

Bathymetry reconstruction from experimental data using PDE-constrained optimisation

Judith Angel^{a,*}, Jörn Behrens^b, Sebastian Götschel^a, Marten Hollm^c, Daniel Ruprecht^a, Robert Seifried^c

^a Chair Computational Mathematics, Institute of Mathematics, Hamburg University of Technology, Am Schwarzenberg-Campus 3, Hamburg 21073, Germany

^b Department of Mathematics, Universität Hamburg, Bundesstraße 55, Hamburg 20146, Germany

^c Institute of Mechanics and Ocean Engineering, Hamburg University of Technology, Eißendorfer Straße 42, Hamburg 21073, Germany

ARTICLE INFO

Dataset link: <https://doi.org/10.15480/882.9403>

Keywords:

Bathymetry reconstruction
Shallow water equations
Wave flume experiment
PDE-constrained optimisation

ABSTRACT

Knowledge of the bottom topography, also called bathymetry, of rivers, seas or the ocean is important for many areas of maritime science and civil engineering. While direct measurements are possible, they are time consuming, expensive and inaccurate. Therefore, many approaches have been proposed how to infer the bathymetry from measurements of surface waves. Mathematically, this is an inverse problem where an unknown system state needs to be reconstructed from observations with a suitable model for the flow as constraint. In many cases, the shallow water equations can be used to describe the flow. While theoretical studies of the efficacy of such a PDE-constrained optimisation approach for bathymetry reconstruction exist, there seem to be few publications that study its application to data obtained from real-world measurements. This paper shows that the approach can, at least qualitatively, reconstruct a Gaussian-shaped bathymetry in a wave flume from measurements of the free surface level at up to three points. Achieved normalised root mean square errors (NRMSE) are in line with other approaches.

1. Introduction

Knowing topography of river beds or sea or ocean floors, also called bathymetry, is important for many areas of science and engineering. Two of many examples would be the prediction of meteorological tsunamis [1] or the prediction of the circulation of hazardous materials in estuaries [2]. Knowledge of shallow water bathymetry is also important for nautical navigation to avoid stranding.

Various methods exist to measure bathymetry. Using sonar measurements from research vessels like ships or remotely operated vehicles, the bathymetry can be measured on a width as much as twice the water depth [3]. However, this technique is time-consuming, expensive and there is the risk of stranding [4]. In a second technique, the bathymetry is measured by an airborne LIDAR (Light Detection And Ranging) system. Airborne LIDAR bathymetry systems can be used to remotely collect accurate, high-density measurements for both bathymetry and topography of dry areas in coastal regions [5]. However, this technique is still expensive and only works in areas that allow flying [4]. As an alternative approach, satellite images can be used to obtain information about the bathymetry of areas that are difficult to reach by airplane or boat. Depth information obtained by analysing satellite images is called satellite derived bathymetry (SDB) [6]. Using satellites, bathymetry information of a large area can be tracked much faster than by

oceanographic vessels [3]. However, data quality and pre-processing steps comprise a critical part of the SDB. Special attention is needed, for example, regarding atmospheric effects, sun reflection, and the ripple of the sea surface [7].

Bathymetry information can also be obtained from measurements of the height of waves on the water surface. The advantage of this approach is that it only requires sea surface measurements which are often easier and cheaper to obtain than measurements of the sea floor. Mathematically speaking, finding the bathymetry from measuring surface waves is an inverse problem where an unknown state of a system is being inferred from observations. There are a variety of approaches in the literature on how to solve this inverse problem. Many papers use a form of minimisation with forward and adjoint models to compute a descent direction. Khan et al. [8,9] reconstruct the bathymetry using the one-dimensional shallow water equations (SWE) and variational assimilation. They show that using more observation locations improves the results and that the application of a low-pass filter on the algorithm removes small-scale noise. Furthermore, they analyse the sensitivity of surface waves to errors in the bathymetry reconstruction. Sanders and Katopodes [2] derive a continuous adjoint problem from the conservation form of the shallow water equations

* Corresponding author.

E-mail address: judith.angel@tuhh.de (J. Angel).

<https://doi.org/10.1016/j.compfluid.2024.106321>

Received 8 April 2024; Received in revised form 21 May 2024; Accepted 23 May 2024

Available online 25 May 2024

0045-7930/© 2024 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

with nonlinear bottom friction term and the corresponding sensitivity equations. Cocquet et al. [10] approximate the unsteady SWE with a Helmholtz equation and use a subspace trust-region method for the minimisation. They consider a continuous optimisation problem for the weak form of their steady-state equation, prove continuity of the control-to-state operator, existence of an optimal solution and convergence of the Finite Element approximation. Schäfer Aguilar and Ulbrich [11] prove convergence of discretisation schemes for the adjoint equation of a hyperbolic conservation law with boundary control. Rizzo [12] performs reconstruction using PDE-constrained optimisation but under assumptions for which the SWE reduce to a steady state Helmholtz problem. Losch and Wunsch [13] use a linearised shallow water model to reconstruct a bathymetry with a sensitivity approach. In the dissertation of Wu [14], coastal bathymetries are inferred from surface waves using a high-order spectral method for simulations and a variational data assimilation method for the reconstruction. In Chapter 3 of the dissertation a detailed list of references on bathymetry reconstruction is provided.

There are also a variety of approaches that do not use an adjoint problem. Vasan et al. [15] consider two inverse problems: Bathymetry reconstruction from free-surface deviation and velocity as well as velocity deduction from surface deviation and approximate bathymetry. They use one reconstruction algorithm for both problems at once. Hajduk et al. [16] discretise the SWE with (dis)continuous Galerkin approaches and use a pseudo-time stepping scheme to reconstruct the bathymetry. Gessese and Sellier [17,18] use a one-shot technique, where they replace water depth over ground in the SWE by the surface elevation and solve numerically for the difference. Drähne et al. [19] simulate long wave run-up on a beach with a DG method and validate with experimental data but do not consider the inverse problem. Hao et al. [20] use machine learning based techniques developed for image inpainting to fill in bathymetry data from measurements at the coast or beach. Their approach can provide normalised root means square errors (NRMSE) of around 20%. They also provide a detailed discussion of the relevance of bathymetry reconstruction. Kar and Guha [21] compute the bathymetry by taking time averages of velocities and the free surface elevation using the fact that a hill in the bathymetry creates a dip in the free surface in the case of sub-critical flow. Xie et al. [22] use a neural network to reconstruct the bathymetry from side scan measurements. Liu et al. [23] use a deep-learning based surrogate model for shallow water. They solve the inverse problem with a gradient descent method and automatic differentiation instead of a continuous adjoint.

While there are different techniques and theoretical analyses, so far there seem to be only few studies of how the PDE-constrained optimisation approaches fare for real data, in particular with a continuous adjoint. One notable exception is the work by Lacasta et al. [24, 25], who use a continuous adjoint of the one-dimensional SWE to reconstruct inlet boundary conditions from experimental data in a similar setting as ours, although their wave tank is significantly larger. However, they treat the bathymetry as known and do not attempt to reconstruct it.

As Khan et al. [8] state in their 2021 paper on bathymetry reconstruction, “while the theoretical results from such models are promising, their applicability to real-world measurements has not been established”. To our knowledge, our paper is the first attempt at demonstrating that PDE-constrained optimisation with a continuous adjoint can be used to reconstruct a bathymetry from real-world measurements in an experimental setting. Both the used code [26] and data [27] are available and allow for the reproduction of all results reported in the paper.

2. Problem description

Fig. 1 shows a sketch of our problem setting. The aim is to reconstruct the bathymetry b from observations of the free surface elevation

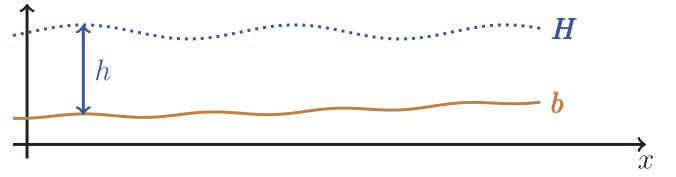


Fig. 1. Free surface elevation H (dotted blue), bathymetry b (brown) and water depth h . Our aim is to reconstruct b from point measurements of H .

$H := h + b$. We will study both simulated data and experimental measurements, generated by inserting a manufactured, roughly Gaussian-shaped bathymetry into a wave flume. A sketch of the experimental setup is shown in Fig. 2. Waves are generated at the left by a back-and-forth motion of a spindle actuated wave flap. Four sensors are positioned throughout the flume at increasing distance from the flap. The bathymetry is positioned between sensor 2 and sensor 3. At the right, a beach is installed to minimise reflections. In all simulations, we use the readings from sensor 1 to define the left boundary condition while readings from sensors 2, 3, 4 are used to reconstruct the bathymetry. A detailed explanation together with the data is available for download [27]. The water depth h and the depth averaged velocity u of a homogeneous, incompressible fluid in the flume can be modelled by the nonlinear shallow water equations (SWE)

$$\begin{pmatrix} h \\ u \end{pmatrix}_t + \begin{pmatrix} hu \\ \frac{1}{2}u^2 + gh \end{pmatrix}_x = \begin{pmatrix} 0 \\ -gb_x - \kappa u \end{pmatrix} \quad (1)$$

with a linear bottom friction term, where we set the corresponding coefficient to $\kappa = 0.2$. We experimented with more complex friction terms as well but found the impact on the accuracy of the model negligible. The gravitational acceleration is set to $g = 9.81 \text{ m/s}^2$. Let $\Omega = [L, R]$ be the spatial domain and $Q := \Omega \times [0, T]$ the space-time domain for some final time T .

Suppose we have an observation H_{obs} of the surface elevation. As we want to model waves in the flume, we need to carefully choose the boundary conditions such that the numerical solution of the SWE coincides with measurements obtained by the sensors. At time $t = 0$ the water is at rest. After a short time, the wave flap, see Fig. 2, starts moving back and forth, generating almost sinusoidal waves. We use the observation H_{obs} at sensor 1 to obtain the left boundary value for h and thus set $L = 1.5 \text{ m}$, corresponding to the distance of sensor 1 from the wave flap. As we use the data from the first sensor to set the left boundary condition, data from sensors 2 to 4 is left to be used in the objective function of the optimisation.

We observed good agreement between measurements and simulation when setting the right boundary condition for the velocity u to zero and defining the right boundary to $R = 15 \text{ m}$ so that the computational domain is longer than the actual water flume. Setting the velocity at the boundary to form a simple wave can be beneficial for matching the experimental data when using finite volume methods [19], but realising this boundary condition in the used simulation framework proved to be unstable. As the water is at rest at the start of the experiment, we set the initial velocity to zero. In summary, the initial and boundary conditions read

$$h(\cdot, 0) = H_{\text{obs}}(\cdot, 0) - b \quad \text{on } \Omega \quad (2a)$$

$$u(\cdot, 0) = 0 \quad \text{on } \Omega \quad (2b)$$

$$h(L, \cdot) = H_{\text{obs}}(L, \cdot) - b(L) \quad \text{on } [0, T] \quad (2c)$$

$$u(R, \cdot) = 0 \quad \text{on } [0, T]. \quad (2d)$$

Note that in forward simulations during the reconstruction process, b is the approximation to the exact bathymetry in the current iteration.

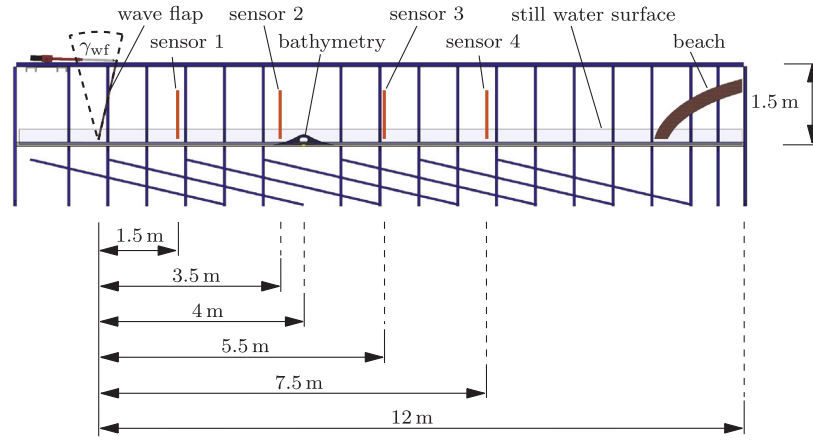


Fig. 2. Sketch of used experimental setup using the wave flume at the Institute of Mechanics and Ocean Engineering at Hamburg University of Technology. The parameter γ_{wf} is the flap angle.

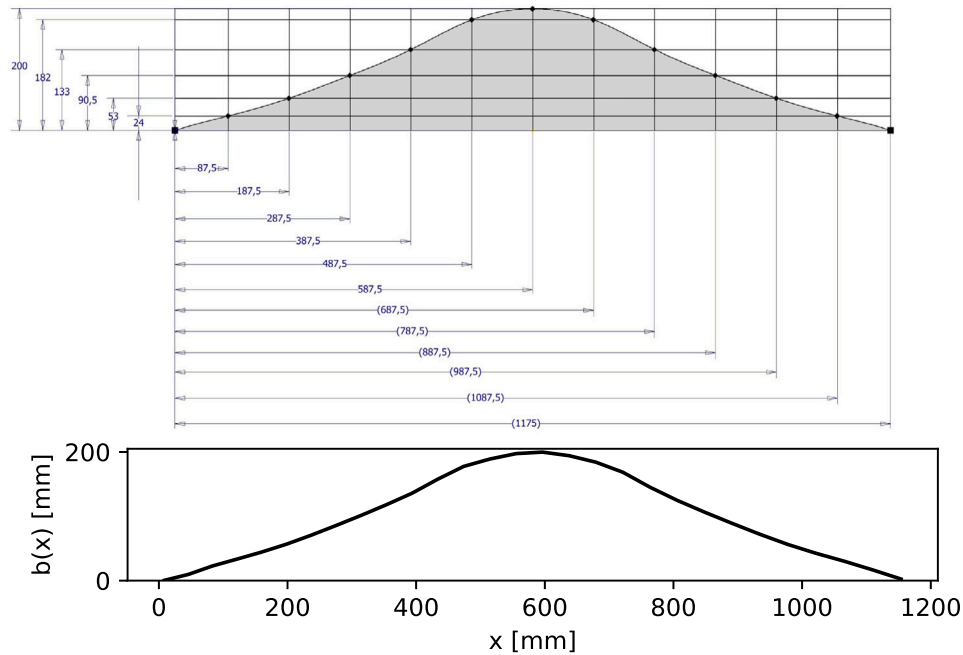


Fig. 3. Measured points (top) and Dedalus representation (bottom) of the bathymetry that was used in the experiment. The Dedalus representation in the lower figure was generated by manually copying the coordinates from the data points in the upper sketch and connecting them using SciPy's CubicSpline function.

2.1. Implementation

The numerical solution of the SWE and its adjoint is computed with the open-source spectral solver framework Dedalus [28] written in Python. We used a Chebyshev basis as ansatz functions and a four-stage Runge Kutta IMEX method of third order included in the Dedalus package. To keep computation times reasonable, we chose a relatively coarse resolution with a grid size of $M = 64$ and time step of 1×10^{-3} s.

The representation of the bathymetry in Dedalus was generated by manually reading the data points in the technical sketch of the actual bathymetry in Fig. 3 (top) and interpolating them using SciPy's CubicSplines function, resulting in the representation shown in Fig. 3 (bottom). Note that the representation of the manufactured bathymetry in Fig. 3 was plotted with a much finer resolution than the one we used for the computations. However, we emphasise that although the two bathymetries look similar, they are not identical. This mismatch between the actual bathymetry and its representation in the code is part of the errors reported below, since the bathymetry reconstructed

by our algorithm is compared against the representation in Dedalus and not the physical bathymetry.

2.2. Validation of the forward model

Fig. 4 shows the simulated surface elevation for the bathymetry sketched in Fig. 3 (bottom) as dotted line, the average of the $n = 20$ measurements with bathymetry as dashed line and the two-sided 95% confidence interval [29] as grey area for three sensors. For the computation of the confidence intervals we use the t -value 1.729 [30] for $n - 1$. Note that because the generated waves change very little between the measurements, the confidence intervals are small and difficult to see in the plot.

Since measurements at sensor 1 provide the left boundary condition, we naturally have perfect agreement at $x = L$. At sensor 2, model and experiment are in very good agreement with both phase and amplitude of the waves being very similar. For sensors 3 and 4, we start to see an underestimation of amplitude in the model and a slightly too slow wave speed. Most likely, this is due to dispersive effects not captured

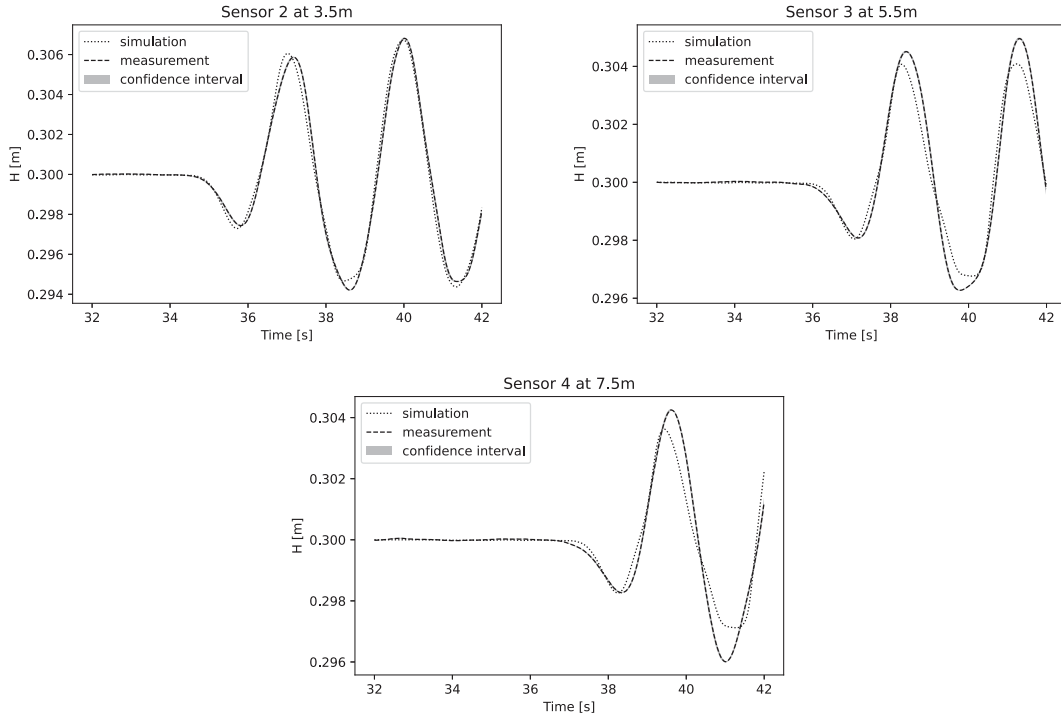


Fig. 4. Measured (dashed) free surface elevation H (dashed) with 95% confidence interval (grey) and numerical simulation (dotted).

by the SWE. While using the Euler- or Navier-Stokes equations to also model changes in density and velocity with depth would likely generate a better match, it would also drastically increase computational cost. Therefore, we retain the relatively cheap SWE as model which, as we will show below, is already sufficient to provide a reasonable reconstruction of the bathymetry. With $M = 64$ and a time step of $\delta t = 1 \times 10^{-3}$ s, the relative ℓ^2 -errors between simulated and measured time series at sensor positions 2, 3, 4 are around 1.9×10^{-3} , 1.6×10^{-3} and 1.3×10^{-3} . Taking a finer spatial resolution with $M = 100$ modes reduces these errors only marginally to 1.2×10^{-3} , 1.2×10^{-3} and 1×10^{-3} . We therefore use $M = 64$ for all reconstructions for the sake of efficiency.

The information that can be used to reconstruct the bathymetry is the difference in the free surface elevation in the presence and absence of bathymetry. Fig. 5 shows the difference between the average of the 20 experiments with and the average of the 20 experiments without bathymetry. Again, the 95% confidence interval is shown in grey. Here we used the t -value 1.686 [30] for $2(n-1)$, because we have a two-sample problem with two different experiments [29]. The maximum differences in the water depth with and without bathymetry are around 1.5 mm at sensor 2 and 3 mm at sensors 3 and 4.

3. Optimisation

Bathymetry reconstruction can be formulated as a minimisation problem with the shallow water Eq. (1) as constraint. We define a functional that measures the error between the simulated free surface elevation and the observation for a given bathymetry b . Let $\gamma, \delta, \lambda_1, \lambda_2 > 0$. We define the objective functional

$$J(b, H) := \frac{\gamma}{2} \int_Q (H - H_{\text{obs}})^2 dx(t) + \frac{\delta}{2} \int_\Omega (H(\cdot, T) - H_{\text{obs}}(\cdot, T))^2 dx + \frac{\lambda_1}{2} \int_\Omega b^2 dx + \frac{\lambda_2}{2} \int_\Omega b_x^2 dx \quad (3)$$

which consists of the L^2 -error in the observation plus regularisation terms. Note that the point in time T has measure zero and therefore does not contribute to the first integral in the objective functional J . This also shows up in the discrete scheme: If the source term of the adjoint is discretised implicitly, the mismatch at time T has no influence

on the adjoint and thus the gradient. To have the flexibility of taking the mismatch at the final time into account we include the second term in the objective functional, leading to a final time condition for the adjoint equation. We regularise to ensure well-posedness of the PDE-constrained optimisation problem

$$\min_{b \in H^1(\Omega)} J(b, H) \text{ subject to (1).} \quad (4)$$

In this context, the SWE are called the forward model. To find a solution of (4), we use gradient descent, and thus solve adjoint equations to compute the gradient. Since the gradient of the bathymetry b_x appears in the SWE, we need to use the H^1 -gradient. We obtain the adjoint equations using the Lagrange function [31] which in our case is defined as

$$\begin{aligned} \mathcal{L}(h, u, b, p) = & J(b, H) - \int_Q (h_t + (hu)_x) p_1 dx(t) \\ & - \int_Q (u_t + uu_x + gh_x + gb_x + \kappa u) p_2 dx(t) \\ & - \int_0^T (h(L, \cdot) - H_{\text{obs}}(L, \cdot) + b(L)) p_3 dt \\ & - \int_\Omega (h(\cdot, 0) - (H_{\text{obs}}(\cdot, 0) - b)) p_4 dx \end{aligned} \quad (5)$$

with Lagrange multipliers $p := (p_1, \dots, p_4)$. Finding the minimum of this function corresponds to solving (4) while partial derivatives of \mathcal{L} yield the continuous adjoint equations and the L^2 -gradient of the objective functional.

3.1. Adjoint problem and gradient

The adjoint equations for the SWE in quasi-linear form can be determined using the optimality condition

$$\begin{aligned} \mathcal{L}_h(h, u, b, p) \varphi &= 0 \quad \forall \varphi \in C_0^\infty(Q) \\ \mathcal{L}_u(h, u, b, p) \varphi &= 0 \quad \forall \varphi \in C_0^\infty(Q). \end{aligned} \quad (6)$$

The adjoint problem reads

$$\begin{aligned} p_{1,t} + up_{1,x} + gp_{2,x} &= -\gamma (h + b - H_{\text{obs}}) \\ p_{2,t} + hp_{1,x} + up_{2,x} &= \kappa p_2 \end{aligned} \quad (7)$$

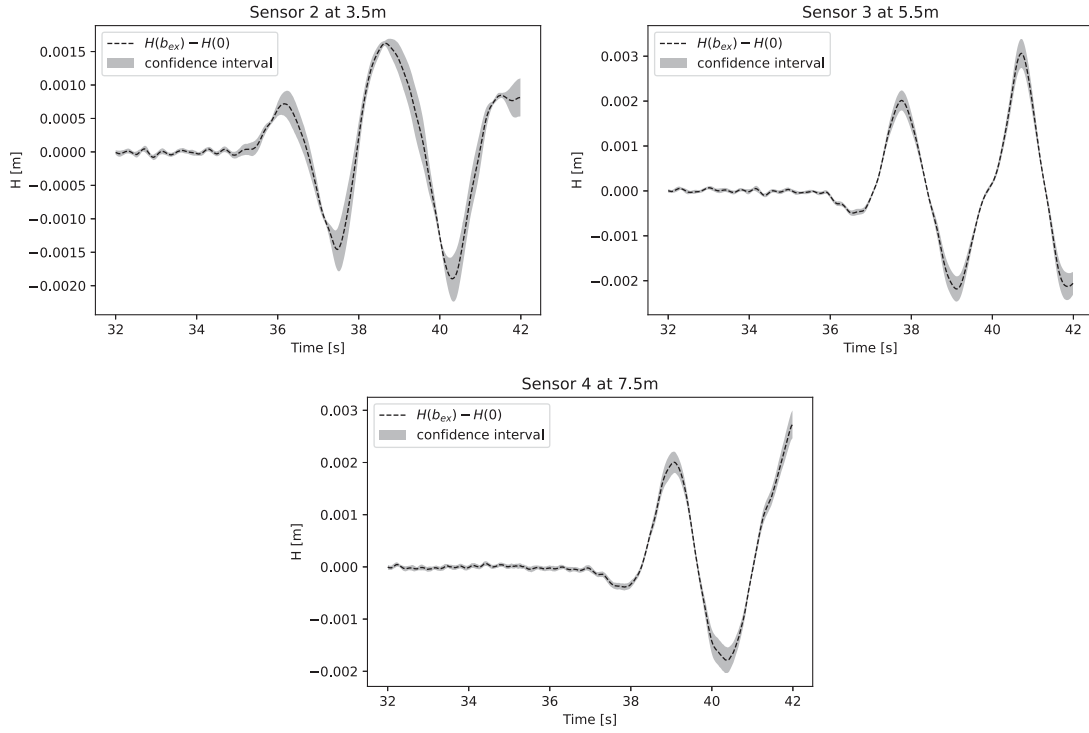


Fig. 5. Difference between the averages of measurements with and without bathymetry (dashed) and 95% confidence interval (grey).

with final and boundary conditions

$$p_1(\cdot, T) = \delta(h(\cdot, T) + b - H_{\text{obs}}(\cdot, T)) \quad (8a)$$

$$p_2(\cdot, T) = 0 \quad (8b)$$

$$p_1(L, \cdot) = -\frac{u(L, \cdot)}{h(L, \cdot)} p_2(L, \cdot) \quad (8c)$$

$$p_2(R, \cdot) = 0, \quad (8d)$$

see Appendix for the detailed derivation. Note that the division in (8c) is meant pointwise and allowed as $h(L, t) > 0 \forall t \in [0, T]$. Let

$$\begin{aligned} \tilde{v} := & \int_0^T \gamma(h + b - H_{\text{obs}}) + gp_{2,x} dt \\ & + \delta(h(\cdot, T) + b - H_{\text{obs}}(\cdot, T)) + \lambda_1 b - \lambda_2 b_{xx} - p_4. \end{aligned} \quad (9)$$

Then the H^1 -gradient is the solution v of the problem

$$\begin{aligned} v - \Delta v &= \tilde{v} \\ v(R) &= 0 \\ v(L) &= 0. \end{aligned} \quad (10)$$

By choosing homogeneous Dirichlet boundary conditions for (10) we assume that the bathymetry does not change at the boundaries. As we initialise the bathymetry with zero throughout all reconstruction experiments in this paper we have $b(L) = 0$. Therefore, in (2c) we always have $h(L, \cdot) = H_{\text{obs}}(L, \cdot)$. A different boundary condition for the gradient, however, could also model non-zero bathymetry at the domain boundary.

3.2. Minimisation of the objective function

We use gradient descent for minimisation as it is easy to implement. Given an observation of the surface elevation H_{obs} , it starts with an initial guess for the bathymetry b . Using an adaptive step size a_j found by Armijo line search [32], the algorithm descends in the direction of the antigradient. In each iteration of gradient descent the forward Eq. (1) and the adjoint problem (7) need to be solved. The results and the current bathymetry b are then used to compute the gradient v .

Our algorithm stops if the ℓ^2 -norm of the gradient is smaller than a user-defined tolerance ϵ .

Spatial derivatives in the objective functional and the gradient are computed with the Dedalus function `Differentiate`. Since Dedalus uses a global spectral basis, derivatives of the approximations are computed analytically in Fourier space. For the computations of the integrals we use Simpson's rule as provided by the SciPy function `simpson`.

4. Reconstruction from simulated observations

In this section, we analyse how well the reconstruction works for data that have been generated by the forward model. This allows us to control noise, eliminates the effect of model error and provides a “best case” to compare reconstruction from experimental data against. We use the Dedalus representation of the exact bathymetry shown in Fig. 3 (bottom). To generate the observation, we compute the solution (h, u) of (1) to (2d) for the exact bathymetry b_{ex} and determine $H_{\text{obs}} = h + b_{\text{ex}}$ on the whole spatial domain. We used $M = 128$ spatial grid points and a time step of 5×10^{-5} s to compute H_{obs} . This is a finer resolution than the one used for forward and adjoint problem in the reconstruction process. Using exactly the same discretisation would ignore the discretisation error in the optimisation and is called an “inverse crime” that should be avoided [33]. In order to deal with the different resolutions, Dedalus's `change_scale` function is used to interpolate between spatial meshes and SciPy's `CubicSpline` function to interpolate in time.

Note that the bathymetries in this section are only shown for $x < 10$ m for better visibility. The oscillations in the interval [10 m, 15 m] are fairly small and thus negligible for the plots.

4.1. Observation on the whole spatial domain

The minimisation is performed as described in Section 3, with a computed observation that is available on the whole spatial domain. We set the coefficients in the objective functional (3) to $\gamma = \delta = 0.5$,

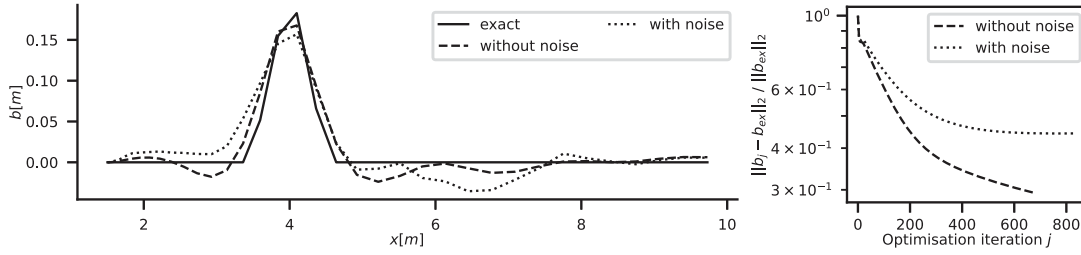


Fig. 6. Reconstructed bathymetries from simulated data with and without noise (left) and relative ℓ^2 -error to exact bathymetry (right).

$\lambda_1 = 10^{-6}$ and $\lambda_2 = 10^{-7}$. For a second reconstruction from noisy data we added normally distributed noise to the computed observation with zero mean and a standard deviation of 5% of the highest observed local wave amplitude. For this case, we increased the regularisation parameter to $\lambda_1 = 10^{-5}$. The corresponding reconstructions and relative errors to the exact bathymetry are shown in Fig. 6. Noise in the observation leads to some additional oscillations in the reconstructed bathymetry. In the case without noise, the algorithm stopped early as there was no step size found for which the update of the bathymetry leads to a smaller value of the objective functional, indicating that the approximated anti-gradient was no longer a descent direction. This is most likely due to the discretisation error which is significantly larger than in the observation. When using an observation on a coarser grid instead, the optimisation algorithm runs until the stopping criterion for gradient descent it satisfied. The reconstruction in Fig. 6 for the observation without noise has a relative ℓ^2 -error of around 29.44%. In order to investigate the sensitivity of the parameters $\gamma, \delta, \lambda_1, \lambda_2$, we did the same reconstruction using 90% and 110% of the chosen values. This slightly changes the relative ℓ^2 -error in the bathymetry to values between 29.08% and 29.89%.

4.2. Observation at sensor positions

In order to get closer to the experimental setup, we additionally reconstructed the bathymetry from observations only at three points in space where there are sensors in the real experiment. Let $x_i, i = 1, \dots, m_p$, be the sensor positions and $H_{obs,i}(t)$ the interpolated corresponding observations. To avoid using Dirac distributions in the objective functional, we replace the point measurements by Gaussians centred at the sensor positions, with maximum value equal to $H_{obs,i}$. The variance of the Gaussians is set to $\sqrt{0.045}$, making it narrow but still somewhat wider than the distance between two adjacent points on the grid. This changes the objective functional (3) to

$$\begin{aligned}
 J(b, H) := & \frac{\gamma}{2} \int_Q \left(\sum_{i=1}^{m_p} (H(x_i, t) - H_{obs,i}(t)) \exp \left(-\frac{1}{2} \frac{(x - x_i)^2}{0.045} \right) \right)^2 dx \\
 & + \frac{\delta}{2} \int_{\Omega} \left(\sum_{i=1}^{m_p} (H(x_i, T) - H_{obs,i}(T)) \exp \left(-\frac{1}{2} \frac{(x - x_i)^2}{0.045} \right) \right)^2 dx \\
 & + \frac{\lambda_1}{2} \int_{\Omega} b^2 dx + \frac{\lambda_2}{2} \int_{\Omega} b_x^2 dx.
 \end{aligned} \tag{11}$$

Fig. 7 shows the reconstruction for this case without noise as well as with noise added to the observation. The parameters are the same as for the previous case with observation on the whole spatial domain. Since the algorithm has a lot less information to work with, the reconstructed bathymetry is less accurate. However, with and without added noise, the maximum of the hill is approximately at the correct position, although the height is underestimated and the hill is somewhat smeared out. There are also more spurious oscillations between sensor 3 and sensor 4.

5. Reconstruction from experimental observations

For the reconstruction from experimental measurements we used the same parameters as for the optimisation from simulated observation at sensor positions with added noise in Section 4. Fig. 8 shows the reconstruction from the measurement data obtained by sensors 2, 3 and 4. The maximum of the reconstructed hill is at the correct position, but second hill at $x \approx 8$ m has appeared. Compared to the reconstruction from the simulated observation with added noise, the unwanted hill on the right is larger, but the height of the hill at 4 m is minimally closer to the exact bathymetry (0.111 m versus 0.105 m). In order to measure the quality of the reconstructions, we compare the ℓ^2 - and ℓ^∞ - errors as well as the normalised root mean square error (NRMSE)

$$\text{NRMSE} = \frac{\sqrt{\frac{1}{M} \sum_{i=1}^M (b_i - b_{ex,i})^2}}{b_{ex, \max} - b_{ex, \min}} \tag{12}$$

with $M = 64$ being the number of points where the representation of the exact bathymetry b_{ex} is defined.

We show relative errors and NRMSEs of different configurations for simulated and experimental observations in Table 1. For simulated observation, where model errors are absent, errors increase when using less data. However, using only sensors 2 and 3 provides a reconstruction that is almost as accurate as using all three sensors, indicating that sensor 4 contributes little useful information. Adding noise increases the ℓ^2 - and ℓ^∞ -error but has little impact on the NRMSE.

For real measurements, ℓ^∞ -errors are comparable to those for simulated observations while ℓ^2 -errors and NRMSE are higher. In contrast to simulated observations, the lowest errors are achieved when using only sensors 2 and 3 for reconstruction. This is likely because of the lack of dispersive effects in the SWE and the resulting mismatch between reality and model at sensor 4. Interestingly, Lacasta et al. [25] also report issues with the shallow water model, although in their case the reason is wave breaking. To put these numbers into context, for their state-of-the-art ML-based approach, Hao et al. report NRMSE between 20.6% and 31.8% [20, Table 2] for real-world test cases. Considering that our setup is simpler, the NRMSE of about 14% for measured observations in Table 1 seems reasonable.

6. Conclusions

This paper analyses the performance of a PDE-constrained optimisation approach with a continuous adjoint for the reconstruction of bathymetry from experimental data. Our setup consists of a roughly Gaussian shaped bathymetry placed in a wave flume with four sensors providing point measurements over time. Data from the first sensor is used to prescribe the boundary condition for the model, leaving measurements from the remaining three sensors for the reconstruction.

To establish a baseline without the influences of modelling errors, we first perform the reconstruction using simulated observations with and without added noise. Then we perform the reconstruction from experimental data using data from all three sensors, two sensors or only a single sensor. Even though the differences in wave heights with and without bathymetry are only a few millimetres, the approach can

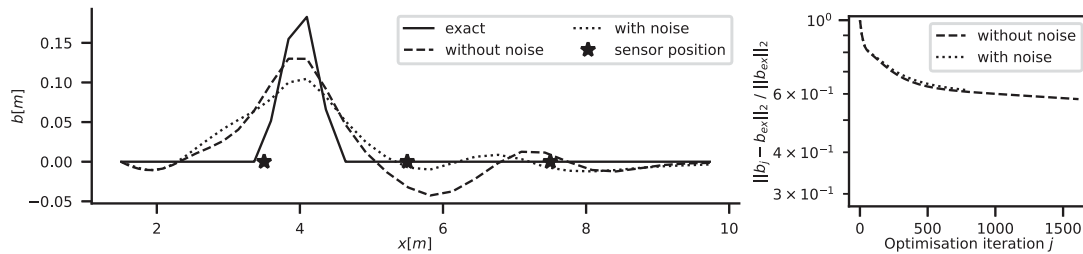


Fig. 7. Reconstructed bathymetries from simulated data at sensor positions (left) and relative ℓ^2 -error to exact bathymetry (right).

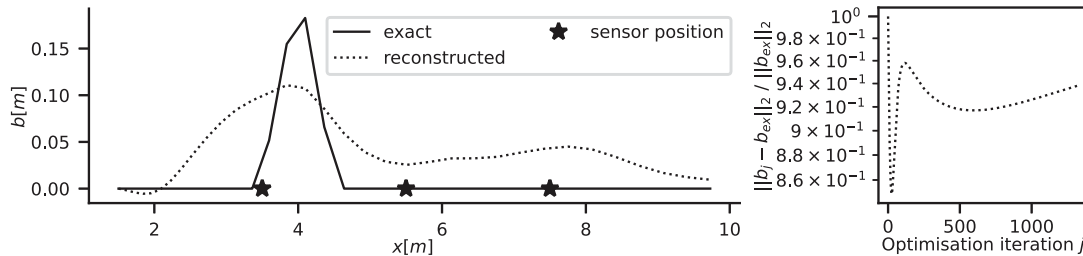


Fig. 8. Reconstructed bathymetry from measurements (left) and relative ℓ^2 -error to exact bathymetry (right).

Table 1

Relative error in the ℓ^2 - and ℓ^∞ -norm and normalised root mean squared error (NRMSE) between reconstructed bathymetry and the model representation of the exact bathymetry for reconstructions using data from different sensors.

Observation		ℓ^2 -error	ℓ^∞ -error	NRMSE
Type	Used sensors			
Simulated	2,3,4	57.84	35.35	10.12
	2,3	59.08	34.91	10.34
	2	65.60	43.33	11.48
	3	89.92	67.83	15.73
Simulated + noise	2,3,4	61.53	42.75	10.76
	2,3	70.42	52.07	12.32
	2	67.77	55.91	11.85
	3	86.77	76.89	15.18
Measured	2,3,4	93.89	51.65	16.42
	2,3	83.76	51.20	14.65
	2	90.43	55.64	15.82
	3	97.82	59.47	17.11

provide a qualitatively correct reconstruction of the bathymetry. The maximum of the bathymetry is placed approximately at the correct position, even though its height is underestimated and the reconstruction is somewhat smeared out compared to the original. Achieved normalised root mean square errors (NRMSE) of around 14% are found to be in line with recent results from the literature for a machine learning based approach. Our results also suggest that the quality of the reconstruction is sensitive to sensor placement. In theory, the optimisation approach could also be modified to also adjust sensor placement but investigating this is left for future work. Other possible directions for future research would be validation against the steady state benchmark problem by Delestre et al. [34, Chapter 3.1.3–3.1.5]. However, this would require a wave flume that can produce constant in- and outflow, which is not possible for the flap-generated waves used here. Our approach could also be extended to the two-dimensional shallow water equations to model variations in the cross section of the flume. This would require the derivation of the 2D adjoint equation as well as measures to mitigate the increase in computational cost but could be done without substantial modifications to the approach.

CRediT authorship contribution statement

Judith Angel: Writing – review & editing, Writing – original draft, Visualization, Software, Methodology, Formal analysis, Conceptualization. **Jörn Behrens:** Writing – review & editing, Supervision, Project administration, Methodology. **Sebastian Götschel:** Writing – review & editing, Supervision, Methodology, Conceptualization. **Marten Hollm:** Writing – review & editing, Investigation, Data curation. **Daniel Ruprecht:** Writing – review & editing, Writing – original draft, Supervision, Project administration, Methodology, Conceptualization. **Robert Seifried:** Supervision, Resources.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Judith Angel reports financial support was provided by European Union. Daniel Ruprecht reports financial support was provided by European Union. Daniel Ruprecht reports financial support was provided by Federal Ministry of Education and Research Bonn Office. Daniel Ruprecht reports financial support was provided by German Research Foundation. Judith Angel reports financial support was provided by Federal Ministry of Education and Research Bonn Office. Judith Angel reports financial support was provided by German Research Foundation. Marten Hollm reports financial support was provided by German Research Foundation. Joern Behrens reports financial support was provided by German Research Foundation. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The measurement data and detailed description of the experiment can be accessed via <https://doi.org/10.15480/882.9403>.

Acknowledgements

This project has received funding from the European High-Performance Computing Joint Undertaking (JU) under grant agreement No 955701. The JU receives support from the European Union's Horizon 2020 research and innovation programme and Belgium, France, Germany, and Switzerland. This project also received funding from the German Federal Ministry of Education and Research (BMBF) grant 16HPC048. We acknowledge the support by the Deutsche Forschungsgemeinschaft (DFG) within the Research Training Group GRK 2583 "Modeling, Simulation and Optimization of Fluid Dynamic Applications". This project was also supported by the Deutsche Forschungsgemeinschaft (DFG) via grant number 528383251.

Our special thanks go to Riza Demir and Wolfgang Brennecke of the Institute of Mechanics and Ocean Engineering at TUHH, who built and measured the bathymetry. We also thank Vanessa Trapp from the Chair for Stochastics at TU Hamburg for providing us with references regarding the computation of confidence intervals.

Appendix. Derivation of adjoint equations and gradient

Continuous adjoints to the nonlinear SWE have been derived before [2,24,25], but for the conservative form of the SWE and a different bottom friction term. Here we derive adjoint equations for the quasi-linear form. The Lagrangian for our problem reads

$$\begin{aligned} \mathcal{L}(h, u, b, p) = & \frac{\gamma}{2} \int_{\Omega} (h + b - H_{\text{obs}})^2 d(x, t) \\ & + \frac{\delta}{2} \int_{\Omega} (h(\cdot, T) + b - H_{\text{obs}}(\cdot, T))^2 dx \\ & + \frac{\lambda_1}{2} \int_{\Omega} b^2 dx + \frac{\lambda_2}{2} \int_{\Omega} b_x^2 dx \\ & - \int_{\Omega} (h_t + (hu)_x) p_1 d(x, t) \\ & - \int_{\Omega} (u_t + uu_x + gh_x + gb_x + \kappa u) p_2 d(x, t) \\ & - \int_0^T (h(L, \cdot) - H_{\text{obs}}(L, \cdot) + b(L)) p_3 dt \\ & - \int_{\Omega} (h(\cdot, 0) - (H_{\text{obs}}(\cdot, 0) - b)) p_4 dx. \end{aligned} \quad (\text{A.1})$$

with Lagrange multipliers p_1, \dots, p_4 . We use the necessary optimality condition

$$\nabla_{(h,u)} \mathcal{L}(h, u, b, p) = 0 \quad (\text{A.2})$$

to determine the continuous adjoint. Using integration by parts we get

$$\begin{aligned} \mathcal{L}_h(h, u, b, p) \varphi = & \gamma \int_{\Omega} (h + b - H_{\text{obs}}) \varphi d(x, t) \\ & + \delta \int_{\Omega} (h(\cdot, T) + b - H_{\text{obs}}(\cdot, T)) \varphi(\cdot, T) dx \\ & - \int_{\Omega} \varphi_1 p_1 d(x, t) - \int_{\Omega} (\varphi u)_x p_1 d(x, t) - \int_{\Omega} g \varphi_x p_2 d(x, t) \\ & - \int_0^T \varphi(L, \cdot) p_3 dt - \int_{\Omega} \varphi(\cdot, 0) p_4 dx \\ = & \gamma \int_{\Omega} (h + b - H_{\text{obs}}) \varphi d(x, t) \\ & + \delta \int_{\Omega} (h(\cdot, T) + b - H_{\text{obs}}(\cdot, T)) \varphi(\cdot, T) dx \\ & + \int_{\Omega} \varphi p_{1,t} d(x, t) - \int_{\Omega} [\varphi p_1]_0^T dx + \int_{\Omega} \varphi u p_{1,x} d(x, t) \\ & - \int_0^T [\varphi u p_1]_L^R dt + \int_{\Omega} g \varphi p_{2,x} d(x, t) - \int_0^T [g \varphi p_2]_L^R dt \\ & - \int_0^T \varphi(L, \cdot) p_3 dt - \int_{\Omega} \varphi(\cdot, 0) p_4 dx, \end{aligned} \quad (\text{A.3})$$

which has to be zero for all $\varphi \in C_0^\infty(Q)$ due to the optimality condition (6). We obtain

$$\int_{\Omega} (\gamma (h + b - H_{\text{obs}}) + p_{1,t} + u p_{1,x} + g p_{2,x}) \varphi d(x, t) = 0$$

and, using the fundamental lemma of calculus of variations, we get the first part of the adjoint problem

$$p_{1,t} + u p_{1,x} + g p_{2,x} = -\gamma (h + b - H_{\text{obs}}). \quad (\text{A.4})$$

Now we let single conditions to the function φ vary in order to determine terminal and boundary conditions for the adjoint equation. By allowing any value for $\varphi(\cdot, T)$ in Eq. (A.3) we get the final time condition for the first adjoint variable

$$p_1(\cdot, T) = \delta (h(\cdot, T) + b - H_{\text{obs}}(\cdot, T)). \quad (\text{A.5})$$

Without the condition $\varphi(\cdot, 0) = 0$ we obtain

$$p_1(\cdot, 0) = p_4 \quad (\text{A.6})$$

and letting $\varphi(R, \cdot)$ vary we get

$$g p_2(R, \cdot) = 0. \quad (\text{A.7})$$

Dropping the condition $\varphi(L, \cdot) = 0$ yields

$$p_3 = u(L, \cdot) p_1(L, \cdot) + g p_2(L, \cdot) \quad (\text{A.8})$$

for the Lagrange multiplier p_3 . The next partial derivative is

$$\begin{aligned} \mathcal{L}_u(h, u, b, p) \varphi = & - \int_{\Omega} (h \varphi)_x p_1 d(x, t) - \int_{\Omega} (\varphi_t + (u \varphi)_x + \kappa \varphi) p_2 d(x, t) \\ = & \int_{\Omega} h \varphi p_{1,x} d(x, t) - \int_0^T [h \varphi p_1]_L^R dt + \int_{\Omega} \varphi p_{2,t} d(x, t) \\ & - \int_{\Omega} [\varphi p_2]_0^T dx + \int_{\Omega} u \varphi p_{2,x} d(x, t) \\ & - \int_0^T [u \varphi p_2]_L^R dt - \int_{\Omega} \kappa \varphi p_2 d(x, t). \end{aligned} \quad (\text{A.9})$$

Hence, the second part of the adjoint equation is

$$p_{2,t} + h p_{1,x} + u p_{2,x} = \kappa p_2. \quad (\text{A.10})$$

We let $\varphi(\cdot, T)$ vary and obtain the terminal condition for the second adjoint variable

$$p_2(\cdot, T) = 0. \quad (\text{A.11})$$

If we leave out the condition $\varphi(L, \cdot) = 0$, we get

$$\begin{aligned} u(L, \cdot) p_2(L, \cdot) &= -p_1(L, \cdot) h(L, \cdot) \\ \Leftrightarrow p_1(L, \cdot) &= -\frac{u(L, \cdot)}{h(L, \cdot)} p_2(L, \cdot). \end{aligned} \quad (\text{A.12})$$

Together with (A.8) this implies that

$$p_3 = \left(g - \frac{(u(L, \cdot))^2}{h(L, \cdot)} \right) p_2(L, \cdot). \quad (\text{A.13})$$

To rewrite the adjoint problem as an initial value problem we set $\tau := T - t$ and define

$$\begin{aligned} \tilde{p}(x, \tau) &:= p(x, T - \tau) \\ \tilde{h}(x, \tau) &:= h(x, T - \tau) \\ \tilde{u}(x, \tau) &:= u(x, T - \tau). \end{aligned}$$

The time transformed adjoint problem reads

$$\begin{aligned}
 \tilde{p}_{1,\tau} - \tilde{u}\tilde{p}_{1,x} - g\tilde{p}_{2,x} &= \gamma (\tilde{h} + b - \tilde{H}_{\text{obs}}) \\
 \tilde{p}_{2,\tau} - \tilde{h}\tilde{p}_{1,x} - \tilde{u}\tilde{p}_{2,x} &= -\kappa\tilde{p}_2 \\
 \tilde{p}_1(\cdot, 0) &= \delta (\tilde{h}(\cdot, 0) + b - \tilde{H}_{\text{obs}}(\cdot, 0)) \\
 \tilde{p}_2(\cdot, 0) &= 0 \\
 \tilde{p}_1(L, \cdot) &= -\frac{\tilde{u}(L, \cdot)}{\tilde{h}(L, \cdot)}\tilde{p}_2(L, \cdot) \\
 \tilde{p}_2(R, \cdot) &= 0.
 \end{aligned} \tag{A.14}$$

The derivative with respect to b is

$$\begin{aligned}
 \mathcal{L}_b(h, u, b, p)\delta b &= \gamma \int_{\Omega} \int_0^T (h + b - H_{\text{obs}}) dt \delta b dx \\
 &+ \delta \int_{\Omega} (h(\cdot, T) + b - H_{\text{obs}}(\cdot, T)) \delta b dx \\
 &+ \lambda_1 \int_{\Omega} b \delta b dx + \lambda_2 \int_{\Omega} b_x (\delta b)_x dx - \int_{\Omega} \int_0^T g p_2 dt (\delta b)_x dx \\
 &- \int_0^T \delta b(L) p_3 dt - \int_{\Omega} p_4 \delta b dx
 \end{aligned} \tag{A.15}$$

Now we use integration by parts. As we demand no change of the bathymetry at the boundaries, we have $\delta b(L) = 0$ and obtain

$$\begin{aligned}
 \mathcal{L}_b(h, u, b, p)\delta b &= \int_{\Omega} \left(\int_0^T \gamma (h + b - H_{\text{obs}}) + g p_{2,x} dt \right) \delta b dx \\
 &+ \int_{\Omega} \left(\delta (h(\cdot, T) + b - H_{\text{obs}}(\cdot, T)) + \lambda_1 b - \lambda_2 b_{xx} - p_4 \right) \delta b dx.
 \end{aligned} \tag{A.16}$$

Thus, the L^2 -gradient is

$$\begin{aligned}
 \tilde{v} &:= \int_0^T \gamma (h + b - H_{\text{obs}}) + g p_{2,x} dt \\
 &+ \delta (h(\cdot, T) + b - H_{\text{obs}}(\cdot, T)) + \lambda_1 b - \lambda_2 b_{xx} - p_4
 \end{aligned} \tag{A.17}$$

References

- [1] Rabinovich A. Twenty-seven years of progress in the science of meteorological tsunamis following the 1992 Daytona Beach event. *Pure Appl Geophys* 2020;177:1193–230. <http://dx.doi.org/10.1007/s00024-019-02349-3>.
- [2] Sanders B, Katopodes N. Adjoint sensitivity analysis for shallow-water wave control. *J Eng Mech-ASCE - J ENG MECH-ASCE* 2000;126. [http://dx.doi.org/10.1061/\(ASCE\)0733-9399\(2000\)126:9\(909\)](http://dx.doi.org/10.1061/(ASCE)0733-9399(2000)126:9(909)).
- [3] Smith WHF, Sandwell DT. Conventional bathymetry, bathymetry from space, and geodetic altimetry. *Oceanography* 2004;17(1):8–23. <http://dx.doi.org/10.5670/oceanog.2004.63>.
- [4] Sagawa T, Yamashita Y, Okumura T, Yamanokuchi T. Satellite derived bathymetry using machine learning and multi-temporal satellite images. *Remote Sens* 2019;11(10):1155. <http://dx.doi.org/10.3390/rs11101155>.
- [5] Irish JL, Lillycrop WJ. Scanning laser mapping of the coastal zone: The SHOALS system. *ISPRS J Photogrammetry Remote Sens* 1999;54(2–3):123–9. [http://dx.doi.org/10.1016/S0924-2716\(99\)00003-9](http://dx.doi.org/10.1016/S0924-2716(99)00003-9), URL <https://www.sciencedirect.com/science/article/pii/S0924271699000039>.
- [6] Ashphaq M, Srivastava PK, Mitra D. Review of near-shore satellite derived bathymetry: Classification and account of five decades of coastal bathymetry research. *J Ocean Eng Sci* 2021;6(4):340–59. <http://dx.doi.org/10.1016/j.joes.2021.02.006>.
- [7] Evagorou E, Argyriou A, Papadopoulos N, Mettas C, Alexandrakis G, Hadjimitsis D. Evaluation of satellite-derived bathymetry from high and medium-resolution sensors using empirical methods. *Remote Sens* 2022;14(3):772. <http://dx.doi.org/10.3390/rs14030772>, URL <https://www.mdpi.com/2072-4292/14/3/772>.
- [8] Khan RA, Kevlahan NK-R. Variational assimilation of surface wave data for bathymetry reconstruction. Part I: Algorithm and test cases. *Tellus A: Dyn Meteorol Oceanogr* 2021. <http://dx.doi.org/10.1080/16000870.2021.1976907>.
- [9] Khan RA, Kevlahan NK-R. Variational Assimilation of Surface Wave Data for Bathymetry Reconstruction. Part II: Second Order Adjoint Sensitivity Analysis. *Tellus A: Dyn Meteorol Oceanogr* 2022. <http://dx.doi.org/10.16993/tellusa.36>.
- [10] Cocquet P-H, Rizzo S, Salomon J. Optimization of bathymetry for long waves with small amplitude. *SIAM J Control. Optim* 2021;59(6):4429–56. <http://dx.doi.org/10.1137/20M1326337>.
- [11] Schäfer Aguilar P, Ulbrich S. Convergence of numerical adjoint schemes arising from optimal boundary control problems of hyperbolic conservation laws. 2021, URL <https://opus4.kobv.de/opus4-trr154/frontdoor/index/index/year/2021/docId/481>.
- [12] Reyes Rizzo S. Mathematical methods for marine energy extraction, École Doctorale de Dauphine [Ph.D. thesis], 2021.
- [13] Losch M, Wunsch C. Bottom topography as a control variable in an ocean model. *J Atmos Ocean Technol* 2003;20(11):1685–96.
- [14] Wu J. Adjoint-based data assimilation for ocean waves [Ph.D. thesis], University of Minnesota; 2022.
- [15] Vasan V, Manisha, Auroux D. Ocean-depth measurement using shallow-water wave models. *Stud Appl Math* 2021;147(4):1481–518. <http://dx.doi.org/10.1111/sapm.12418>.
- [16] Hajduk H, Kuzmin D, Aizinger V. Bathymetry reconstruction using inverse ShallowWater models: Finite element discretization and regularization. In: *Lecture notes in computational science and engineering*, Springer International Publishing; 2020, p. 223–30. http://dx.doi.org/10.1007/978-3-030-30705-9_20.
- [17] Gessese AF. Algorithms for bed topography reconstruction in geophysical flows [Ph.D. thesis], University of Canterbury; 2013.
- [18] Gessese A, Sellier M. A Direct Solution Approach to the Inverse Shallow-Water Problem. *Math Probl Eng* 2012;2012:1–18. <http://dx.doi.org/10.1155/2012/417950>.
- [19] Drähne U, Goseberg N, Vater S, Beisiegel N, Behrens J. An experimental and numerical study of long wave run-up on a plane beach. *J Mar Sci Eng* 2016;4(1). <http://dx.doi.org/10.3390/jmse4010001>, URL <https://www.mdpi.com/2077-1312/4/1/1>.
- [20] Hao Z, Chen F, Jia X, Cai X, Yang C, Du Y, et al. GRDL: A New Global reservoir area-storage-depth data set derived through deep learning-based bathymetry reconstruction. *Water Resour Res* 2024;60(1):e2023WR035781. <http://dx.doi.org/10.1029/2023WR035781>.
- [21] Kar S, Guha A. Letter: Ocean bathymetry reconstruction from surface data using hydraulics theory. *Phys Fluids* 2018;30(12):121701. <http://dx.doi.org/10.1063/1.5055944>.
- [22] Xie Y, Bore N, Folkesson J. Neural network normal estimation and bathymetry reconstruction from sidescan sonar. *IEEE J Ocean Eng* 2023;48(1):218–32. <http://dx.doi.org/10.1109/JOE.2022.3194899>.
- [23] Liu X, Song Y, Shen C. Bathymetry inversion using a deep-learning-based surrogate for shallow water equations solvers. 2022, [arXiv:2203.02821](https://arxiv.org/abs/2203.02821). URL <https://arxiv.org/abs/2203.02821>.
- [24] Lacasta A, Morales-Hernández M, Brufau P, García-Navarro P. Application of an adjoint-based optimization procedure for the optimal control of internal boundary conditions in the shallow water equations. *J Hydraul Res* 2018;56(1):111–23. <http://dx.doi.org/10.1080/00221686.2017.1300196>.
- [25] Lacasta A, Caviedes-Voullème D, García-Navarro P. Application of the adjoint method for the reconstruction of the boundary condition in unsteady shallow water flow simulation. *Springer International Publishing*; 2019, p. 157–72. http://dx.doi.org/10.1007/978-3-319-89988-6_10.
- [26] Angel J, Ruprecht D, Götschel S. Waterchannel. 2024, <http://dx.doi.org/10.5281/zenodo.11085597>.
- [27] Angel J, Behrens J, Götschel S, Hollm M, Ruprecht D, Seifried R. Data artefact: Bathymetry reconstruction from experimental data with PDE-constrained optimisation. 2024, <http://dx.doi.org/10.15480/882.9403>.
- [28] Burns KJ, Vasil GM, Oishi JS, Lecoanet D, Brown BP. Dedalus: A flexible framework for numerical simulations with spectral methods. *Phys Rev Res* 2020;2:023068. <http://dx.doi.org/10.1103/PhysRevResearch.2.023068>, URL <https://link.aps.org/doi/10.1103/PhysRevResearch.2.023068>.
- [29] Dümbgen L. Einführung in die statistik. Springer; 2016.
- [30] Tabelle der t-Verteilung. 2017, URL https://www.statistik.tu-dortmund.de/fileadmin/user_upload/Lehrstuehle/Oekonometrie/Lehre/WiSoOekoSS17/tabelleT.vf. [Accessed 26 March 2024].
- [31] Tröltzsch F. Optimal control of partial differential equations: Theory, methods, and applications, vol. 112, American Mathematical Soc.; 2010.
- [32] Armijo L. Minimization of functions having Lipschitz continuous first partial derivatives. *Pacific J Math* 1966;16(1):1–3.
- [33] Kaipio J, Somersalo E. Statistical inverse problems: Discretization, model reduction and inverse crimes. *J Comput Appl Math* 2007;198(2):493–504. <http://dx.doi.org/10.1016/j.cam.2005.09.027>, URL <https://www.sciencedirect.com/science/article/pii/S0377042705007296>, special Issue: Applied Computational Inverse Problems.
- [34] Delestre O, Lucas C, Ksinant P-A, Darboux F, Laguerre C, Vo T-N-T, et al. SWASHES: A compilation of shallow water analytic solutions for hydraulic and environmental studies. *Internat J Numer Methods Fluids* 2013;72(3):269–300. <http://dx.doi.org/10.1002/fld.3741>, [arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/fld.3741](https://onlinelibrary.wiley.com/doi/pdf/10.1002/fld.3741). URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/fld.3741>.