XBeach and orange η from Shallow-Water model.	91
Figure 3.1	Representation of two sea bottom profiles ψ and $\psi + \varepsilon n$. To calculate the gradient, we need to calculate at all points the associated normal vector n	98
Figure 3.1	Illustration of ψ and $\psi + \varepsilon n$ with the function $\psi : x \rightarrow c$	99
Figure 3.2	Illustration of ψ , $\psi + \varepsilon n$ and A with the function $\psi : x \rightarrow ax + b$ and $A = \cos(\psi)$	100
Figure 3.3	A) Calculation of $\nabla_\psi H$ using Hadamard approximation with the following problem (see 3.2.2): $\psi = ax + b$, $H = \cos(\psi)$, with $a = 0.02$, $b = -2$. B) L2 error and order of convergence for a comparison between the analytical solution of the simple problem described in 3.2.2 with $H = \cos(\psi)$	101
Figure 3.4	Comparison of numerical and analytical solution of $\nabla_\psi H$ using OptiMorph model. Configuration without (A) / with (B) perturbations, $H_0 = 2\text{ m}$, $h_0 = 10\text{ m}$ and the wave period $T_0 = 10\text{s}$. In dodgerblue, the wave height H , in brown the bottom profile ψ , in red $\nabla_\psi H$ calculated analytically, in blue $\nabla_\psi H$ calculated by Hadamard strategy.	103
Figure 3.5	OptiMorph workflow coupled with wave model.	104

LIST OF FIGURES

Figure 3.6 A) Hydro-Morphodynamic results obtained with OptiMorph model using Hadamard strategy with wave models (Shoaling (green), SWAN (red) and XBeach (blue)). Bottom profile configuration from the SANDS erosive experience. Black: bottom profile, green: H and ψ from improved shoaling with Hadamard strategy, red: H and ψ from SWAN with Hadamard strategy, blue: H and ψ from XBeach with Hadamard strategy, dark red: ψ from experience. B) Morphodynamic ecarts of $\psi_f - \psi_i$ obtained with the Shoaling, SWAN, XBeach models and experiment. Bottom profile configuration from the SANDS channel experiment. In green: morphodynamic differences from shoaling with Hadamard strategy ($\psi_{RMSE} = 11.7$ cm), red: morphodynamic differences from SWAN with Hadamard strategy ($\psi_{RMSE} = 12.7$ cm), blue: morphodynamic differences from XBeach with Hadamard strategy ($\psi_{RMSE} = 13.5$ cm), dark red: morphodynamic differences from the experiment.	107
Figure 3.7 A) Hydro-Morphodynamic results obtained with OptiMorph model using Hadamard strategy with wave models (Shoaling (green), SWAN (red) and XBeach (blue)). Bottom profile configuration from the LIP 1C channel experiment. Red points: measured HRMS, black: bottom profile, green: H and ψ from improved shoaling with Hadamard strategy, red: H and ψ from SWAN with Hadamard strategy, blue: H and ψ from XBeach with Hadamard strategy, dark red: ψ from experience. B) Morphodynamic ecarts of $\psi_f - \psi_i$ obtained with the Shoaling, SWAN, XBeach models and experiment. Bottom profile configuration from the LIP 1C channel experiment. In green: morphodynamic differences from shoaling with Hadamard strategy, red: morphodynamic differences from SWAN with Hadamard strategy, blue: morphodynamic differences from XBeach with Hadamard strategy, dark red: morphodynamic differences from the experiment.	108
Figure 3.8 Evolution of ψ using Hadamard strategy with extended Shoaling (green), SWAN (red), XBeach (blue) and Shallow-Water (orange) models. Simulation on open-sea configuration with linear, convex and concave configurations. Simulation parameters of $H_0 = 2$ m, $T_0 = 12$ s, $h_0 = 20$ m, $\Omega = 600$ m.	109
Figure 3.9 A) Morphodynamic results by the OptiMorph model augmented by the transport mechanisms for $p = 0, 1, 2$ and the XBeach wave model, for the LIP 1C channel experiment. B) Velocity distribution for $p = 1$	112
Figure 3.10 Figure from (D. J. Roelvink et al. 2009): Measured and modeled bed level (with XBeach) after 1, 2, 4 and 8 h of wave action, for a water level of 4.56 m above the flume bottom	113
Figure 3.11 Hydro-morphodynamic results with different breaking criterion γ - Simulation with OptiMorph (Hadamard strategy) using SWAN model - $H_0 = 2$ m, $T_0 = 12$ s, $h_0 = 20$ m.	114
Figure 3.12 Comparison between H from models Shoaling (green), SWAN (red) XBeach (blue) and η_{RMS} from Shallow-Water (orange) models. H and η_{RMS} are on the same scale. A) Simulation parameters of $H_0 = 1.5$ m, $T_0 = 8$ s, $h_0 = 20$ m, $\Omega = 180$ m. B) Simulation parameters of $H_0 = 1.5$ m, $T_0 = 8$ s, $h_0 = 20$ m, $\Omega = 600$ m.	115
Figure 4.1 OptiMorph 2D using Hadamard with multi-1D simple shoaling models with linear bottom elevation. A) Seabed and Wave Height B) Superposition of analytical and numerical gradients C) Differences between analytical and numerical gradients.	120

Figure 4.2	OptiMorph 2D using Hadamard with multi-1D simple shoaling models with linear bottom elevation + geotube. A) Seabed and Wave Height B) Superposition of analytical and numerical gradients C) Differences between analytical and numerical gradients.	121
Figure 4.3	Nodes representation.	122
Figure 4.4	A) Schema of Copter's 2D configuration. B) Photo of the 2D Copter experiment.	123
Figure 4.5	(Top) Morphodynamic results using OptiMorph in 2D using the Multi-1D Shoaling model (left), REF/DIF (center) and comparing with experiments (right). (Bottom) Variability of the transects (in \vec{x} direction) with the mean (black) and standard deviation (gray).	124
Figure 4.6	Initial seabed with a linear slope of 1/100 with a geotextile tube at $x = 150$ m.	125
Figure 4.7	2D simulation of OptiMorph using Shoaling and REF/DIF models. Linear seabed (1/100) with geotube. Offshore height $H_0 = 2$ m and wave period $T_0 = 6$ s.	125
Figure 5.1	Illustration of the cross-shore profile where breaking occurs once at $x = x_B$	132
Figure 5.2	Illustration of notations.	134
Figure 5.3	Illustration of f_1 , f_2 and g defined in $[0, 1] \rightarrow [0, 1]$	134
Figure 5.4	Illustration of the orbital velocity over the cross-shore profile from (Cook 2021)	140
Figure 5.5	Variation of the parameter Λ over the cross-shore profile from (Cook 2021)	141
Figure 5.6	Illustration of gradient descent with $\psi \leq \alpha$. The optimum does not necessarily correspond to the critical point $\nabla_\psi \mathcal{J} = 0$	142
Figure 5.7	Representation of two sea bottom profiles ψ and $\psi + \varepsilon n$. To calculate the gradient, we need to calculate at all points the associated normal vector n	142
Figure 5.8	Slope limiter applied to a curve disturbed by heavisides functions	145
Figure 5.9	Slope constraint (5.23) from (Cook 2021)	146
Figure 5.10	Sand conservation (5.24) from (Cook 2021)	146
Figure 5.11	Illustration of the new direction of the descent in \mathbb{R}^2 : the direction $\nabla_\psi \mathcal{J}$ is projected onto the orthogonal of C_{sand} to yield d	147
Figure 5.12	OptiMorph workflow coupled with hydrodynamic model	148
Figure 5.13	Initial sandy beach configuration	164
Figure 5.14	Results halfway through the simulation	167
Figure 5.15	Results at the end of the simulation	167
Figure 5.16	Initial seabed at the beginning of the simulation	168
Figure 5.17	Seabed halfway through the simulation	168
Figure 5.18	Final seabed at the end of the simulation	168
Figure 5.19	Variation of the sand stock over time	168
Figure 5.20	Variation of d over time	168
Figure 5.21	Initial sandy beach configuration with a submerged breakwater located at $x = 600$ m	169
Figure 5.22	Results halfway through the simulation	172
Figure 5.23	Results at the end of the simulation	172
Figure 5.24	Initial seabed at the beginning of the simulation	173
Figure 5.25	Seabed halfway through the simulation	173
Figure 5.26	Final seabed at the end of the simulation	173
Figure 5.27	Variation of the sand stock over time	173
Figure 5.28	Variation of d over time	173

LIST OF FIGURES

Figure 5.29 Initial sandy beach configuration with a submerged breakwater located at $x = 150$ m	174
Figure 5.30 Initial sandy beach configuration with a submerged breakwater located at $x = 150$ m	176
Figure 5.31 Seabed halfway through the simulation	176
Figure 5.32 Final seabed at the end of the simulation	177
Figure 5.33 Seabed halfway through the simulation	177
Figure 5.34 Final seabed at the end of the simulation	178
Figure A.1 Illustration of a sinusoidal half-wrap of amplitude $A = 1$ and length $L = 1$	183
Figure A.2 Evolution of \mathcal{J} during duels.	185
Figure B.1 Forcing for the one-week storm simulation.	187
Figure B.2 Weaknesses (circled in turquoise) of the Copter simulation at the morphodynamic level.	187
Figure B.3 Results for different ϵ with the dissipation functional for simulation B.1.1.	190
Figure B.4 Results for different ϵ with the dissipation functional for simulation B.1.2.	191
Figure B.5 Rapports $\frac{J_C}{J_{tot}}$ and $\frac{J_H}{J_{tot}}$ at different times of the simulation: at the beginning, at the peak and at the end: according to different values of ϵ	192
Figure B.6 Diagram of momentum balance on a wave.	193
Figure B.7 Evolution of ψ_f bathymetry for different speeds with the descent transport equation.	196
Figure B.8 Evolution of the seabed ψ_f for different speeds with the descent transport equation.	196

LIST OF TABLES

Table 1.1	Table of different forcing parameters	42
Table 3.1	Computation time with 180 points calculated: LIP11D - 1C with different wave models. Simulations made with a 2.4 GHz computer using a single core on an Intel Xeon E5-2680 processor.	110
Table 3.2	Computation time with 600 points calculated with different wave models. Simulations made with a 2.4 GHz computer using a single core on an Intel Xeon E5-2680 processor.	110
Table 5.1	Input parameters	163
Table B.1	Parameter of the storm simulation.	186
Table B.2	Old cost functions \mathcal{J}	188
Table B.3	Nouvelles fonctions de coût \mathcal{J}	188
Table S.1	Simulation of Introduction	208
Table S.2	Simulation of Chapter 1	208
Table S.3	Simulation of Chapter 2	209
Table S.4	Simulation of Chapter 3	209
Table S.5	Simulation of Chapter 4	209
Table S.6	LIP 11 flume experiment parameters	210

LIST OF SIMULATIONS

Introduction		
Simulation description	Software	Bathymetry type ψ
<i>Hydrodynamic simulation to calculate different wave energies \mathcal{E}_H for different configurations</i>	XBeach	LIP 1B
<i>Computation of the optimal solution of ψ minimizing \mathcal{E}_H by genetic algorithm</i>	Python3 Algo-Genetic	Linear

Table S.1 – Simulation of Introduction

Chapter 1		
Simulation description	Software	Bathymetry type ψ
<i>Hydro-Morphodynamic simulation in 1D : short-term</i>	OptiMorph	Copter
<i>Hydro-Morphodynamic simulation in 1D : long-term</i>	OptiMorph	Copter
<i>Hydro-Morphodynamic simulation in Multi-1D</i>	OptiMorph	Near Montpellier

Table S.2 – Simulation of Chapter 1

Chapter 2		
Simulation description	Software	Bathymetry type ψ
<i>Hydrodynamic simulation in 1D</i>	Shoaling	LIP 1C
	SWAN	
	XBeach	
	Shallow-Water	
<i>Hydrodynamic simulation in 1D</i>	Shoaling	LIP 1C with linear interpolation
	SWAN	
	XBeach	
	Shallow-Water	

Table S.3 – Simulation of Chapter 2

Chapter 3		
Simulation description	Software	Bathymetry type ψ
<i>Hydrodynamic simulation in 1D to validate the Hadamard derivative with the analytical solution of Shoaling</i>	Shoaling	Linear Linear with perturbations
<i>Hydro-Morphodynamic simulation in 1D with different type of models using Hadamard approach</i>	Shoaling SWAN XBeach	SANDS
<i>Hydro-Morphodynamic simulation in 1D with different type of models using Hadamard approach</i>	Shoaling SWAN XBeach	LIP 1C
<i>Hydro-Morphodynamic simulation in 1D with diffrent type of models using Hadamard approach</i>	Shoaling SWAN XBeach Shallow-Water	Open-Sea configurations (linear, convex, concave)

Table S.4 – Simulation of Chapter 3

Chapter 4		
Simulation description	Software	Bathymetry type ψ
<i>Hydrodynamic simulation in 2D to validate the Hadamard derivative with the analytical solution of Shoaling</i>	Shoaling	Linear Linear with Géotube
<i>Hydro-Morphodynamic simulation in 2D with different type of models</i>	Multi-Shoaling REF/DIF	Copter 2D
<i>Hydro-Morphodynamic simulation in 2D with different type of models</i>	Multi-Shoaling REF/DIF	Linéar with Géotube

Table S.5 – Simulation of Chapter 4

LIST OF SIMULATIONS

Experiment	Initial Geometry	H_S [m]	T_P [s]	Duration [h]
LIP11-1A	Initial beach profile	0.9	5	
LIP11-1B	Result of 1A	1.4	5	18
LIP11-1C	Result of 1B	0.6	8	13

Table S.6 – LIP 11 flume experiment parameters

BIBLIOGRAPHY

- Airy, George Biddell (1845). *Tides and waves*. B. Fellowes.
- Alsina, José M. and Iván Cáceres (2011). “Sediment suspension events in the inner surf and swash zone. Measurements in large-scale and high-energy wave conditions”. In: *Coastal Engineering* 58.8, pp. 657–670. DOI: <https://doi.org/10.1016/j.coastaleng.2011.03.002>.
- Andrews, D. G. and M. E. McIntyre (1978). “An exact theory of nonlinear waves on a Lagrangian-mean flow”. In: *Journal of Fluid Mechanics* 89.4, pp. 609–646. DOI: [10.1017/S0022112078002773](https://doi.org/10.1017/S0022112078002773).
- Bagnold, Ralph Alger (1966). *An approach to the sediment transport problem from general physics*. US government printing office.
- Bascom, Willard N (1951). “The relationship between sand size and beach-face slope”. In: *Eos, Transactions American Geophysical Union* 32.6, pp. 866–874.
- Battjes, Jurjen A and JPJM Janssen (1978). “Energy loss and set-up due to breaking of random waves”. In: *Coastal engineering 1978*, pp. 569–587.
- Berkhoff, J.C.W. (1972). “COMPUTATION OF COMBINED REFRACTION - DIFFRACTION”. In: *Coastal Engineering Proceedings* 1.13, p. 23. DOI: [10.9753/icce.v13.23](https://doi.org/10.9753/icce.v13.23).
- Berthon, Christophe and Victor Michel-Dansac (2018). “A simple fully well-balanced and entropy preserving scheme for the shallow-water equations”. In: *Applied Mathematics Letters* 86, pp. 284–290.
- Biyanto, Totok Ruki, Henokh Yernias Fibrianto, Gunawan Nugroho, Agus Muhamad Hatta, Erny Listijorini, Titik Budiati, and Hairul Huda (2016). “Duelist Algorithm: An Algorithm Inspired by How Duelist Improve Their Capabilities in a Duel”. In: *Advances in Swarm Intelligence*. Ed. by Ying Tan, Yuhui Shi, and Ben Niu. Cham: Springer International Publishing, pp. 39–47.
- Blenkinsopp, Chris E, Paul M Bayle, Daniel C Conley, Gerd Masselink, Emily Gulson, Isabel Kelly, Rafael Almar, Ian L Turner, Tom E Baldock, Tomas Beuzen, et al. (2021).

BIBLIOGRAPHY

- “High-resolution, large-scale laboratory measurements of a sandy beach and dynamic cobble berm revetment”. In: *Scientific Data* 8.1, p. 22.
- Booij, N, LH Holthuijsen, and RC Ris (1996). “The" SWAN" wave model for shallow water”. In: *Coastal Engineering 1996*, pp. 668–676.
- Booij, N. (1983). “A note on the accuracy of the mild-slope equation”. In: *Coastal Engineering* 7.3, pp. 191–203. DOI: [https://doi.org/10.1016/0378-3839\(83\)90017-0](https://doi.org/10.1016/0378-3839(83)90017-0).
- Bouchette, Frédéric (2017). *Coastal defense strategy along Hatzuk beach (northern Tel Aviv, Israel). Insights from the COPTER physical experimentation with moveable bed.* Tech. rep. 17-1. Nîmes: BRL Ingénierie.
- Bouharguane, Afaf, Pascal Azerad, Frédéric Bouchette, Fabien Marche, and Bijan Mohammadi (2010). “Low complexity shape optimization & a posteriori high fidelity validation”. In: *Discrete and Continuous Dynamical Systems - B* 13.4, pp. 759–772. DOI: <10.3934/dcdsb.2010.13.759>.
- Bouharguane, Afaf and Bijan Mohammadi (2012). “Minimisation principles for the evolution of a soft sea bed interacting with a shallow sea”. In: *International Journal of Computational Fluid Dynamics* 26.3, pp. 163–172. DOI: <10.1080/10618562.2012.669831>.
- Breivik, Øyvind, Kristian Mogensen, Jean-Raymond Bidlot, Magdalena Alonso Balmaseda, and Peter AEM Janssen (2015). “Surface wave effects in the NEMO ocean model: Forced and coupled experiments”. In: *Journal of Geophysical Research: Oceans* 120.4, pp. 2973–2992.
- Briand, Marie-Hélène and J.William Kamphuis (1993). “Sediment transport in the surf zone: A quasi 3-D numerical model”. In: *Coastal Engineering* 20, pp. 135–156. DOI: [10.1016/0378-3839\(93\)90058-G](10.1016/0378-3839(93)90058-G).
- Bruun, Per. (1954). *Coast erosion and the development of beach profiles*. Technical memorandum - Beach Erosion Board ;no. 44. [Washington]: U.S. Beach Erosion Board.
- Bugajny, Natalia, Kazimierz Furmanczyk, Joanna Dudzinska-Nowak, and Barbara Paplińska-Swerpel (2013). “Modelling morphological changes of beach and dune induced by storm on the Southern Baltic coast using XBeach (case study: Dziwnow Spit)”. In: *Journal of Coastal Research* I, pp. 672–677. DOI: <10.2112/SI65-114.1>.
- Chen, W., J.J. van der Werf, and S.J.M.H. Hulscher (2023). “A review of practical models of sand transport in the swash zone”. In: *Earth-Science Reviews* 238, p. 104355. DOI: <https://doi.org/10.1016/j.earscirev.2023.104355>.
- Coeffe, Y and Ph Pechon (1982). “Modelling of sea-bed evolution under waves action”. In: *Proc. 18th ICCE* 1. DOI: <10.9753/icce.v18.71>.

- Cook, Megan (2021). "Calcul optimal pour la modélisation de la dynamique naturelle des plages sableuses et la conception d'ouvrages de défense du littoral à faible impact anthropique". Theses. Université Montpellier.
- Cook, Megan, Frédéric Bouchette, Bijan Mohammadi, and Nicolas Fraysse (2021a). "Opti-Morph User Guide". In: p. 50.
- Cook, Megan, Frédéric Bouchette, Bijan Mohammadi, and Nicolas Fraysse (2021b). "Application of Opti-Morph: Optimized beach protection by submerged geotextile tubes".
- Cook, Megan, Frédéric Bouchette, Bijan Mohammadi, Léa Sprunck, and Nicolas Fraysse (2021c). "Optimal Port Design Minimizing Standing Waves with A Posteriori Long Term Shoreline Sustainability Analysis". en. In: *China Ocean Engineering* 35.6, pp. 802–813. DOI: [10.1007/s13344-021-0071-7](https://doi.org/10.1007/s13344-021-0071-7).
- Daly, C (2009). "Low Frequency Waves in the Shoaling and Nearshore Zone A Validation of XBeach". In: *Erasmus Mundus Master in Coastal and Marine Engineering and Management (CoMEM), Delft University of Technology*.
- Dean, Robert and Robert Dalrymple (2004). "Coastal Processes With Engineering Applications". In: *Coastal Processes with Engineering Applications, by Robert G. Dean and Robert A. Dalrymple, pp. 487. ISBN 0521602750. Cambridge, UK: Cambridge University Press, March 2004*.
- Ding, Yan, Sam Wang, and Yafei Jia (2006). "Development and Validation of a Quasi-Three-Dimensional Coastal Area Morphological Model". In: *Journal of Waterway Port Coastal and Ocean Engineering* 132, pp. 462–476. DOI: [10.1061/\(ASCE\)0733-950X\(2006\)132:6\(462\)](https://doi.org/10.1061/(ASCE)0733-950X(2006)132:6(462)).
- Droenen, Nils and Rolf Deigaard (2007). "Quasi-three-dimensional modelling of the morphology of longshore bars". In: *Coastal Engineering* 54, pp. 197–215. DOI: [10.1016/j.coastaleng.2006.08.011](https://doi.org/10.1016/j.coastaleng.2006.08.011).
- Dupont, R., F. Bouchette, and B. Mohammadi (2024). "Modelling beaches morphodynamic by Hadamard sensitivity analysis". In: *Ocean Modelling* 189, p. 102370. DOI: <https://doi.org/10.1016/j.ocemod.2024.102370>.
- Dupont, Ronan, Frédéric Bouchette, Bijan Mohammadi, and Damien Sous (2022). "OptiMorph: un modèle de morphodynamique du littoral par principe de minimisation. Analyse de sensibilité en 1D et application Multi-1D". In: *JNGCGC* 17, pp. 327–336. DOI: [10.5150/jngcgc.2022.034](https://doi.org/10.5150/jngcgc.2022.034).
- Dupont, Ronan, Megan Cook, Frédéric Bouchette, Bijan Mohammadi, and Samuel Meulé (2023). "Sandy beach dynamics by constrained wave energy minimization". In: *Ocean Modelling*, p. 102197. DOI: <https://doi.org/10.1016/j.ocemod.2023.102197>.

BIBLIOGRAPHY

- Eichentopf, Sonja, Iván Cáceres, and José M. Alsina (2018). “Breaker bar morphodynamics under erosive and accretive wave conditions in large-scale experiments”. In: *Coastal Engineering* 138, pp. 36–48. DOI: <https://doi.org/10.1016/j.coastaleng.2018.04.010>.
- Engineers, US Army Corps of (2002). *Coastal Engineering Manual, Engineer Manual 1110-2-1100*. Washington, D.C.: US Army Corps of Engineers.
- Euler, Leonhard (1752). “Découverte d’un nouveau principe de mécanique”. In: *Mémoires de l’académie des sciences de Berlin*, pp. 185–217.
- Fleming, C. and J. Hunt (1977). “Application of Sediment Transport Model”. In: pp. 1184–1202. DOI: [10.1061/9780872620834.070](https://doi.org/10.1061/9780872620834.070).
- Fowler, A. C. (2001). “Dunes and Drumlins”. In: *Geomorphological Fluid Mechanics*. Ed. by N. J. Balmforth and A. Provenzale. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 430–454. DOI: [10.1007/3-540-45670-8_18](https://doi.org/10.1007/3-540-45670-8_18).
- Fredsøe, Jørgen and Rolf Deigaard (1992). *Mechanics of Coastal Sediment Transport*. WORLD SCIENTIFIC. DOI: [10.1142/1546](https://doi.org/10.1142/1546).
- Freilich, MH and RT Guza (1984). “Nonlinear effects on shoaling surface gravity waves”. In: *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences* 311.1515, pp. 1–41.
- Galappatti, G and CB Vreugdenhil (1985). “A depth-integrated model for suspended sediment transport”. In: *Journal of Hydraulic Research* 23.4, pp. 359–377.
- Gâteaux, René (1913). “Sur les fonctionnelles continues et les fonctionnelles analytiques”. In: *CR Acad. Sci. Paris* 157.325–327, p. 65.
- Grasso, F., H. Michallet, and E. Barthélémy (2011). “Experimental simulation of shoreface nourishments under storm events: A morphological, hydrodynamic, and sediment grain size analysis”. In: *Coastal Engineering* 58.2, pp. 184–193. DOI: <https://doi.org/10.1016/j.coastaleng.2010.09.007>.
- Gravens, Mark (1997). “An Approach to Modeling Inlet and Beach Evolution”. In: pp. 4477–4490. DOI: [10.1061/9780784402429.348](https://doi.org/10.1061/9780784402429.348).
- Griewank, Andreas and Andrea Walther (2008). *Evaluating derivatives: principles and techniques of algorithmic differentiation*. SIAM.
- Hadamard, J (1914). “On the Maximum Module of a Function and Its Derivatives”. In: *CR Acad. Sci. Paris* 41, pp. 68–72.
- Hascoet, L. and V. Pascual (2004). “Tapenade user’s guide”. In: *INRIA Technical report*. INRIA, pp. 1–31.

- Hasselmann, Klaus, T.P. Barnett, E. Bouws, H. Carlson, D. Cartwright, K. Enke, J.A. Ewing, H. Gienapp, D. Hasselmann, P. Kruseman, A. Meerburg, Peter Muller, Dirk Olbers, K. Richren, W. Sell, and H. Walden (1973). *Measurements of wind-wave growth and swell decay during the Joint North Sea Wave Project (JONSWAP)*. Tech. rep. Hamburg, Germany: Deutsches Hydrographisches Institut, pp. 1–95.
- Hattori, Masataro and Ryoichi Kawamata (1980). “Onshore-Offshore Transport and Beach Profile Change”. In: *Coastal Engineering 1980*, pp. 1175–1193. DOI: [10.1061/9780872622647.072](https://doi.org/10.1061/9780872622647.072).
- Helmholtz, Professor (1868). “XLIII. On discontinuous movements of fluids”. In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 36.244, pp. 337–346.
- Holthuijsen, Leo, N. Booij, and T.H.C. Herbers (1989). ““A prediction model for stationary, short crested waves in shallow water with ambient current””. In: *Coastal Engineering* 13, pp. 23–54. DOI: [10.1016/0378-3839\(89\)90031-8](https://doi.org/10.1016/0378-3839(89)90031-8).
- Isèbe, Damien, Pascal Azérad, Frédéric Bouchette, Benjamin Ivorra, and Bijan Mohammadi (2008a). “Shape optimization of geotextile tubes for sandy beach protection”. In: *International Journal for Numerical Methods in Engineering* 74.8, pp. 1262–1277. DOI: [10.1002/nme.2209](https://doi.org/10.1002/nme.2209).
- Isèbe, Damien, Pascal Azérad, Frédéric Bouchette, and Bijan Mohammadi (2014). “Design of Passive Defense Structures in Coastal Engineering”. en. In: *International Review of Civil Engineering (IRECE)* 5.2, p. 75. DOI: [10.15866/irece.v5i2.2029](https://doi.org/10.15866/irece.v5i2.2029).
- Isèbe, Damien, Pascal Azérad, Bijan Mohammadi, and Frédéric Bouchette (2008b). “Optimal shape design of defense structures for minimizing short wave impact”. In: *Coastal Engineering* 55.1, pp. 35–46. DOI: [10.1016/j.coastaleng.2007.06.006](https://doi.org/10.1016/j.coastaleng.2007.06.006).
- Johnson, Hakeem, Ida Brøker, and Julio Zyberman (1995). “Identification of Some Relevant Processes in Coastal Morphological Modelling”. In: pp. 2871–2885. DOI: [10.1061/9780784400890.208](https://doi.org/10.1061/9780784400890.208).
- Kana, Timothy, E.J. Hayter, and Paul Work (1999). “Mesoscale sediment transport at Southeastern U.S. tidal inlets: conceptual model applicable to mixed energy settings”. In: *Journal of Coastal Research* 15, pp. 303–313.
- Kim, Jang Whan, Kwang June Bai, R Cengiz Ertekin, and William C Webster (2001). “A derivation of the Green-Naghdi equations for irrotational flows”. In: *Journal of Engineering Mathematics* 40, pp. 17–42.
- Kirby, James T (1986a). “Higher-order approximations in the parabolic equation method for water waves”. In: *Journal of Geophysical Research: Oceans* 91.C1, pp. 933–952.

BIBLIOGRAPHY

- Kirby, James T. (1986b). "Rational approximations in the parabolic equation method for water waves". In: *Coastal Engineering* 10.4, pp. 355–378. DOI: [https://doi.org/10.1016/0378-3839\(86\)90021-9](https://doi.org/10.1016/0378-3839(86)90021-9).
- Kirby, James Thornton and Robert A Dalrymple (1994). *Combined Refraction/Diffraction Model REF/DIF 1, Version 2.5: Documentation and User's Manual*. Center for Applied Coastal Research, Department of Civil Engineering ...
- Kouakou, Kouamé Kan Jacques and Pierre-Yves Lagrée (2006). "Evolution of a model dune in a shear flow". In: *European Journal of Mechanics - B/Fluids* 25.3, pp. 348–359. DOI: <https://doi.org/10.1016/j.euromechflu.2005.09.002>.
- Larson, Magnus and Nicholas Kraus (1989). *SBEACH: Numerical Model for Simulating Storm-Induced Beach Change. Report 1. Empirical Foundation and Model Development*. Tech. rep. Washington, DC, USA: DEPARTMENT OF THE ARMY US Army Corps of Engineers, p. 266.
- Larson, Magnus, Nicholas Kraus, and Mark Byrnes (1990). *SBEACH: Numerical Model for Simulating Storm-Induced Beach Change. Report 2. Numerical Formulation and Model Tests*. Tech. rep. Washington, DC, USA: DEPARTMENT OF THE ARMY US Army Corps of Engineers, p. 120.
- Latteux, B. (1980). "Harbour Design Including Sedimentological Problems Using Mainly Numerical Technics". In: pp. 2213–2229. DOI: <10.1061/9780872622647.133>.
- Lesser, G.R., Dano J.A. Roelvink, J.A.T.M. Kester, and Guus Stelling (2004). "Development and Validation of a Three-Dimensional Morphological Model". In: *Coastal Engineering* 51, pp. 883–915. DOI: <10.1016/j.coastaleng.2004.07.014>.
- Liu, Philip L.-F. and Inigo J. Losada (2002). "Wave propagation modeling in coastal engineering". In: *Journal of Hydraulic Research* 40.3, pp. 229–240. DOI: <10.1080/00221680209499939>.
- Longuet-Higgins, Michael S (1963). "The effect of non-linearities on statistical distributions in the theory of sea waves". In: *Journal of fluid mechanics* 17.3, pp. 459–480.
- Longuet-Higgins, Michael S and RW Stewart (1962). "Radiation stress and mass transport in gravity waves, with application to 'surf beats'". In: *Journal of Fluid Mechanics* 13.4, pp. 481–504.
- Lv, Xiangcui, Dekui Yuan, Xiaodi Ma, and Jianhua Tao (2014). "Wave characteristics analysis in Bohai Sea based on ECMWF wind field". In: *Ocean Engineering* 91, pp. 159–171.
- Madsen, Per A., Russel Murray, and Ole R. Sørensen (1991). "A new form of the Boussinesq equations with improved linear dispersion characteristics". In: *Coastal Engineer-*

- ing* 15.4, pp. 371–388. DOI: [https://doi.org/10.1016/0378-3839\(91\)90017-B](https://doi.org/10.1016/0378-3839(91)90017-B).
- Madsen, Per A. and Ole R. Sørensen (1992). “A new form of the Boussinesq equations with improved linear dispersion characteristics. Part 2. A slowly-varying bathymetry”. In: *Coastal Engineering* 18.3, pp. 183–204. DOI: [https://doi.org/10.1016/0378-3839\(92\)90019-Q](https://doi.org/10.1016/0378-3839(92)90019-Q).
- Marche, F, Ph Bonneton, P Fabrie, and Nicolas Seguin (2007). “Evaluation of well-balanced bore-capturing schemes for 2D wetting and drying processes”. In: *International Journal for Numerical Methods in Fluids* 53.5, pp. 867–894.
- Marche, Fabien (2007). “Derivation of a new two-dimensional viscous shallow water model with varying topography, bottom friction and capillary effects”. In: *European Journal of Mechanics - B/Fluids* 26.1, pp. 49–63. DOI: <https://doi.org/10.1016/j.euromechflu.2006.04.007>.
- Marchesiello, Patrick, Francis Auclair, Laurent Debreu, James McWilliams, Rafael Almar, Rachid Benshila, and Franck Dumas (2021). “Tridimensional nonhydrostatic transient rip currents in a wave-resolving model”. In: *Ocean Modelling* 163, p. 101816.
- Marchesiello, Patrick, Julien Chauchat, Hassan Shafiei, Rafael Almar, Rachid Benshila, Franck Dumas, and Laurent Debreu (2022). “3D wave-resolving simulation of sandbar migration”. In: *Ocean Modelling* 180, p. 102127. DOI: <https://doi.org/10.1016/j.ocemod.2022.102127>.
- Martins, Kévin, Philippe Bonneton, Paul M. Bayle, Chris E. Blenkinsopp, Arthur Mouragues, and Hervé Michallet (2020). “Surf Zone Wave Measurements from Lidar Scanners: Analysis of Non-hydrostatic Processes”. In: *Journal of Coastal Research* 95.SI, pp. 1189–1194. DOI: [10.2112/SI95-231.1](https://doi.org/10.2112/SI95-231.1).
- Maruyama, K. and T. Takagi (1988). “A simulation system of near-shore sediment transport for the coupling of the sea-bottom topography, waves and currents”. In: *Proc. IAHR Symp. Math. Mod. Sed. Transp. Coastal Zone*, pp. 300–309.
- McCall, Robert T, JSM Van Thiel De Vries, NG Plant, AR Van Dongeren, JA Roelvink, DM Thompson, and AJHM Reniers (2010). “Two-dimensional time dependent hurricane overwash and erosion modeling at Santa Rosa Island”. In: *Coastal Engineering* 57.7, pp. 668–683.
- McCarroll, R. Jak, Gerd Masselink, Nieves G. Valiente, Timothy Scott, Mark Wiggins, Josie-Alice Kirby, and Mark Davidson (2021). “A rules-based shoreface translation and sediment budgeting tool for estimating coastal change: ShoreTrans”. In: *Marine Geology* 435, p. 106466. DOI: <https://doi.org/10.1016/j.margeo.2021.106466>.

BIBLIOGRAPHY

- Mei, Chiang C and EO Tuck (1980). “Forward scattering by long thin bodies”. In: *SIAM Journal on Applied Mathematics* 39.1, pp. 178–191.
- Mohammadi, Bijan (2007). “Global optimization, level set dynamics, incomplete sensitivity and regularity control”. en. In: *International Journal of Computational Fluid Dynamics* 21.2, pp. 61–68. DOI: [10.1080/10618560701455855](https://doi.org/10.1080/10618560701455855).
- (2010). “Hadamard incomplete sensitivity and shape optimization”. eng. In: *Control and Cybernetics* 39.3, pp. 615–626.
- (2017). “Uncertainty quantification in littoral erosion”. In: *Computers & Fluids* 143, pp. 120–133. DOI: <https://doi.org/10.1016/j.compfluid.2016.10.017>.
- Mohammadi, Bijan and Frédéric Bouchette (2014). “Extreme scenarios for the evolution of a soft bed interacting with a fluid using the Value at Risk of the bed characteristics”. In: *Computers and Fluids* 89, pp. 78–87. DOI: [10.1016/j.compfluid.2013.10.021](https://doi.org/10.1016/j.compfluid.2013.10.021).
- Mohammadi, Bijan and Afaf Bouharguane (2011). “Optimal dynamics of soft shapes in shallow waters”. In: *Computers and Fluids* 40, pp. 291–298. DOI: [10.1016/j.compfluid.2010.09.031](https://doi.org/10.1016/j.compfluid.2010.09.031).
- Mohammadi, Bijan and Olivier Pironneau (2009). *Applied Shape Optimization for Fluids*. Oxford University Press. DOI: [10.1093/acprof:oso/9780199546909.001.0001](https://doi.org/10.1093/acprof:oso/9780199546909.001.0001).
- Munk, David J, Gareth A Vio, and Grant P Steven (2015). “Topology and shape optimization methods using evolutionary algorithms: a review”. In: *Structural and Multidisciplinary Optimization* 52, pp. 613–631.
- Munk, Walter (1949). “The solitary wave theory and its application to surf problems”. In: *Annals of the New York Academy of Sciences* 51, pp. 376–424. DOI: [10.1111/j.1749-6632.1949.tb27281.x](https://doi.org/10.1111/j.1749-6632.1949.tb27281.x).
- Murray, A. Brad (2007). “Reducing model complexity for explanation and prediction”. In: *Geomorphology* 90.3, pp. 178–191. DOI: <https://doi.org/10.1016/j.geomorph.2006.10.020>.
- Nairn, Robert and Howard Southgate (1993). “Deterministic profile modelling of nearshore processes. Part 2. Sediment transport and beach profile development”. In: *Coastal Engineering* 19, pp. 57–96. DOI: [10.1016/0378-3839\(93\)90019-5](https://doi.org/10.1016/0378-3839(93)90019-5).
- Nicholson, J., Ida Brøker, Dano J.A. Roelvink, D. Price, Jean-Michel Tanguy, and L. Moreno (1997). “Intercomparison of coastal area morphodynamic models”. In: *Coastal Engineering - COAST ENG* 31, pp. 97–123. DOI: [10.1016/S0378-3839\(96\)00054-3](https://doi.org/10.1016/S0378-3839(96)00054-3).

- Nielsen, Peter (1992). *Coastal bottom boundary layers and sediment transport*. Vol. 4. World scientific.
- (2002). “Shear stress and sediment transport calculations for swash zone modelling”. In: *Coastal Engineering* 45.1, pp. 53–60. DOI: [https://doi.org/10.1016/S0378-3839\(01\)00036-9](https://doi.org/10.1016/S0378-3839(01)00036-9).
- Nussbaumer, Henri J and Henri J Nussbaumer (1982). *The fast Fourier transform*. Springer.
- Nwogu, Okey (1993). “Alternative Form of Boussinesq Equations for Nearshore Wave Propagation”. In: *Journal of Waterway, Port, Coastal, and Ocean Engineering* 119.6, pp. 618–638. DOI: [10.1061/\(ASCE\)0733-950X\(1993\)119:6\(618\)](https://doi.org/10.1061/(ASCE)0733-950X(1993)119:6(618)).
- Paola, C. and V. R. Voller (2005). “A generalized Exner equation for sediment mass balance”. In: *Journal of Geophysical Research: Earth Surface* 110.F4. DOI: <https://doi.org/10.1029/2004JF000274>.
- Pender, Doug and Harshnie Karunarathna (2013). “A statistical-process based approach for modelling beach profile variability”. In: *Coastal Engineering* 81, pp. 19–29.
- Peregrine, D. H. (1967). “Long waves on a beach”. In: *Journal of Fluid Mechanics* 27.4, pp. 815–827. DOI: [10.1017/S0022112067002605](https://doi.org/10.1017/S0022112067002605).
- Quick, M. (1991). “Onshore-offshore sediment transport on beaches”. In: *Coastal Engineering* 15, pp. 313–332.
- Reineck, H.-E. and I. B. Singh (1973). *Depositional sedimentary environments; with reference to terrigenous clastics [by] H.-E. Reineck [and] I. B. Singh*. English. Springer-Verlag Berlin, New York, xvi, 439 p.
- Reniers, Ad and Marcel Zijlema (2022). “SWAN SurfBeat-1D”. In: *Coastal Engineering* 172, p. 104068. DOI: <https://doi.org/10.1016/j.coastaleng.2021.104068>.
- Roelvink, Dano J.A. (1993). “Dissipation in random wave groups incident on a beach”. In: *Coastal Engineering - COAST ENG* 19, pp. 127–150. DOI: [10.1016/0378-3839\(93\)90021-Y](https://doi.org/10.1016/0378-3839(93)90021-Y).
- Roelvink, Dano J.A., Ad Reniers, Ap van Dongeren, Jaap Thiel de Vries, Jamie Lescinski, and Robert McCall (2010). *XBeach Model – Description and Manual*. Tech. rep. Delft, Netherlands: Unesco-IHE Institute for Water Education, Deltares and Delft University of Technology.
- Roelvink, Dano J.A., Ad Reniers, Ap van Dongeren, Jaap Thiel de Vries, Robert McCall, and Jamie Lescinski (2009). “Modelling storm impacts on beaches, dunes and barrier islands”. In: *Coastal Engineering* 56, pp. 1133–1152. DOI: [10.1016/j.coastaleng.2009.08.006](https://doi.org/10.1016/j.coastaleng.2009.08.006).

BIBLIOGRAPHY

- Roelvink, Dano J.A., Dirk-Jan Walstra, and Z. Chen (1995). “Morphological Modelling of Keta Lagoon Case”. In: pp. 3223–3236. DOI: [10.1061/9780784400890.233](https://doi.org/10.1061/9780784400890.233).
- Roelvink, J. A., G. K. F. M. Van Banning, and A. Verwey (1994). “Design and development of DELFT3D and application to coastal morphodynamics, 1st International conference, Hydroinformatics 94”. In: *Hydroinformatics 94, HYDROINFORMATICS -PROCEEDINGS-, 1st International conference, Hydroinformatics 94*. Vol. 1. Rotterdam: Balkema, pp. 451–456.
- Roelvink, J.A. (2006). “Coastal morphodynamic evolution techniques”. In: *Coastal Engineering* 53.2, pp. 277–287. DOI: <https://doi.org/10.1016/j.coastaleng.2005.10.015>.
- Roelvink, JA and Ad Reniers (1995). *LIP 11D Delta Flume experiments: a dataset for profile model validation*. WL/Delft Hydraulics.
- Rooijen, Arnold van, Ad Reniers, Jaap van Thiel de Vries, Chris Blenkinsopp, and Robert McCall (2012). “Modeling swash zone sediment transport at Truc Vert Beach”. In: *Coastal Engineering Proceedings* 1.33, sediment.105. DOI: [10.9753/icce.v33.sediment.105](https://doi.org/10.9753/icce.v33.sediment.105).
- Ruessink, Gerben and J.H.J Terwindt (2000). “The behaviour of nearshore bars on the time scale of years: A conceptual model”. In: *Marine Geology* 163, pp. 289–302. DOI: [10.1016/S0025-3227\(99\)00094-8](https://doi.org/10.1016/S0025-3227(99)00094-8).
- Rusu, Liliana and C. Guedes Soares (2012). “Wave energy assessments in the Azores islands”. In: *Renewable Energy* 45, pp. 183–196. DOI: <https://doi.org/10.1016/j.renene.2012.02.027>.
- Schimmels, Stefan and Chris Blenkinsopp (2020). “DynaRev-Dynamic Coastal Protection: Resilience of Dynamic Revetments Under Sea Level Rise”. In.
- Serrano-Pacheco, Alberto, Javier Murillo, and Pilar Garcia-Navarro (2012). “Finite volumes for 2D shallow-water flow with bed-load transport on unstructured grids”. In: *Journal of Hydraulic Research* 50.2, pp. 154–163.
- Shafiei, Hassan, Julien Chauchat, Cyrille Bonamy, and Patrick Marchesiello (2023). “Adaptation of the SANTOSS transport formula for 3D nearshore models: Application to cross-shore sandbar migration”. In: *Ocean Modelling* 181, p. 102138. DOI: <https://doi.org/10.1016/j.ocemod.2022.102138>.
- Shchepetkin, Alexander F and James C McWilliams (2005). “The regional oceanic modeling system (ROMS): a split-explicit, free-surface, topography-following-coordinate oceanic model”. In: *Ocean modelling* 9.4, pp. 347–404.

- Short, Andrew D. (1996). "The role of wave height, period, slope, tide range and embayment in beach classifications: A review". In: *Revista Chilena de Historia Natural* 69, pp. 589–604.
- Small, Christopher and Robert J. Nicholls (2003). "A Global Analysis of Human Settlement in Coastal Zones". In: *Journal of Coastal Research* 19.3, pp. 584–599.
- Soulsby, R.L. (1987). "Calculating bottom orbital velocity beneath waves". In: *Coastal Engineering - COAST ENG* 11, pp. 371–380. DOI: [10.1016/0378-3839\(87\)90034-2](https://doi.org/10.1016/0378-3839(87)90034-2).
- Soulsby, Richard (1997). *Dynamics of marine sands*. Thomas Telford Publishing. DOI: [10.1680/doms.25844](https://doi.org/10.1680/doms.25844).
- Sous, Damien, Guillaume Dodet, Frédéric Bouchette, and M Tissier (2020). "Momentum balance across a barrier reef". In: *Journal of Geophysical Research: Oceans* 125.2, e2019JC015503.
- Stokes, George Gabriel (1847). "On the theory of oscillatory waves". In: *Trans. Cam. Philos. Soc.* 8, pp. 441–455.
- Storms, Joep E.A., G.J. Weltje, J.J. van Dijke, C.R. Geel, and S.B. Kroonenberg (2002). "Process-Response Modeling of Wave-Dominated Coastal Systems: Simulating Evolution and Stratigraphy on Geological Timescales". In: *Journal of Sedimentary Research* 72.2, pp. 226–239. DOI: [10.1306/052501720226](https://doi.org/10.1306/052501720226).
- Tayfun, M Aziz (1980). "Narrow-band nonlinear sea waves". In: *Journal of Geophysical Research: Oceans* 85.C3, pp. 1548–1552.
- Tolman, Hendrik L et al. (2009). "User manual and system documentation of WAVEWATCH III TM version 3.14". In: *Technical note, MMAB Contribution* 276.220.
- Vos, Kilian, Mitchell D Harley, Kristen D Splinter, Andrew Walker, and Ian L Turner (2020). "Beach slopes from satellite-derived shorelines". In: *Geophysical Research Letters* 47.14, e2020GL088365.
- Vriend, Huib de, W.T. Bakker, and D.P. Bilse (1994). "A morphological behaviour model for the outer delta of mixed-energy tidal inlets". In: *Coastal Engineering* 23, pp. 305–327. DOI: [10.1016/0378-3839\(94\)90008-6](https://doi.org/10.1016/0378-3839(94)90008-6).
- Walstra, Dirk-Jan, Dano J.A. Roelvink, and Jacco Groeneweg (2000). "Calculation of Wave-Driven Currents in a 3D Mean Flow Model". In: vol. 276. DOI: [10.1061/40549\(276\)81](https://doi.org/10.1061/40549(276)81).
- Wang, Hsiang, Gang Miao, and Li-Hwa Lin (1993). "A Time—Dependent Nearshore Morphological Response Model". In: pp. 2513–2527. DOI: [10.1061/9780872629332.192](https://doi.org/10.1061/9780872629332.192).

BIBLIOGRAPHY

- Watanabe, Akira, Kohki Maruyama, Takao Shimizu, and Tsutomu Sakakiyama (1986). “Numerical Prediction Model of Three-Dimensional Beach Deformation Around a Structure”. In: *Coastal Engineering Journal* 29, pp. 179–194. DOI: [10.1080/05785634.1986.11924437](https://doi.org/10.1080/05785634.1986.11924437).
- Whitham, G. B. (1965). “A general approach to linear and non-linear dispersive waves using a Lagrangian”. In: *Journal of Fluid Mechanics* 22.2, pp. 273–283.
- Whitham, Gerald Beresford (2011). *Linear and nonlinear waves*. John Wiley & Sons.
- Wiberg, Patricia L. and Christopher R. Sherwood (2008). “Calculating wave-generated bottom orbital velocities from surface-wave parameters”. In: *Computers & Geosciences* 34.10, pp. 1243–1262. DOI: <https://doi.org/10.1016/j.cageo.2008.02.010>.
- Williams, J., Luciana Esteves, and Liam Rochford (2015). “Modelling storm responses on a high-energy coastline with XBeach”. In: *Modeling Earth Systems and Environment* 1. DOI: [10.1007/s40808-015-0003-8](https://doi.org/10.1007/s40808-015-0003-8).
- Wright, L.D and A.D Short (1984). “Morphodynamic variability of surf zones and beaches: A synthesis”. In: *Marine Geology* 56.1, pp. 93–118. DOI: [https://doi.org/10.1016/0025-3227\(84\)90008-2](https://doi.org/10.1016/0025-3227(84)90008-2).
- Yamaguchi, Masataka and Y Nishioka (1985). “Numerical Simulation on the Change of Bottom Topography by the Presence of Coastal Structures”. In: pp. 1732–1748. DOI: [10.1061/9780872624382.118](https://doi.org/10.1061/9780872624382.118).
- Yang, Chih Ted et al. (1996). “Sediment transport: theory and practice”. In.
- Yue, Dick K. P. and Chiang C. Mei (1980). “Forward diffraction of Stokes waves by a thin wedge”. In: *Journal of Fluid Mechanics* 99.1, pp. 33–52. DOI: [10.1017/S0022112080000481](https://doi.org/10.1017/S0022112080000481).
- Zimmermann, Nicolas, Koen Trouw, Li Wang, Mieke Mathys, Rosalia Delgado, and Toon Verwaest (2012). “Longshore transport and sedimentation in a navigation channel at Blankenberge (Belgium)”. In: *Coastal Engineering Proceedings* 1. DOI: [10.9753/icce.v33.sediment.111](https://doi.org/10.9753/icce.v33.sediment.111).
- Zyberman, Julio and Hakeem Johnson (2002). “Modelling morphological processes in the vicinity of shore-parallel breakwaters”. In: *Coastal Engineering* 45, pp. 261–284. DOI: [10.1016/S0378-3839\(02\)00037-6](https://doi.org/10.1016/S0378-3839(02)00037-6).