Multi-level Monte Carlo finite volume methods for shallow water equations with uncertain topography in multi-dimensions

S. Mishra, Ch. Schwab and J. Šukys

MULTI-LEVEL MONTE CARLO FINITE VOLUME METHODS FOR SHALLOW WATER EQUATIONS WITH UNCERTAIN TOPOGRAPHY IN MULTI-DIMENSIONS

S. MISHRA, CH. SCHWAB, AND J. ŠUKYS

ABSTRACT. The initial data and bottom topography, used as inputs in shallow water models, are prone to uncertainty due to measurement errors. We model this uncertainty statistically in terms of random shallow water equations. We extend the Multi-Level Monte Carlo (MLMC) algorithm to numerically approximate the random shallow water equations efficiently. The MLMC algorithm is suitably modified to deal with uncertain (and possibly uncorrelated) data on each node of the underlying topography grid by the use of a hierarchical topography representation. Numerical experiments in one and two space dimensions are presented to demonstrate the efficiency of the MLMC algorithm.

1. INTRODUCTION

1.1. **The model.** Many interesting flows, for instance, in lakes, rivers, irrigation channels, avalanches, landslides and tsunamis in oceans, have a common property that the vertical scale (depth) of the flow is much smaller than the horizontal scales of motion. Hence, the full three-dimensional incompressible Navier-Stokes equations of fluid dynamics can be simplified to the so-called shallow water equations [27]:

$$
(1.1) \quad
\begin{cases}
h_t + (hu)_x + (hv)_y = 0, \\
(hu)_t + \left(hu^2 + \dfrac{1}{2}gh^2\right)_x + (huv)_y = -ghb_x, \\
(hv)_t + (huv)_x + \left(hv^2 + \dfrac{1}{2}gh^2\right)_y = -ghb_y.
\end{cases}
$$

Here, h is the height of the fluid column above the bottom topography $b = b(x, y)$ over which the fluid flows and (u, v) is the vertically averaged horizontal fluid velocity field. The constant g denotes the acceleration due to gravity.

Denoting conserved variables as $\mathbf{U} = \mathbf{U}(\mathbf{x}, t) : \mathbf{D} \times \mathbb{R}_+ \subset \mathbb{R}^2 \times \mathbb{R}_+ \to \mathbb{R}^3$, directional fluxes as $\mathbf{F}, \mathbf{G} : \mathbb{R}^3 \to \mathbb{R}^3$ and sources as $\mathbf{S}$,

$$\mathbf{U} = \begin{bmatrix} h \\ hu \\ hv \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} hu \\ hu^2 + \frac{1}{2}gh^2 \\ huv \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} hv \\ huv \\ hv^2 + \frac{1}{2}gh^2 \end{bmatrix}, \quad \mathbf{S} = \begin{bmatrix} 0 \\ -ghb_x \\ -ghb_y \end{bmatrix},$$

the system (1.1) with given initial data $\mathbf{U}_0$ is written as the *system of balance laws*,

$$(1.2) \qquad \begin{cases} \mathbf{U}(\mathbf{x}, t)_t + \mathbf{F}(\mathbf{U})_x + \mathbf{G}(\mathbf{U})_y = \mathbf{S}(\mathbf{x}, \mathbf{U}), \\ \qquad\qquad\quad \mathbf{U}(\mathbf{x}, 0) = \mathbf{U}_0(\mathbf{x}). \end{cases} \quad \mathbf{x} = (x, y) \in \mathbf{D}, \quad t > 0.$$

It is well known that solutions to (1.2) can develop shock discontinuities in finite time even for smooth initial data. Hence, the solutions of system of balance laws (1.2) are considered in the weak (distributional) sense and are well-defined provided the source term $\mathbf{S}$ remains uniformly bounded [7], i.e. $b \in W^{1,\infty}(\mathbb{R}^2)$. In the cases where the topography function b is discontinuous, one can use the theory of non-conservative products of [8] to define weak solutions.

Weak solutions of (1.2) need not be unique. Additional admissibility criterion called *entropy conditions* are imposed. For shallow water equations, the total energy

$$E(\mathbf{U}) = \frac{1}{2}(hu^2 + hv^2 + gh^2 + ghb)$$

plays the role of the entropy function. The admissible weak solutions of the shallow water equations satisfy the entropy inequality (in the sense of distributions),
(1.3)
$$E(\mathbf{U})_t + \left(\frac{1}{2} \left(hu^3 + huv^2 \right) + ghu(h + b) \right)_x + \left(\frac{1}{2} \left(hu^2v + hv^3 \right) + ghv(h + b) \right)_y \leq 0.$$

Furthermore, shallow water flows are often realized as perturbations of some stationary solutions. As an example, tsunami waves in an ocean [17] are perturbations of the lake (ocean) at rest steady state:

$$(1.4) \qquad\qquad u \equiv 0, \quad v \equiv 0, \quad h + b \equiv \text{constant}.$$

1.2. **Numerical schemes.** In last few decades, finite volume methods (FVM) [16] have emerged as a very popular framework for approximating systems of balance (conservation) laws. Let $\mathcal{T} = \mathcal{T}^1 \times \cdots \times \mathcal{T}^d$ denote a uniform axiparallel quadrilateral mesh of the spatial Cartesian domain of interest $\mathbf{D} = I_1 \times \cdots \times I_d \subset \mathbb{R}^d, I_r \subset \mathbb{R}, d = 1, 2$. The mesh $\mathcal{T}$ will be understood as a partition of the physical domain $\mathbf{D}$ into a finite set of equal disjoint open quadrilaterals (cells), i.e. in two dimensions,

$$C_{i,j} := C_i \times C_j \subset I_1 \times I_2 \subset \mathbb{R}^2, \quad i = 1, \dots, \#\mathcal{T}^1, \ j = 1, \dots, \#\mathcal{T}^2.$$

Let $\Delta x := \frac{|I_1|}{\#\mathcal{T}_1}$ and $\Delta y := \frac{|I_2|}{\#\mathcal{T}_2}$ denote the *mesh widths* in x and y directions. Define the approximations to cell averages of the solution $\mathbf{U}$ by

$$\mathbf{U}_{i,j}(t) \approx \frac{1}{\Delta x \Delta y} \int_{C_{i,j}} \mathbf{U}(\mathbf{x}, t) d\mathbf{x}.$$

Then, a semi-discrete finite volume scheme [16] for approximating (1.2) is given by

$$(1.5) \qquad \frac{\partial}{\partial t} \mathbf{U}_{i,j}(t) = -\frac{1}{\Delta x}(\mathbf{F}_{i+\frac{1}{2},j} - \mathbf{F}_{i-\frac{1}{2},j}) - \frac{1}{\Delta y}(\mathbf{G}_{i,j+\frac{1}{2}} - \mathbf{G}_{i,j-\frac{1}{2}}) - \mathbf{S}_{i,j}.$$

where $\mathbf{F}_{\cdot,j}, \mathbf{G}_{i,\cdot}$ are numerical fluxes and $\mathbf{S}_{i,j}$ is the cell average of the source $\mathbf{S}$.

The *numerical fluxes* $\mathbf{F}_{i+\frac{1}{2},j} = \mathbf{F}(\mathbf{U}_{i,j}, \mathbf{U}_{i+1,j})$ and $\mathbf{G}_{i,j+\frac{1}{2}} = \mathbf{G}(\mathbf{U}_{i,j}, \mathbf{U}_{i,j+1})$ are defined by using (approximate) solutions of local Riemann problems (in the normal direction) at each cell interface. High order spatial accuracy is achieved by using non-oscillatory TVD, ENO, WENO and DG procedures. Time integration can be performed using strong stability preserving Runge-Kutta methods.

Although a wide variety of schemes for the shallow water equations have been proposed and are routinely used in realistic computations, we require that a robust scheme should be *entropy stable*, i.e. satisfy a discrete version of the entropy inequality (1.3), and should also be *well-balanced*, i.e. preserve a discrete version of the lake at rest steady state (1.4). Such a scheme was proposed in a recent paper [14] and will be used by us throughout this paper.

1.3. **Uncertainty quantification.** The initial data $\mathbf{U}_0$, bottom topography function b and boundary conditions serve as inputs to any numerical scheme that approximates shallow water equations (1.1). However, these inputs need to be measured and the measurements are prone to uncertainty. The modeling of uncertainty in inputs has been studied extensively in recent years. Statistical (stochastic) models of uncertainty are widely employed. A detailed description of how uncertainty in the bottom topography is modeled statistically can be found in [13, 5] and other references therein. Here, we denote a complete probability space by $(\Omega, \mathcal{F}, \mathbb{P})$ and model the initial data $\mathbf{U}_0$ and the source $\mathbf{S}$ as $(L^1(\mathbb{R}^d)^m, \mathcal{B}(L^1(\mathbb{R}^d)^m))$-measurable random fields

$$\mathbf{U}_0 : \Omega \ni \omega \mapsto \mathbf{U}_0(\mathbf{x}, \omega), \quad \mathbf{S} : \Omega \ni \omega \mapsto \mathbf{S}(\mathbf{x}, \omega).$$

Uncertainty in inputs like the initial data and the bottom topography leads to uncertainty in the solution $\mathbf{U}$ of the shallow water equations. We model this output uncertainty by realizing the entropy solution as a random field, i.e. a measurable mapping $\omega \mapsto \mathbf{U}(\mathbf{x}, t, \omega)$, where

$$\mathbf{U} : (\Omega, \mathcal{F}) \to (C_b([0, T], L^1(\mathbb{R}^d)^m), \ \mathcal{B}(C_b([0, T], L^1(\mathbb{R}^d)^m)),$$

that is a weak solution of the *random shallow water equations*,

$$(1.6) \quad \begin{cases} \mathbf{U}(\mathbf{x}, t, \omega)_t + \mathbf{F}(\mathbf{U})_x + \mathbf{G}(\mathbf{U})_y = \mathbf{S}(\mathbf{x}, \mathbf{U}, \omega), \\ \qquad\qquad \mathbf{U}(\mathbf{x}, 0, \omega) = \mathbf{U}_0(\mathbf{x}, \omega), \end{cases} \quad \mathbf{x} \in \mathbf{D}, \ t > 0, \ \forall \omega \in \Omega,$$

and also satisfies the entropy inequality (1.3) (in the sense of distributions) for $\mathbb{P}$-a.e. $\omega \in \Omega$.

1.4. **Scope and outline of the paper.** The main aim to this paper is to present *efficient* numerical methods to approximate random shallow water equations (1.6).

The design of efficient numerical schemes for quantifying uncertainty in solutions of conservation (balance) laws has seen a lot of activity in recent years. Methods include the stochastic Galerkin methods based on generalized Polynomial Chaos (gPC) [3, 6, 18, 25, 22, 26] and stochastic collocation method [29, 19, 28]. Some of these methods (particularly stochastic Galerkin) have the huge disadvantage of being highly *intrusive*: existing codes for computing deterministic solutions of conservation laws need to be completely reconfigured for implementation. Furthermore, none of these methods are currently able to handle even a moderate number of sources of uncertainty (stochastic dimensions).

Another class of methods are the so-called *Monte Carlo* (MC) methods in which the probability space is *sampled*, the underlying deterministic PDE is solved for each

sample and the samples are combined to ascertain statistical information about the random field. Although non-intrusive, easy to code and to parallelize, MC methods converge at rate $1/2$ as the number M of MC samples increases. The asymptotic convergence rate $M^{-1/2}$ is non-improvable by the central limit theorem. Therefore, MC methods require a large number of "samples" (with each "sample" involving the numerical solution of (1.6) with a given draw of initial data $\mathbf{U}_0$) in order to ensure low statistical errors. This slow convergence entails high computational costs for MC type methods and makes them infeasible for computing uncertainty in complex shallow water flows. We refer to [20] for a detailed error and computational complexity analysis for the MC method in the context of scalar conservation laws.

This slow convergence has inspired the development of *Multi-Level Monte Carlo* or MLMC methods (see [12, 9, 10, 4]). In particular, recent papers [20] and [21] extended and analyzed the MLMC algorithm for scalar conservation laws and for systems of conservation laws, respectively. The asymptotic analysis for the MLMC method, presented in [20], showed that the method allows the computation of approximate statistical moments with the *same accuracy versus cost ratio as a single deterministic solve on the same mesh*. An optimal static load balancing strategy proposed in [21] enabled us to compute realistic solutions of the multi-dimensional random Euler and magnetohydrodynamics (MHD) equations.

Our *first aim* in this paper is to extend the MLMC method to a system of balance laws like the random shallow water equations (1.6) and to demonstrate that it constitutes a considerable speed-up over the MC method. The key difference from the recent paper [21] lies in the fact that the uncertain bottom topography acts as a random source term. In practice [13, 5], the uncertainty in bottom topography is realized as random (in general, spatially correlated) fluctuations on each node of an underlying mesh. This implies possibly infinitely many sources of randomness. As the MLMC algorithm of [20, 21] entails that the bottom topography (modeled on a very fine mesh) needs to be sampled on coarse meshes, the very large number of topography samples with high spatial resolution will render the MLMC algorithm inefficient. Therefore, our *second aim* in this paper is to propose a *novel multi-level representation* of random nodal values of the underlying (uncertain) bottom topography in terms of a hierarchical basis. This representation, together with the structure of the spatial discretization, allows us to make the bottom topography samples from a finer mesh redundant when the solution is computed on a given mesh. Hence, the new MLMC algorithm is considerably faster.

The efficiency of the algorithm, its robustness and favorable parallelization performance are demonstrated throughout the numerical experiments in both one and two space dimensions. The rest of the paper is organized as follows: we present the MC and MLMC algorithms from [20, 21] in Sections 3 and 4, respectively. In Section 5, we describe an efficient multi-level alias-free representation of the uncertain bottom topography. Numerical experiments in one and two space dimensions are presented in section 6.

2. Well-balanced energy stable finite volume schemes

Any numerical approximation of the random shallow water equations requires a discretization of the underlying deterministic problem (1.2). As mentioned in the introduction, we will use the schemes developed in a recent paper [14]. For

simplicity, we present the schemes in one space dimension. Denote,

$$\bar{a}_{i+\frac{1}{2}} = \frac{a_i + a_{i+1}}{2}, \quad [[a]]_{i+\frac{1}{2}} = a_{i+1} - a_i.$$

The scheme proposed in [14] is of the form:

$$(2.1) \qquad \frac{\partial}{\partial t}\mathbf{U}_i = -\frac{1}{\Delta x}\left(\mathbf{F}^{ES1}_{i+\frac{1}{2}} - \mathbf{F}^{ES1}_{i-\frac{1}{2}}\right) - \frac{g}{2\Delta x}\begin{bmatrix} 0 \\ \bar{h}_{i+\frac{1}{2}}[[b]]_{i+\frac{1}{2}} + \bar{h}_{i-\frac{1}{2}}[[b]]_{i-\frac{1}{2}} \end{bmatrix}.$$

For first-order schemes, the numerical flux is given by

$$\mathbf{F}^{ES1}_{i+\frac{1}{2}} = \mathbf{F}^{EC}_{i+\frac{1}{2}} - \frac{1}{2}D^{ES1}_{i+\frac{1}{2}}[[\mathbf{V}]]_{i+\frac{1}{2}},$$

where $\mathbf{V} = \partial_{\mathbf{U}}E = [g(h+b) - \frac{u^2}{2}, u]^\top$ is the vector of entropy variables. The *entropy conservative flux* is

$$\mathbf{F}^{EC}_{i+\frac{1}{2}} = \begin{bmatrix} \bar{h}_{i+\frac{1}{2}}\bar{u}_{i+\frac{1}{2}} \\ \bar{h}_{i+\frac{1}{2}}(\bar{u}_{i+\frac{1}{2}})^2 + \frac{g}{2}\overline{(h^2)}_{i+\frac{1}{2}} \end{bmatrix},$$

and the numerical diffusion operator is

$$D^{ES1}_{i+\frac{1}{2}} = R_{i+\frac{1}{2}}|\Lambda_{i+\frac{1}{2}}|R^\top_{i+\frac{1}{2}},$$

with

$$R_{i+\frac{1}{2}} = \frac{1}{\sqrt{2g}}\begin{bmatrix} 1 & 1 \\ \lambda_- & \lambda_+ \end{bmatrix}, \quad \lambda_\pm = \bar{u}_{i+\frac{1}{2}} \pm \sqrt{g\bar{h}_{i+\frac{1}{2}}}, \quad |\Lambda_{i+\frac{1}{2}}| = \begin{bmatrix} |\lambda_-| & 0 \\ 0 & |\lambda_+| \end{bmatrix}.$$

A second-order scheme is obtained by replacing the flux $\mathbf{F}^{ES1}_{i+\frac{1}{2}}$ in (2.1) with $\mathbf{F}^{ES2}_{i+\frac{1}{2}}$,

$$(2.2) \qquad \mathbf{F}^{ES2}_{i+\frac{1}{2}} = \mathbf{F}^{EC}_{i+\frac{1}{2}} - \frac{1}{2}D^{ES1}_{i+\frac{1}{2}}\left(\mathbf{V}^+_{i+\frac{1}{2}} - \mathbf{V}^-_{i+\frac{1}{2}}\right),$$

where $\mathbf{V}^\pm$ are obtained from either a MINMOD or ENO reconstruction of the *scaled entropy variables*, following the procedure of [15].

Both first- and second-order schemes satisfy a discrete version of the energy inequality (1.3) and preserve the discrete lake at rest [14]

$$u_i \equiv 0, \quad h_i + b_i \equiv \text{constant}.$$

Time integration is performed using the forward Euler method and a strong stability preserving Runge-Kutta 2 method for first and second order schemes, respectively. The extension of this scheme to two space dimensions is detailed in [14].

3. Monte Carlo Finite Volume Method

The next step in approximating the random shallow water equations (1.6) is to discretize the probability space. The simplest sampling method is the Monte Carlo (MC) algorithm consisting of the following three steps:

1. **Sample:** We draw M independent identically distributed (i.i.d.) initial data and source samples $\{\mathbf{U}_0^k, \mathbf{S}^k\}$ with $k = 1, 2, \ldots, M$ from the random fields $\{\mathbf{U}_0, \mathbf{S}\}$ and approximate these by piecewise constant cell averages.
2. **Solve:** For each realization $\{\mathbf{U}_0^k, \mathbf{S}^k\}$, the underlying balance law (1.2) is solved numerically by the finite volume method (1.5). We denote the FVM solutions by $\mathbf{U}^{k,n}_{\mathcal{T}}$, i.e. by cell averages $\{\mathbf{U}^{k,n}_C : C \in \mathcal{T}\}$ at the time level t^n,

$$\mathbf{U}^{k,n}_{\mathcal{T}}(\mathbf{x}) = \mathbf{U}^{k,n}_C, \quad \forall \mathbf{x} \in C,\ C \in \mathcal{T}.$$

3. **Estimate Statistics:** We estimate the expectation of the random solution field with the sample mean (ensemble average) of the approximate solution:

$$(3.1) \qquad E_M[\mathbf{U}_{\mathcal{T}}^n] := \frac{1}{M} \sum_{i=1}^{M} \mathbf{U}_{\mathcal{T}}^{k,n}.$$

Higher statistical moments can be approximated analogously under suitable statistical regularity of the underlying random entropy solutions [20].

The above algorithm is quite simple to implement. We remark that step 1 requires a (pseudo) random number generator. In step 2, any standard (high-order) finite volume scheme can be used. Hence, existing code for FVM can be used and there is no need to rewrite FVM code. Furthermore, the only (data) interaction between different samples is in step 3 when ensemble averages are computed. Thus, the MC-FVM is non-intrusive as well as easily parallelizable.

Although a rigorous error estimate for the MC-FVM approximating shallow water equations is currently out of reach, we rely on our analysis for a scalar conservation law in [20] and on our numerical experience with the MLMC-FVM solution of non-linear hyperbolic systems of conservation laws with random initial data in [21] to *postulate* that the following estimate holds:

$$(3.2) \qquad \|\mathbb{E}[\mathbf{U}(\cdot, t^n)] - E_M[\mathbf{U}_{\mathcal{T}}^n]\|_{L^2(\Omega; L^1(\mathbb{R}^d))} \le C_{\text{stat}} M^{-\frac{1}{2}} + C_{\text{st}} \Delta x^s.$$

Here, the $L^2(\Omega; L^1(\mathbb{R}^d))$-norm of the random function $f(\mathbf{x}, \omega)$ is defined as

$$\|f\|_{L^2(\Omega; L^1(\mathbb{R}^d))} := \left(\int_{\omega \in \Omega} \|f(\cdot, \omega)\|_{L^1(\mathbb{R}^d)}^2 d\mathbb{P}(\omega) \right)^{\frac{1}{2}},$$

and $C_{\text{stat}}, C_{\text{st}}$ are constants that depend only on the initial data and on the source term in (1.6). In the above, we have assumed that the underlying finite volume scheme converges to the solutions of the deterministic shallow water equations (1.1) at a rate of $s > 0$. Moreover, in (3.2) and throughout the following, we adopted the (customary in the analysis of MC methods) convention to interpret the MC samples $\mathbf{U}_{\mathcal{T}}^{k,n}$ in (3.1) as i.i.d. random functions, with the same law as $\mathbf{U}$.

Note that the error estimate for the mean requires that the solution has finite second moments. Based on the error analysis of [20], we need to choose [20, 21]

$$(3.3) \qquad M = \mathcal{O}(\Delta x^{-2s})$$

in-order to equilibrate the statistical error with the spatio-temporal error in (3.2). Consequently, it is straightforward to deduce that the asymptotic error vs. (computational) work estimate is then given by

$$(3.4) \qquad \|\mathbb{E}[\mathbf{U}(\cdot, t^n)] - E_M[\mathbf{U}_{\mathcal{T}}^n]\|_{L^2(\Omega; L^1(\mathbb{R}^d))} \lesssim (\text{Work})^{-s/(d+1+2s)}.$$

The above error vs. work estimate is considerably more expensive when compared to the deterministic FVM error which scales as $(\text{Work})^{-s/(d+1)}$ [20].

4. Multi-level Monte Carlo Finite Volume Method

Given the slow convergence of MC-FVM, we propose the Multi-Level Monte Carlo finite volume method (MLMC-FVM). The key idea behind MLMC-FVM is to simultaneously draw MC samples on a hierarchy of nested grids [20].

4.1. **MLMC-FVM algorithm.** The algorithm consists of the following four steps:

1. **Nested meshes:** Consider *nested* meshes $\{\mathcal{T}_\ell\}_{\ell=0}^\infty$ of the domain $\mathbf{D}$ with corresponding mesh widths $\Delta x_\ell = 2^{-\ell}\Delta x_0$, where Δx_0 is the mesh width for the coarsest resolution and corresponds to the lowest level $\ell = 0$.
2. **Sample:** For each level of resolution $\ell \in \mathbb{N}_0$, we draw M_ℓ independent identically distributed (i.i.d) samples $\{\mathbf{U}_{0,\ell}^k, \mathbf{S}_\ell^k\}$ with $k = 1, 2, \ldots, M_\ell$ from the random fields $\{\mathbf{U}_0, \mathbf{S}\}$ and approximate these by cell averages.
3. **Solve:** For each resolution level ℓ and each realization $\{\mathbf{U}_{0,\ell}^k, \mathbf{S}_\ell^k\}$, the underlying balance law (1.2) is solved by the finite volume method (1.5) with mesh width Δx_ℓ. Denote the finite volume solutions by $\mathbf{U}_{\mathcal{T}_\ell}^{k,n}$, i.e. by cell averages $\{\mathbf{U}_C^{k,n} : C \in \mathcal{T}_\ell\}$ at the time level t^n and resolution level ℓ.
4. **Estimate solution statistics:** Fix some positive integer $L < \infty$ corresponding to the highest level. We estimate the expectation of the random solution field with the following estimator:

$$(4.1) \qquad E^L[\mathbf{U}_\mathcal{T}^n] := \sum_{\ell=0}^L E_{M_\ell}[\mathbf{U}_{\mathcal{T}_\ell}^n - \mathbf{U}_{\mathcal{T}_{\ell-1}}^n],$$

with E_{M_ℓ} being the MC estimator defined in (3.1) for the level ℓ. Higher statistical moments can be approximated analogously (see, e.g., the sparse tensor discretization of [20]).

MLMC-FVM is *non-intrusive* as any standard FVM code can be used in step 3. Furthermore, MLMC-FVM is amenable to *efficient parallelization* as data from different grid resolutions and different samples only interacts in step 4.

Again, based on the rigorous estimate for scalar conservation laws in [20] and on the experience for systems of conservation laws [21], we *postulate* the following error estimate:
(4.2)

$$\|\mathbb{E}[\mathbf{U}(\cdot, t^n)] - E^L[\mathbf{U}_\mathcal{T}^n]\|_{L^2(\Omega; L^1(\mathbb{R}^d))} \le C_1 \Delta x_L^s + C_2 \Big\{ \sum_{\ell=0}^L M_\ell^{-\frac{1}{2}} \Delta x_\ell^s \Big\} + C_3 M_0^{-\frac{1}{2}}.$$

Here s refers to the convergence rate of the deterministic finite volume scheme and $C_{1,2,3}$ are constants depending only on the initial data and the source term.

From the error estimate (4.2), we obtain that the number of samples to equilibrate the statistical and spatio-temporal discretization errors in (4.1) is given by

$$(4.3) \qquad M_\ell = \mathcal{O}(2^{2(L-\ell)s}).$$

Notice that (4.3) implies that the largest number of MC samples is required on the coarsest mesh level $\ell = 0$, whereas only a small fixed number of MC samples are needed on the finest discretization levels.

The corresponding error vs. work estimate for MLMC-FVM is given by [21, 20],

$$(4.4) \qquad \|\mathbb{E}[u(\cdot, t^n)] - E^L[u(\cdot, t^n)]\|_{L^2(\Omega; L^1(\mathbb{R}^d))} \lesssim (\text{Work})^{-s/(d+1)} \cdot \log(\text{Work}),$$

provided $s < (d + 1)/2$. The above estimates show that MLMC-FVM (4.4) is superior to MC-FVM (3.4). Furthermore, (4.4) is almost (up to logarithmic term) of the same order as the estimate for the *deterministic* finite volume scheme. For the same error, MLMC-FVM is expected to be considerably faster than MC-FVM.

5. Multi-level alias-free representation of uncertain bottom topography

An approximation to the exact bottom topography $b(\mathbf{x}) \in W^{1,\infty}(\mathbf{D})$ is often obtained from the measurements. For instance [5, 13], in two-dimensional case, nodal measurements $b_{i+\frac{1}{2},j+\frac{1}{2}} := b(\mathbf{x}_{i+\frac{1}{2},j+\frac{1}{2}})$ are obtained at locations $\mathbf{x}_{i+\frac{1}{2},j+\frac{1}{2}} = (x_{i+\frac{1}{2}}, y_{j+\frac{1}{2}})$, i.e. at vertices of an axiparallel quadrilateral *topography* mesh $\bar{\mathcal{T}}$ (possibly different from the FVM mesh $\mathcal{T}$) on the rectangular two-dimensional domain $\mathbf{D}$. Since each measurement $b_{i+\frac{1}{2},j+\frac{1}{2}}$ is prone to uncertainty [13], all measured values are treated as random variables with some prescribed probability distribution; we choose

$$(5.1) \qquad b_{i+\frac{1}{2},j+\frac{1}{2}}(\omega) := b(x_{i+\frac{1}{2},j+\frac{1}{2}}) + Y_{i,j}(\omega), \quad Y_{i,j} \sim \mathcal{U}(-\varepsilon_{i,j}, \varepsilon_{i,j}), \ \varepsilon_{i,j} > 0,$$

i.e. $b_{i+\frac{1}{2},j+\frac{1}{2}}(\omega) \in L^2(\Omega, \mathbb{R})$ are random variables (*not* necessarily independent), which deviate from the measurements $b_{i+\frac{1}{2},j+\frac{1}{2}}$ by $\pm \varepsilon_{i+\frac{1}{2},j+\frac{1}{2}}$ with $\mathcal{U}$ being the uniform distribution (other distributions can be considered analogously.) Thus, (5.1) provides an approximation to the *uncertain* topography $b(\mathbf{x}, \omega) \in L^2(\Omega, W^{1,\infty}(\mathbf{D}))$.

In two space dimensions, if we assume that the bottom topography $b(\mathbf{x}, \omega)$ is a continuous piece-wise linear function, the *energy conservative well-balanced* discretization of the source term $\mathbf{S}_{i,j}$ is given by [14]

$$(5.2) \qquad \mathbf{S}_{i,j}^{\mathrm{EC}} := \begin{bmatrix} 0 \\ \frac{g}{2\Delta x} \left(\bar{h}_{i+\frac{1}{2},j}[[b]]_{i+\frac{1}{2},j} + \bar{h}_{i-\frac{1}{2},j}[[b]]_{i-\frac{1}{2},j} \right) \\ \frac{g}{2\Delta y} \left(\bar{h}_{i,j+\frac{1}{2}}[[b]]_{i,j+\frac{1}{2}} + \bar{h}_{i,j-\frac{1}{2}}[[b]]_{i,j-\frac{1}{2}} \right) \end{bmatrix},$$

which amounts to computing the averages of $S_{i\pm\frac{1}{2},j}(\omega)$ and $S_{i,j\pm\frac{1}{2}}(\omega)$,

$$(5.3) \qquad \mathbf{S}_{i,j}(\omega) := \begin{bmatrix} 0 \\ \frac{g}{2\Delta x} \left(S_{i-\frac{1}{2},j}(\omega) + S_{i+\frac{1}{2},j}(\omega) \right) \\ \frac{g}{2\Delta y} \left(S_{i,j-\frac{1}{2}}(\omega) + S_{i,j+\frac{1}{2}}(\omega) \right) \end{bmatrix},$$

where $S_{i+\frac{1}{2},j}(\omega)$ and $S_{i,j+\frac{1}{2}}(\omega)$ are the integrals of the *weak directional derivatives* of a *continuous piecewise linear* bottom topography $b(\mathbf{x}, \omega)$ over "shifted" cells $C_{i+\frac{1}{2},j}, C_{i,j+\frac{1}{2}}$ of the mesh $\mathcal{T}$,

$$(5.4) \qquad \begin{aligned} S_{i+\frac{1}{2},j}(\omega) &:= \int_{C_{i+\frac{1}{2},j}} \bar{h}_{i+\frac{1}{2},j}(\omega) \partial_x b(\mathbf{x}, \omega) \, d\mathbf{x}, \\ S_{i,j+\frac{1}{2}}(\omega) &:= \int_{C_{i,j+\frac{1}{2}}} \bar{h}_{i,j+\frac{1}{2}}(\omega) \partial_y b(\mathbf{x}, \omega) \, d\mathbf{x}. \end{aligned}$$

In one-dimensional case, the last term in (5.3) is dropped and the first integral in (5.4) is taken over cells $C_{i+\frac{1}{2}} = (x_i, x_{i+1})$, resulting in the one-dimensional energy conservative source discretization in (2.1). Since MLMC methods require that a deterministic problem is solved by the means of such FVM on the coarsest mesh level for a *very large* number (often around 10^{10}) of samples, the evaluation of the full bottom topography for *each* sample becomes computationally infeasible.

To remedy this, we consider a *hierarchical multi-level* representation of the bottom topography. The key to an efficient MLMC simulation of the uncertain random topography (with very large numbers of sources of uncertainty) is that only some of

the hierarchical bottom topography levels need to be evaluated; in particular, only these levels, which are coarser than or coincide with the active FVM discretization level, are necessary. An active level is that level of resolution on which the FVM computation is being performed for a given sample.

5.1. Preliminaries. To introduce the multi-level topography representation, we recall some notation: levels $\ell = 0, \ldots, L$ enumerate nested grids $\mathcal{T}_0, \ldots, \mathcal{T}_L$ that are used in the MLMC-FVM *solver*. Apart from $\mathcal{T}_0, \ldots, \mathcal{T}_L$, we consider an additional hierarchical structure, that will be used in the multi-level representation of the *bottom topography*. More precisely, assume a nested sequence $\{\bar{\mathcal{T}}_{\bar{\ell}} = \bar{\mathcal{T}}_{\bar{\ell}}^1 \times \cdots \times \bar{\mathcal{T}}_{\bar{\ell}}^d, \ \bar{\ell} = 0, \ldots, \bar{L}\}$ of isotropic regular d-dimensional axiparallel quadrilateral meshes for the physical *bounded* domain $\mathbf{D} = I_1 \times \cdots \times I_d \subset \mathbb{R}^d$, $I_r \subset \mathbb{R}$, $d = 1, 2$, each of them obtained by $\bar{\ell}$ uniform refinements of some initial, regular mesh $\bar{\mathcal{T}}_0$ (of domain $\mathbf{D}$) consisting of the cells C_k^0, $k = 1, \ldots, \#\bar{\mathcal{T}}_0$. Note, that a-priori we do *not* assume any relation between $\bar{L}$ and L. However, for the sake of consistency, we assume

$$\bar{\mathcal{T}}_{\bar{\ell}} = \mathcal{T}_\ell, \quad \text{provided } \bar{\ell} = \ell.$$

For $p \in \mathbb{N}_0$, define $\mathcal{Q}^p(\mathbf{D}, \bar{\mathcal{T}})$ to be the space of piece-wise multivariate tensor product polynomials of degree p on a mesh $\bar{\mathcal{T}}$ of a bounded domain $\mathbf{D}$ having essentially bounded weak derivatives up to order p, i.e.

$$\mathcal{Q}^p(\mathbf{D}, \bar{\mathcal{T}}) := \{f \in W^{p,\infty}(\mathbf{D}) : f|_C \in \mathcal{Q}_p(C), \ \forall C = C_1 \times \cdots \times C_d \in \bar{\mathcal{T}}\},$$

where $\mathcal{Q}_p(C)$ is the space of multivariate tensor product polynomials on cell C,

$$\mathcal{Q}_p(C) := \{\mathbf{x} \mapsto p_1(\mathbf{x}_1) \cdot \cdots \cdot p_d(\mathbf{x}_d) : p_r \in \mathcal{P}_d(C_r), \ \forall r = 1, \ldots, d\}.$$

5.1.1. Haar wavelets in 1d. Assume $d = 1$ and observe, that for any interval $I \subset \mathbb{R}$, there is an orthogonal decomposition of $\mathcal{Q}^0(I, \bar{\mathcal{T}}_{\bar{L}})$,

$$(5.5) \qquad \mathcal{Q}^0(I, \bar{\mathcal{T}}_{\bar{L}}) = \bigoplus_{\bar{\ell}=0}^{\bar{L}} R_{\bar{\ell}}, \quad R_{\bar{\ell}} := \mathcal{Q}^0(I, \bar{\mathcal{T}}_{\bar{\ell}}) \cap \mathcal{Q}^0(I, \bar{\mathcal{T}}_{\bar{\ell}-1})^\perp.$$

An $L^2(I)$-orthogonal basis of $\mathcal{Q}^0(I, \bar{\mathcal{T}}_{\bar{L}})$ can be explicitly constructed as follows. For each $\bar{\ell} \in \mathbb{N}_0$, every cell $C_k^{\bar{\ell}} \in \bar{\mathcal{T}}_{\bar{\ell}}$ is affinely equivalent to the reference cell $\hat{C} = (0, 1)$, i.e. for all $k = 1, \ldots, \#\bar{\mathcal{T}}_{\bar{\ell}}$, there are affine mappings

$$F_k^{\bar{\ell}} : \hat{C} \ni \hat{x} \to x \in C_k^{\bar{\ell}} \in \bar{\mathcal{T}}_{\bar{\ell}}, \quad |DF_k^{\bar{\ell}}| = |C_k^{\bar{\ell}}| = \mathcal{O}(2^{-\bar{\ell}}).$$

Let $\hat{\mathcal{T}}_0 = \{\hat{C}\}$ and define $\hat{\mathcal{T}}_1 = \{\hat{C}_1, \hat{C}_2\}$ to be the set of two cells $\hat{C}_1 = (0, 1/2)$, $\hat{C}_2 = (1/2, 1)$ that are obtained by uniform subdivision of the reference cell $\hat{C}$. Then

$$\dim\left(\mathcal{Q}^0(\hat{C}, \hat{\mathcal{T}}_0)\right) = 1, \quad \dim\left(\mathcal{Q}^0(\hat{C}, \hat{\mathcal{T}}_1) \cap \mathcal{Q}^0(\hat{C}, \hat{\mathcal{T}}_0)^\perp\right) = 1.$$

Denote by $\{\hat{\varphi}\}$ the $L^2(\hat{C})$-normalized basis of $\hat{W}_0 = \mathcal{Q}^0(\hat{C}, \hat{\mathcal{T}}_0)$ and by $\{\hat{\psi}\}$ the $L^2(\hat{C})$-normalized basis of $\hat{W}_1 = \mathcal{Q}^0(\hat{C}, \hat{\mathcal{T}}_1) \cap \mathcal{Q}^0(\hat{C}, \hat{\mathcal{T}}_0)^\perp$. Next we define Ψ_0 by

$$\Psi_0 := \left\{\psi_k^0 : \psi_k^0 \circ F_k^0 = \hat{\varphi} \in \hat{W}_0, \ k = 1, \ldots, \#\bar{\mathcal{T}}_0\right\},$$

and, for every $\bar{\ell} \geq 1$, we define $\Psi_{\bar{\ell}}$ to be the set of affine images of the (mother-wavelets) $\hat{\psi}_n$ under the affine mappings $F_k^{\bar{\ell}-1}$, i.e.

$$\Psi_{\bar{\ell}} := \left\{\psi_k^{\bar{\ell}} : \psi_k^{\bar{\ell}} \circ F_k^{\bar{\ell}-1} = \hat{\psi} \in \hat{W}_1, \ k = 1, \ldots, \#\bar{\mathcal{T}}_{\bar{\ell}-1}\right\}, \quad \bar{\ell} \geq 1.$$

By construction, $\psi_k^{\bar{\ell}}$ forms an $L^2(I)$-orthogonal system and $R_{\bar{\ell}} = \mathrm{span}(\Psi_{\bar{\ell}})$ in (5.5).

In the remainder of this paper, we confine ourselves to the so-called Haar wavelets in one dimension, i.e. $\hat{\varphi}(x) \equiv 1$ and the mother-wavelet $\hat{\psi}$ is given by

$$\hat{\psi}(x) = \chi_{[0, \frac{1}{2})}(x) - \chi_{[\frac{1}{2}, 1)}(x), \quad \forall x \in \hat{C}.$$

5.2. Multi-level representation of bottom topography in one dimension.

For $d = 1$ (i.e. $\mathbf{D} = I_1$), we will assume that *uncertain* measurements $b_{i+\frac{1}{2}}(\omega) := b(x_{i+\frac{1}{2}}, \omega)$ of the exact bottom topography $b(x)$ are available, as described in (5.1). Then $b_{i+\frac{1}{2}}(\omega)$ are treated as nodal values and are linearly interpolated, see Figure 1.
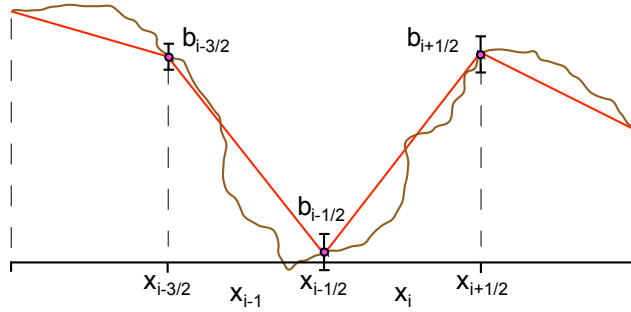
FIGURE 1. Uncertain measurements $b_{i+\frac{1}{2}}(\omega)$ of the exact bottom topography $b(x)$ at locations $x_{i+\frac{1}{2}}$ are treated as nodal values and then are linearly interpolated on the finest mesh level $\bar{L}$.

Furthermore, we use the nodal *hierarchical* "hat" basis interpolation operator $\mathcal{I}^{\bar{L}}$,

$$(5.6) \qquad \mathcal{I}^{\bar{L}} b(x, \omega) = \sum_{\bar{\ell}=0}^{\bar{L}} b^{\bar{\ell}}(x, \omega), \quad b^{\bar{\ell}} := \mathcal{I}_{\bar{\ell}} b - \mathcal{I}_{\bar{\ell}-1} b, \quad \mathcal{I}_{-1} \equiv 0,$$

where $\mathcal{I}_{\bar{\ell}}$ denotes linear *nodal* interpolation operator on the mesh $\bar{\mathcal{T}}_{\bar{\ell}}$. The finest level $\bar{L} < \infty$ corresponds to the "pixel level" of a terrain imaging device.

Each $b^{\bar{\ell}}(x, \omega) \in L^2(\Omega, \mathcal{Q}^1(I_1, \bar{\mathcal{T}}_{\bar{\ell}}))$ is a linear combination of the *hierarchical* "hat" basis ("Schauder basis", see Figure 2) functions $\varphi_k^{\bar{\ell}}(x)$, i.e.

$$(5.7) \qquad b^{\bar{\ell}}(x, \omega) = \sum_{k=1}^{\hat{N}_{\bar{\ell}}} b_k^{\bar{\ell}}(\omega) \varphi_k^{\bar{\ell}}(x), \quad b_k^{\bar{\ell}} \in L^2(\Omega, \mathbb{R}),$$

where

$$\hat{N}_{\bar{\ell}} := \dim(\mathcal{Q}^1(I_1, \bar{\mathcal{T}}_{\bar{\ell}})) - \sum_{\bar{\ell}'=0}^{\bar{\ell}-1} \dim(\mathcal{Q}^1(I_1, \bar{\mathcal{T}}_{\bar{\ell}'})),$$
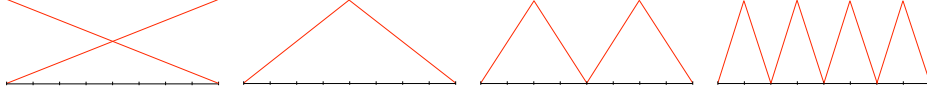
i.e. $\hat{N}_0 = 2$ and $\hat{N}_{\bar{\ell}} = \#\bar{\mathcal{T}}_{\bar{\ell}-1} = 2^{\bar{\ell}-1}$, $\forall \bar{\ell} \geq 1$.

The interpolated bottom topography belongs to the space

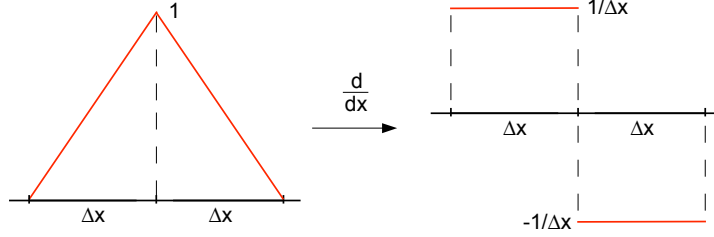$$\mathcal{I}_{\bar{L}} b(x, \omega) \in L^2(\Omega, \mathcal{Q}^1(I_1, \bar{\mathcal{T}}_{\bar{L}})),$$

and the weak spatial derivative of the interpolated topography belongs to the space

$$\partial_x(\mathcal{I}_{\bar{L}} b)(x, \omega) \in L^2(\Omega, \mathcal{Q}^0(I_1, \bar{\mathcal{T}}_{\bar{L}})).$$

FIGURE 2. Hierarchical "hat" basis functions $\varphi_k^{\bar{\ell}}(x)$ for the first 4 levels.

Since $\partial_x \varphi_k^{\bar{\ell}}(x) = \frac{1}{\Delta x_{\bar{\ell}}} \psi_k^{\bar{\ell}}(x)$, (see Figure 3), we have

$$\partial_x b^{\bar{\ell}}(x, \omega) = \frac{1}{\Delta x_{\bar{\ell}}} \sum_{k=1}^{\hat{N}_{\bar{\ell}}} b_k^{\bar{\ell}}(\omega) \psi_k^{\bar{\ell}}(x).$$

FIGURE 3. Derivative of a one-dimensional hierarchical basis function $\varphi_k^{\bar{\ell}}(x)$ is a scaled Haar wavelet $\frac{1}{\Delta x_{\bar{\ell}}} \psi_k^{\bar{\ell}}(x)$.

Consequently, $B := \partial_x(\mathcal{I}_{\bar{L}} b)$ admits the Haar wavelet representation,

(5.8)
$$B(x, \omega) := B^{\bar{L}}(x, \omega) := \sum_{\bar{\ell}=0}^{\bar{L}} \sum_{k=1}^{\#\bar{\mathcal{T}}_{\bar{\ell}-1}} B_k^{\bar{\ell}}(\omega) \psi_k^{\bar{\ell}}(x),$$

where the "coefficients" $B_k^{\bar{\ell}}(\omega) \in L^2(\Omega, \mathbb{R})$ are random variables given by

$$B_k^0(\omega) = \frac{1}{\Delta x_0} \left(b_1^0(\omega) + b_2^0(\omega) \right), \quad B_k^{\bar{\ell}}(\omega) = \frac{1}{\Delta x_{\bar{\ell}}} b_k^{\bar{\ell}}(\omega), \quad \forall \bar{\ell} > 0.$$

Let $h_i^{\ell}(\omega)$ be cell averages of water level $h(x, \omega)$ above bottom topography at the mesh level $0 \le \ell \le L$, i.e. $h^{\ell}(x, \omega) \in L^2(\Omega, \mathcal{Q}^0(I_1, \mathcal{T}_{\ell}))$, $h^{\ell}(x, \omega) := h_i^{\ell}(\omega)$, $\forall x \in C_i^{\ell}$. The terms $S_{i+\frac{1}{2}}^{\ell, \bar{L}} \in L^2(\Omega, \mathbb{R})$ needed in one-dimensional analogue of (5.4) are given by integrating over "shifted" cells $C_{i+\frac{1}{2}}^{\ell} = (x_i^{\ell}, x_{i+1}^{\ell})$ of the mesh $\mathcal{T}_{\ell}$,

(5.9)
$$S_{i+\frac{1}{2}}^{\ell, \bar{L}}(\omega) = \int_{C_{i+\frac{1}{2}}^{\ell}} \bar{h}_{i+\frac{1}{2}}^{\ell}(\omega) B^{\bar{L}}(x, \omega) \, dx, \quad i = 0, \dots, \#\mathcal{T}_{\ell}.$$

The first superscript ℓ of $S_i^{\ell, \bar{L}}$ denotes the FVM mesh level. The second superscript $\bar{L}$ denotes "pixel level" in the multi-level topography representation (5.8).

Lemma 5.1. *Assume $d = 1$ and that $B(x, \omega)$ is given in the form (5.8). By $B^{\hat{L}}(x, \omega)$ denote the hierarchical representation (5.8) truncated up to level $\hat{L} \le \bar{L}$. Then, for all $\bar{L} \ge \hat{L} \ge \ell + 1$, we have*

(5.10)
$$S_{i+\frac{1}{2}}^{\ell, \bar{L}}(\omega) = S_{i+\frac{1}{2}}^{\ell, \hat{L}}(\omega), \quad i = 0, \dots, \#\mathcal{T}_{\ell}.$$

Lemma 5.1 states that it is sufficient to compute only terms up to level $\ell + 1$ in the hierarchical bottom topography representation (5.6) when the underlying FVM solver is on the mesh level ℓ. This way, computational work is significantly reduced on the coarsest levels $\ell \ll \bar{L}$.

Proof. Notice, that due to the vanishing moments of the wavelet basis functions,

$$\int_{C^\ell_{i+\frac{1}{2}}} \psi^{\bar{\ell}}_k(x)dx = 0 \quad \text{for } \bar{\ell} > \ell + 1.$$

Since $\bar{h}^\ell_{i+\frac{1}{2}}(\omega)$ is constant and equals $(h^\ell_i(\omega) + h^\ell_{i+1}(\omega))/2$ in each cell $C^\ell_{i+\frac{1}{2}}$,

$$\int_{C^\ell_{i+\frac{1}{2}}} \bar{h}^\ell_{i+\frac{1}{2}}(\omega)B^{\bar{\ell}}_k(\omega)\psi^{\bar{\ell}}_k(x)dx = B^{\bar{\ell}}_k(\omega)\bar{h}^\ell_{i+\frac{1}{2}}(\omega) \int_{C^\ell_{i+\frac{1}{2}}} \psi^{\bar{\ell}}_k(x)dx = 0 \quad \text{for } \bar{\ell} > \ell + 1.$$

The terms for levels $\bar{\ell}$ in the sum (5.8)-(5.9) vanish provided $\bar{\ell} > \ell + 1$, hence these levels ($\bar{\ell} > \ell + 1$) can be disregarded entirely. □

5.3. Multi-level representation of bottom topography in two dimensions.
For two-dimensional case (i.e. $d = 2$, and Cartesian domain $\mathbf{D} = I_1 \times I_2$), we assume that *uncertain* measurements $b_{i+\frac{1}{2},j+\frac{1}{2}}(\omega) := b(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2}}, \omega)$ of the exact bottom topography $b(x, y)$ are available, as in (5.1). Then $b_{i+\frac{1}{2},j+\frac{1}{2}}(\omega)$ are treated as nodal values and are linearly interpolated in each dimension using the bilinear *hierarchical* interpolation operator,

$$(5.11) \quad \mathcal{I}^{\bar{L}}b(x,y,\omega) = \sum_{\bar{\ell}=0}^{\bar{L}} b^{\bar{\ell}}(x,y,\omega), \quad b^{\bar{\ell}} := \mathcal{I}_{\bar{\ell},\bar{\ell}}b - \mathcal{I}_{\bar{\ell}-1,\bar{\ell}-1}b, \quad \mathcal{I}_{-1,-1} \equiv 0,$$

where $\mathcal{I}_{\bar{\ell},\bar{\ell}}$ denotes bilinear *nodal* interpolation operator on the mesh $\bar{\mathcal{T}}_{\bar{\ell}} = \bar{\mathcal{T}}^1_{\bar{\ell}} \times \bar{\mathcal{T}}^2_{\bar{\ell}}$. The finest level $\bar{L} < \infty$ corresponds to the "pixel level" of a terrain imaging device.

In order to construct $\mathcal{I}^{\bar{L}}b$, we consider *isotropic* tensorization of the *hierarchical* "hat" ("Schauder") basis functions, i.e. each $b^{\bar{\ell}}(\mathbf{x}, \omega) \in L^2(\Omega, \mathcal{Q}^1(I_1 \times I_2, \bar{\mathcal{T}}_{\bar{\ell}}))$ is a linear combination of the multivariate tensor products of such basis functions,

$$
\begin{aligned}
b^{\bar{\ell}}(\mathbf{x}, \omega) = &\sum_{k,k'=1}^{\hat{N}_{\bar{\ell}}} b^{\bar{\ell}}_{k,k'}(\omega)\varphi^{\bar{\ell}}_k(x)\varphi^{\bar{\ell}}_{k'}(y) + \sum_{k=1}^{\hat{N}_{\bar{\ell}}}\sum_{k'=1}^{\hat{N}_{\bar{\ell}}-1} \check{b}^{\bar{\ell}}_{k,k'}(\omega)\varphi^{\bar{\ell}}_k(x)\varphi^{\bar{\ell}}_{k'+\frac{1}{2}}(y) \\
&+ \sum_{k=1}^{\hat{N}_{\bar{\ell}}-1}\sum_{k'=1}^{\hat{N}_{\bar{\ell}}} \hat{b}^{\bar{\ell}}_{k,k'}(\omega)\varphi^{\bar{\ell}}_{k+\frac{1}{2}}(x)\varphi^{\bar{\ell}}_{k'}(y), \quad b^{\bar{\ell}}_{k,k'}, \check{b}^{\bar{\ell}}_{k,k'}, \hat{b}^{\bar{\ell}}_{k,k'} \in L^2(\Omega, \mathbb{R}),
\end{aligned}
$$

(5.12)

where $\varphi^{\bar{\ell}}_{k+\frac{1}{2}}(x) := \varphi^{\bar{\ell}}_k(x - \frac{1}{2}\Delta x_\ell)$ and $\varphi^{\bar{\ell}}_{k'+\frac{1}{2}}(y) := \varphi^{\bar{\ell}}_{k'}(y - \frac{1}{2}\Delta y_\ell)$.
The interpolated bottom topography belongs to the space

$$\mathcal{I}_{\bar{L}}b(\mathbf{x}, \omega) \in L^2(\Omega, \mathcal{Q}^1(I_1 \times I_2, \bar{\mathcal{T}}_{\bar{L}})).$$

For $p_1, p_2 \in \mathbb{N}_0$, define the tensor product of *mixed* degree polynomial spaces (5.1),

$$\mathcal{Q}^{p_1,p_2}(I_1 \times I_2, \bar{\mathcal{T}}^1 \times \bar{\mathcal{T}}^2) := \mathcal{Q}^{p_1}(I_1, \bar{\mathcal{T}}^1) \otimes \mathcal{Q}^{p_2}(I_2, \bar{\mathcal{T}}^2).$$

Using this notation, the weak gradient of $\mathcal{I}_{\bar{L}}b$ belongs to the space

$$\mathbf{B}(\mathbf{x}, \omega) := \nabla(\mathcal{I}_{\bar{L}}b)(\mathbf{x}, \omega) \in L^2\left[\Omega, \mathcal{Q}^{0,1}(I_1 \times I_2, \bar{\mathcal{T}}_{\bar{L}}) \times \mathcal{Q}^{1,0}(I_1 \times I_2, \bar{\mathcal{T}}_{\bar{L}})\right].$$

Each component of the weak gradient $\mathbf{B} := (B_1, B_2)^\top$ is a tensor product of piece-wise constant and piece-wise linear continuous functions; in particular,

$$B_1(\cdot, y, \cdot) \in L^2(\Omega, \mathcal{Q}^0(I_1, \bar{\mathcal{T}}_{\bar{L}}^1)), \quad B_2(x, \cdot, \cdot) \in L^2(\Omega, \mathcal{Q}^0(I_2, \bar{\mathcal{T}}_{\bar{L}}^2)).$$

Hence, B_1 admits a *one-dimensional* Haar wavelet representation in x variable and B_2 admits *one-dimensional* Haar wavelet representation in y variable, i.e.

(5.13)
$$B_1(x, y, \omega) := B_1^{\bar{L}}(x, y, \omega) := \sum_{\bar{\ell}=0}^{\bar{L}} \sum_{k=1}^{\#\bar{\mathcal{T}}_{\bar{\ell}-1}^1} B_{1,k}^{\bar{\ell}}(y, \omega) \psi_k^{\bar{\ell}}(x),$$

$$B_2(x, y, \omega) := B_2^{\bar{L}}(x, y, \omega) := \sum_{\bar{\ell}=0}^{\bar{L}} \sum_{k'=1}^{\#\bar{\mathcal{T}}_{\bar{\ell}-1}^2} B_{2,k'}^{\bar{\ell}}(x, \omega) \psi_{k'}^{\bar{\ell}}(y),$$

where the coefficients $B_{1,k}^{\bar{\ell}}(y, \cdot), B_{2,k'}^{\bar{\ell}}(x, \cdot) \in L^2(\Omega, \mathbb{R})$ depend on $b_{\cdot,\cdot}^{\bar{\ell}}, \check{b}_{\cdot,\cdot}^{\bar{\ell}}, \hat{b}_{\cdot,\cdot}^{\bar{\ell}}$.

Let $h_{i,j}^\ell(\omega)$ be cell averages of water level $h(\mathbf{x}, \omega)$ above bottom topography at the mesh level $0 \leq \ell \leq L$, i.e. $h^\ell(\mathbf{x}, \omega) \in \mathcal{Q}^0(I_1 \times I_2, \mathcal{T}_\ell)$, $h^\ell(\mathbf{x}, \omega) := h_{i,j}^\ell(\omega), \ \forall \mathbf{x} \in C_{i,j}^\ell$. The terms $S_{i+\frac{1}{2},j}^{\ell,\bar{L}}, S_{i,j+\frac{1}{2}}^{\ell,\bar{L}} \in L^2(\Omega, \mathbb{R}^2)$ in (5.4) are given by integrating over "shifted" cells $C_{i+\frac{1}{2},j}^\ell = (x_i^\ell, x_{i+1}^\ell) \times (y_{j-\frac{1}{2}}^\ell, y_{j+\frac{1}{2}}^\ell)$, $C_{i,j+\frac{1}{2}}^\ell = (x_{i-\frac{1}{2}}^\ell, x_{i+\frac{1}{2}}^\ell) \times (y_j^\ell, y_{j+1}^\ell)$ of $\mathcal{T}_\ell$,

(5.14)
$$S_{i+\frac{1}{2},j}^{\ell,\bar{L}}(\omega) = \int_{C_{i+\frac{1}{2},j}^\ell} \bar{h}_{i+\frac{1}{2},j}^\ell(\omega) B_1^{\bar{L}}(\mathbf{x}, \omega) \ d\mathbf{x},$$

$$S_{i,j+\frac{1}{2}}^{\ell,\bar{L}}(\omega) = \int_{C_{i,j+\frac{1}{2}}^\ell} \bar{h}_{i,j+\frac{1}{2}}^\ell(\omega) B_2^{\bar{L}}(\mathbf{x}, \omega) \ d\mathbf{x}.$$

The first superscript ℓ of $S_{\cdot,\cdot}^{\ell,\bar{L}}$ denotes the FVM mesh level. The second superscript $\bar{L}$ denotes the "pixel level" in the multi-level topography representations (5.13).

Lemma 5.2. *Assume $d = 2$ and that the weak gradient $\mathbf{B}(\mathbf{x}, \omega)$ is given in the form (5.13). By $B_1^{\hat{L}}(\mathbf{x}, \omega), B_2^{\hat{L}}(\mathbf{x}, \omega)$ denote the hierarchical representations (5.13) truncated up to level $\hat{L} \leq \bar{L}$. Then, for all $\bar{L} \geq \hat{L} \geq \ell + 1$, we have*

(5.15)
$$S_{i+\frac{1}{2},j}^{\ell,\bar{L}}(\omega) = S_{i+\frac{1}{2},j}^{\ell,\hat{L}}(\omega), \quad i = 0, \ldots, \#\mathcal{T}_\ell^1, \ j = 1, \ldots, \#\mathcal{T}_\ell^2,$$

$$S_{i,j+\frac{1}{2}}^{\ell,\bar{L}}(\omega) = S_{i,j+\frac{1}{2}}^{\ell,\hat{L}}(\omega), \quad i = 1, \ldots, \#\mathcal{T}_\ell^1, \ j = 0, \ldots, \#\mathcal{T}_\ell^2.$$

Lemma 5.2 states that it is sufficient to compute only terms up to level $\ell+1$ in the hierarchical bottom topography representation (5.13). Hence, the computational work is significantly reduced on the coarsest mesh levels where $\ell \ll \bar{L}$.

Proof. The proof follows the main ideas presented in the proof of Lemma (5.1). Since the wavelet basis functions $\psi_k^{\bar{\ell}}(x), \ \psi_{k'}^{\bar{\ell}'}(y)$ have vanishing moments and since $\bar{h}_{i+\frac{1}{2},j}^\ell(\omega), \ \bar{h}_{i,j+\frac{1}{2}}^\ell(\omega)$ are constant in cells $C_{i+\frac{1}{2},j}^\ell, \ C_{i,j+\frac{1}{2}}^\ell$, respectively, we obtain

$$\int_{C_{i+\frac{1}{2},j}^\ell} \bar{h}_{i+\frac{1}{2},j}^\ell B_{1,k}^{\bar{\ell}}(y, \omega) \psi_k^{\bar{\ell}}(x) d\mathbf{x} = \bar{h}_{i+\frac{1}{2},j}^\ell \underbrace{\int_{C_{i+\frac{1}{2}}^\ell} \psi_k^{\bar{\ell}}(x) dx}_{=0 \text{ if } \bar{\ell} > \ell + 1} \int_{C_j^\ell} B_{1,k}^{\bar{\ell}}(y, \omega) dy = 0$$

and, analogously,

$$\int_{C^\ell_{i,j+\frac{1}{2}}} \bar{h}^\ell_{i,j+\frac{1}{2}} B^{\bar{\ell}}_{2,k'}(x,\omega)\psi^{\bar{\ell}}_{k'}(y)d\mathbf{x} = \bar{h}^\ell_{i,j+\frac{1}{2}} \int_{C^\ell_i} B^{\bar{\ell}}_{2,k'}(x,\omega)dx \underbrace{\int_{C^\ell_{j+\frac{1}{2}}} \psi^{\bar{\ell}}_{k'}(y)dy}_{=0 \text{ if } \bar{\ell}\,>\ell\,+1} = 0$$

provided $\bar{\ell}\,>\ell\,+1$. The terms for levels $\bar{\ell}$ in the sums (5.13) vanish provided $\bar{\ell}\,>\ell\,+1$, hence these levels can be discarded. $\square$

6. Numerical experiments

We tested the proposed MC and MLMC algorithms extensively for the shallow water equations with uncertain bottom topography, in one and two space dimensions. In particular, we are interested in verifying the following claims; i) MC and MLMC algorithms can handle a very large number of sources of uncertainty which are beyond the reach of existing deterministic methods like stochastic Galerkin and stochastic collocation, ii) MLMC method is considerably superior (faster) than the corresponding MC algorithm, iii) the hierarchical multi-level representation of the bottom topography (see the previous section) speeds up the MLMC algorithm and iv) an (efficient) massively parallel version of the MLMC algorithm can simulate complex and physically relevant shallow water flows with uncertain initial data and bottom topography.

All the simulations reported below are performed with the recently developed massively parallel code `ALSVID-UQ`. An extensive technical description of the implementation of `ALSVID-UQ` together with the results for linear scaling tests on massively parallel homogeneous distributed computing architectures can be found in [24]. All simulations were executed on two high performance computing clusters: "Brutus" in ETH Zürich [30] and "Palu" (Cray XE6) in CSCS, Manno [31].

As second-order high-resolution schemes are the basis of production codes [17], we concentrate on the second-order entropy stable schemes of [14]. In particular, we follow the second-order TeCNO implementation suggested in a recent paper [15]. Furthermore, the following parameters are required:

Parameter	Description
L	number of hierarchical mesh levels
M_L	number of samples at the finest mesh level
grid size	number of cells in X and in Y directions
CFL	CFL number based on the fastest wave
cores	total number of cores used in the simulation
runtime	clock-time (serial runs) or wall-time (parallel runs); hrs:min:sec
efficiency	MPI efficiency, as defined in (6.9)

6.1. **A 1-D dam break problem.** The computational domain is $[0,1]$ with outflow boundary conditions. We have an initial *dam*:

$$(6.1) \qquad \{h_0(x,\omega),\ u_0(x,\omega)\} = \begin{cases} \{2.0 + Y(\omega) - b(x,\omega),\ 0.0\} & \text{if}\quad x < 1.0, \\ \{1.5 - b(x,\omega),\ 0.0\} & \text{if}\quad x > 1.0, \end{cases}$$

with $Y(\omega) \sim \mathcal{U}(0,\frac{1}{10})$. Thus we have a dambreak problem with uncertain dam height. The uncertain bottom topography $b(x,\omega)$ is represented in terms of the hierarchical "Schauder" basis (5.6) - (5.7) with 10 levels (i.e. $\bar{L} = 9$, $\bar{\ell} = 0,\ldots,9$)

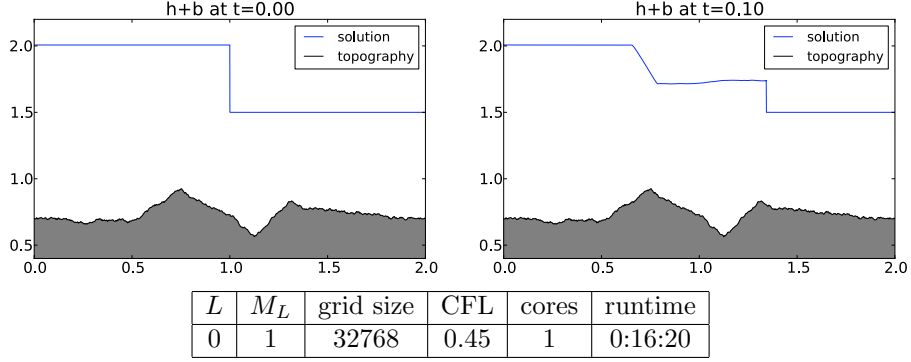| L	M_L	grid size	CFL	cores	runtime
0	1	32768	0.45	1	0:16:20

FIGURE 4. Typical sample path of the solution for the dambreak problem (6.1). Initial shock splits into a rarefaction and a shock.

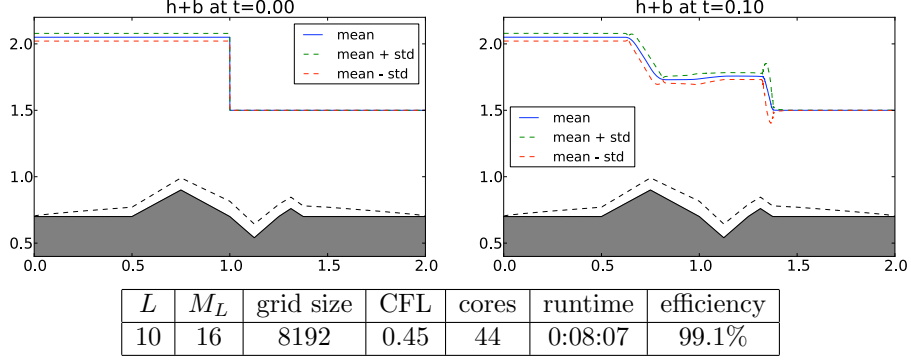| L	M_L	grid size	CFL	cores	runtime	efficiency
10	16	8192	0.45	44	0:08:07	99.1%

FIGURE 5. Mean and standard deviation of the reference solution for the dambreak problem (6.1) computed using MLMC-FVM with ES flux (2.2). Variance is concentrated around the shock.

where the coefficients $b_k^{\bar{\ell}}(\omega)$ are given by mean values $\mu_k^{\bar{\ell}}$ which are perturbed by *independent* uniformly distributed centered random variables with decaying variances, i.e.

$$(6.2) \quad b_k^{\bar{\ell}}(\omega) = 0.7 + \mu_k^{\bar{\ell}} + Y_k^{\bar{\ell}}(\omega) \sim \frac{1}{5}\mathcal{U}(-\varepsilon_{\bar{\ell}}, \varepsilon_{\bar{\ell}}), \quad \varepsilon_0 = 0, \quad \varepsilon_{\bar{\ell}} = \frac{1}{\bar{\ell}^{1.5}}, \ \forall \bar{\ell} \geq 1,$$

where all coefficients $\mu_k^{\bar{\ell}}$ are zero except

$$(6.3) \qquad\qquad \mu_2^3 = 0.2, \quad \mu_6^4 = -0.16, \quad \mu_{11}^5 = 0.06.$$

A realization of the bottom topography is shown in figure 4 and the mean (and variance) of the topography are shown in figure 5. To the best of our knowledge, such high dimensional problems (with $2^9 = 512$ sources of uncertainty) have not been considered in the literature.

A single realization of the reference height is shown in figure 4 and the solution mean and variance are shown in figure 5. The solution clearly has a left-moving rarefaction wave and a right-moving shock wave. The variance is large at the shock

wave but is also significant in other parts of the domain, due to uncertain bottom topography.

6.1.1. *Numerical convergence analysis.* Using MLMC-FVM approximation from Figure 5 (computed on 9 levels of resolution with the finest resolution being on a mesh of 8192 cells) as a reference solution $\mathbf{U}_{\mathrm{ref}}$, we run MC-FVM and MLMC-FVM on the series of mesh resolutions ranging from 32 cells up to 2048 cells and monitor the convergence behavior. The number of levels for the MLMC-FVM method is chosen so that the coarsest level always contains 8 cells. As all the simulations are performed using the massively parallel `ALSVID-UQ` [2], the runtime of the *parallel* algorithm is obtained by measuring the so-called *wall clock time*, i.e. the total time passed during the simulation. It is accessible as `MPI_Wtime()` routine in `MPI2.0`. In the convergence plots we use the *cumulative wall clock time* (obtained by multiplying wall clock time by number of cores); this way the dependence on the used number of cores is reduced allowing for straightforward comparison of the runtimes.

Error estimator. Since the solution is a random field, the discretization error is a random quantity as well. For convergence analysis we therefore compute a statistical estimator by averaging estimated discretization errors from several independent runs. We will compute the errors in (3.2) and (4.2) by approximating $L^2(\Omega; L^1(\mathbb{R}^d))$ norm with MC quadrature. Let $\{\mathbf{U}_k\}_{k=1,\ldots,K}$ be a sequence of independent approximate solutions obtained by running MC-FVM or MLMC-FVM solver K times corresponding to K realizations of the stochastic space. Denoting the reference solution by $\mathbf{U}_{\mathrm{ref}}$, the $L^2(\Omega; L^1(\mathbb{R}^d))$-based relative error estimator is defined as in [20],

$$(6.4) \qquad \mathcal{RE} = \sqrt{\sum_{k=1}^{K} (\mathcal{RE}_k)^2/K}, \qquad \mathcal{RE}_k = 100 \times \frac{\|\mathbf{U}_{\mathrm{ref}} - \mathbf{U}_k\|_{\ell^1(\mathcal{T})}}{\|\mathbf{U}_{\mathrm{ref}}\|_{\ell^1(\mathcal{T})}}.$$

The extensive analysis for the appropriate choice of K is conducted in [20]; we choose $K = 30$ which was shown to be more than sufficient.
We compare the following schemes:

MC2	Monte Carlo with 2nd order FVM scheme	$M = \mathcal{O}(\Delta x^{-2})$,
MLMC2	Multi-Level MC with 2nd order FVM scheme	$M_\ell = M_L 4^{(L-\ell)}$.

The parameter M_L corresponds to the number of samples in the finest level and can be freely chosen. Analysis in [20] suggests that $M_L = 16$ is a reasonable choice and we use this value in our simulations.

Dashed lines in Figure 6 (and all subsequent figures) indicate *expected* convergence rate slopes obtained by the theory in [20] for the *scalar* case (see (3.2) and (4.2)). We expect them to coincide with the observed convergence rates for *systems* of balance laws and in this particular case they are actually very similar. Our findings coincide with the results published in [20, 21] confirming the robustness of the implementation.

In Figure 7, we show convergence plots for variance. Both figures show that MLMC methods are two orders of magnitude faster than MC methods in computing the mean as well as in computing the variance. This numerical experiment clearly illustrates the superiority of the MLMC algorithm over the MC algorithm.

6.2. **2-D numerical experiments: Random perturbation of lake at rest.** We consider (1.2) with $d = 2$ in a computational domain $\mathbf{D} = [0, 2] \times [0, 2]$, and
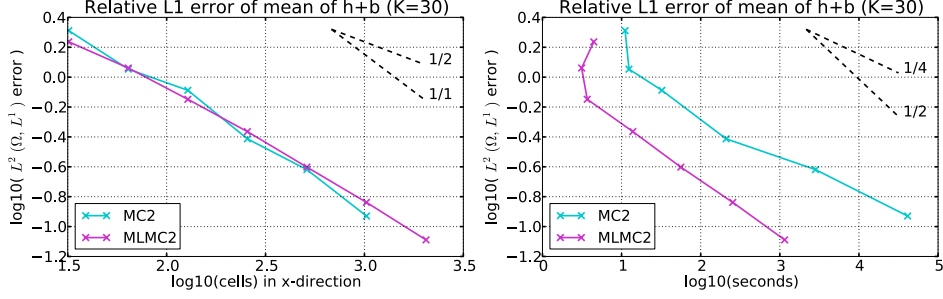
FIGURE 6. Convergence of estimated mean for 1-D dambreak (6.1). Both MLMC2 and MC2 give similar errors for the same spatial resolution. However, there is a significant difference in runtime: MLMC methods are 2 orders of magnitude faster than MC.
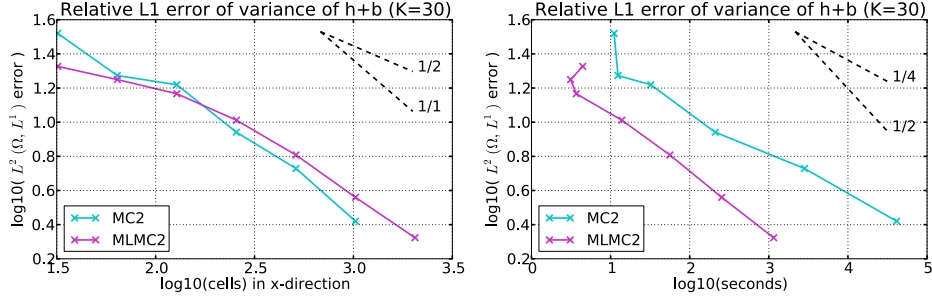

FIGURE 7. Convergence of estimated variance for 1-D dambreak (6.1). MLMC methods are 2 orders of magnitude faster than MC.

investigate the evolution of an uncertain perturbation of the lake at rest coupled with outflow boundary conditions.

The uncertain bottom topography $b(x, \omega)$ is represented in terms of the nodal, bivariate hierarchical basis (5.11) - (5.12) with random amplitudes. Notice that, formally, this bilinear basis can be obtained by tensorizing the univariate Schauder basis of $C^0([0, 2])$. Notice also that we used in the present study only *isotropically supported* product functions. The bottom topography was resolved to 6 levels (i.e. $\bar{L} = 5$, $\bar{\ell} = 0, \ldots, 5$) where coefficients $b^{\bar{\ell}}_{k,k'}(\omega), \check{b}^{\bar{\ell}}_{k,k'}(\omega), \hat{b}^{\bar{\ell}}_{k,k'}(\omega)$ are given by mean values $\mu^{\bar{\ell}}_{k,k'}, \check{\mu}^{\bar{\ell}}_{k,k'}, \hat{\mu}^{\bar{\ell}}_{k,k'}$, respectively. Mean values are then perturbed by independent uniformly distributed centered random variables with decaying variances,

$$b^{\bar{\ell}}_{k,k'}(\omega) = \mu^{\bar{\ell}}_{k,k'} + Y^{\bar{\ell}}_{k,k'}(\omega) \sim \frac{2}{5}\mathcal{U}(-\varepsilon_{\bar{\ell}}, \varepsilon_{\bar{\ell}}),$$

(6.5)
$$\check{b}^{\bar{\ell}}_{k,k'}(\omega) = \check{\mu}^{\bar{\ell}}_{k,k'} + \check{Y}^{\bar{\ell}}_{k,k'}(\omega) \sim \frac{2}{5}\mathcal{U}(-\varepsilon_{\bar{\ell}}, \varepsilon_{\bar{\ell}}),$$

$$\hat{b}^{\bar{\ell}}_{k,k'}(\omega) = \hat{\mu}^{\bar{\ell}}_{k,k'} + \hat{Y}^{\bar{\ell}}_{k,k'}(\omega) \sim \frac{2}{5}\mathcal{U}(-\varepsilon_{\bar{\ell}}, \varepsilon_{\bar{\ell}}),$$

where all coefficients $\mu^{\bar{\ell}}_{k,k'}, \check{\mu}^{\bar{\ell}}_{k,k'}, \hat{\mu}^{\bar{\ell}}_{k,k'}$ are zero except

(6.6)
$$\mu^3_{2,2} = 0.4, \quad \mu^4_{6,6} = -0.32, \quad \mu^5_{11,11} = 0.12,$$

and the magnitudes of the uncertainties are set by $\varepsilon_0 = 0$, $\varepsilon_{\bar{\ell}} = 2^{-\bar{\ell}}$, $\forall \bar{\ell} \geq 1$.

A realization of the uncertain bottom topography and the corresponding mean and variance are shown in figure 8.
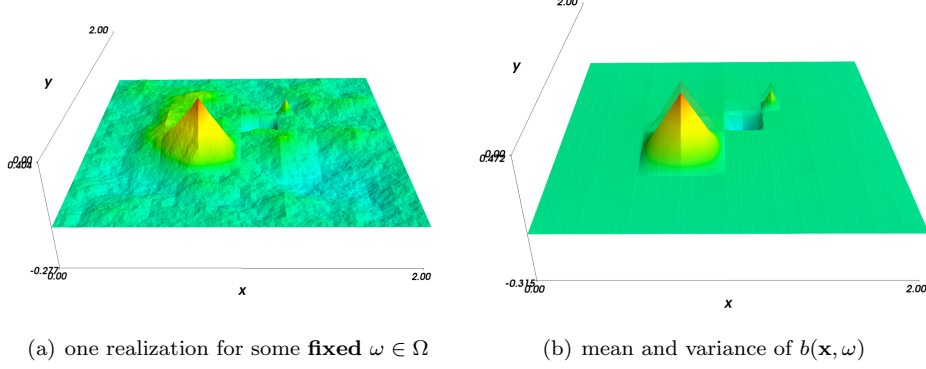

(a) one realization for some **fixed** $\omega \in \Omega$ (b) mean and variance of $b(\mathbf{x}, \omega)$

FIGURE 8. Uncertain bottom topography (6.5) with 9 hierarchical levels ($\bar{L} = 8$).

Next, we consider the initial data $\mathbf{U}_0$ to be a *random* perturbation of a lake-at-rest. Let $Y \sim \frac{1}{50} + \frac{1}{100}\mathcal{U}(-1,1)$ be a random variable independent of $\{Y_{k,k'}^{\bar{\ell}}, \check{Y}_{k,k'}^{\bar{\ell}}, \hat{Y}_{k,k'}^{\bar{\ell}}\}$. An initial perturbation around $\mathbf{x}_0 = (x_0, y_0) = (1.0, 0.7)$ with a radius $r = \frac{1}{10}$ reads

$$(6.7) \qquad h_0(x, y, \omega) = \begin{cases} 1.0 + Y(\omega) - b(x, y, \omega) & \text{if} \quad |\mathbf{x} - \mathbf{x}_0| < r, \\ 1.0 - b(x, y, \omega) & \text{if} \quad |\mathbf{x} - \mathbf{x}_0| > r, \end{cases}$$

with $b(x, \omega)$ as defined in (6.5) and the initial layer velocities set to zero, i.e.

$$(6.8) \qquad\qquad \{u_0(x, y, \omega), \ v_0(x, y, \omega)\} = \{0.0, \ 0.0\}.$$

Note, that here we have even more sources of uncertainty $((2^5 - 1)^2 - 1 = 962)$ than in one-dimensional case (6.2).

Reference solutions, computed with the second-order entropy stable TeCNO scheme [14, 15] at time $T = 0.1$ is depicted in Figure 9. The results are computed on 9 nested levels of resolution ($L = 8$) with the finest resolution being on a 4096×4096 mesh and with time steps reduced accordingly in order to maintain the same CFL constant over all discretization levels. The simulation is run on 2044 cores and 16 samples are taken for the finest mesh resolution.

The above problem is quite involved due to large number of sources of uncertainty as well as the underlying difficulty of simulating small perturbations of steady states. The reference solution show that the wave (in mean) spreads out of the initial source. The variance is distributed in a non-linear and complicated manner with large amount of variance corresponding to the uncertainties in the bottom topography.

6.2.1. *Numerical convergence analysis.* We investigate convergence of error vs. work in Figure 10 and Figure 11. Here we use the MLMC-FVM simulation from Figure 9 with 9 levels of resolution with the finest resolution being on a 4096×4096 mesh as the reference solution $\mathbf{U}_{\text{ref}}$. The error in the mean field converges at expected

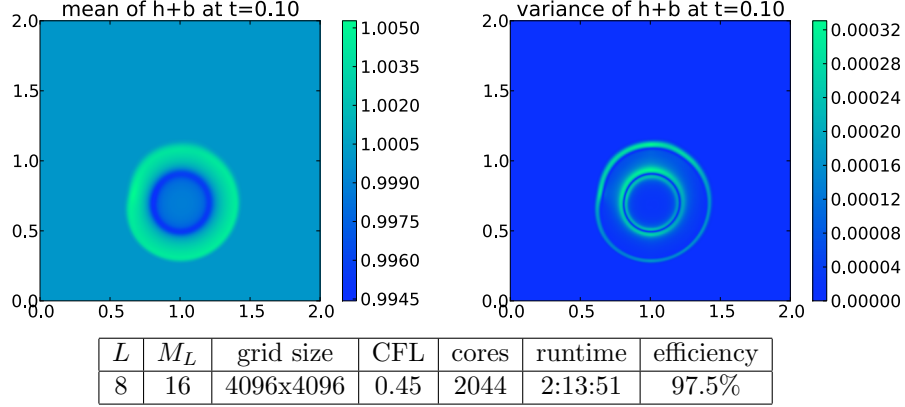| L	M_L	grid size	CFL	cores	runtime	efficiency
8	16	4096x4096	0.45	2044	2:13:51	97.5%

FIGURE 9. Reference solution for perturbed steady-state (6.7) using MLMC-FVM with ES flux (2.2). Initial perturbation evolves into asymmetric ribbon wave with uncertain amplitude.

rates. At comparable numerical resolution and accuracy, the MLMC2 is about two orders of magnitude faster than the MC2 method for this problem. We observe a slight deterioration in the estimated convergence rates for the variance. This could well be a pre-asymptotic effect. Again, the MLMC2 appears to be slightly faster than the corresponding MC2 method in delivering variance estimates of comparable numerical accuracy.
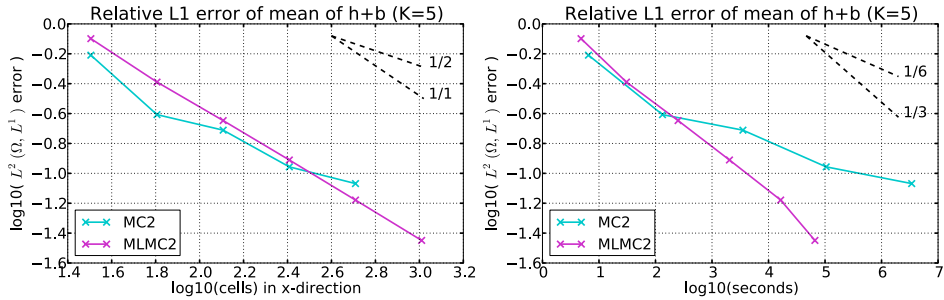
FIGURE 10. Convergence of estimated mean in the 2-D simulation (6.7). MLMC methods are 3 orders of magnitude faster than MC.

6.2.2. *Speed up due to hierarchical topology representation.* We test the gain in efficiency due to the multi-level hierarchical representation of the uncertain bottom topography (5.11) by comparing with a simulation that uses the classic MLMC algorithm of [21]. In other words, the MLMC2 (full) simulation uses the underlying bottom topography (at the resolution of the underlying topography mesh) for all shallow water samples. In particular, simulations at the coarsest level of the FVM mesh use the topography at the finest level of the underlying topography mesh. We compare MLMC2 (full) with MLMC2 (truncated) which uses the representation (5.11) on the perturbations of lake at rest steady state problem in Figure 12. As suggested by the theory of section 5, the two methods should lead to an identical
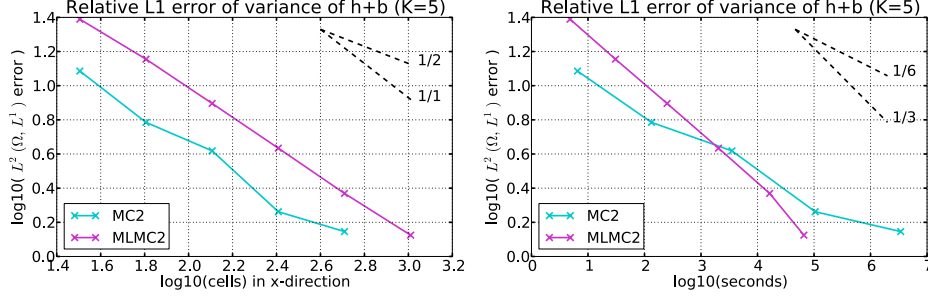
FIGURE 11. Convergence of estimated variance in the 2-D simulation (6.7). MLMC methods are asymptotically faster than MC.

order of the error for a given space-time resolution. We verify this in Figure 12. On the other hand, the MLMC2 (truncated) is at least an order of magnitude faster than the MLMC2 (full) showing that the multi-level representation of the uncertain bottom topography really provides a significant gain in efficiency.
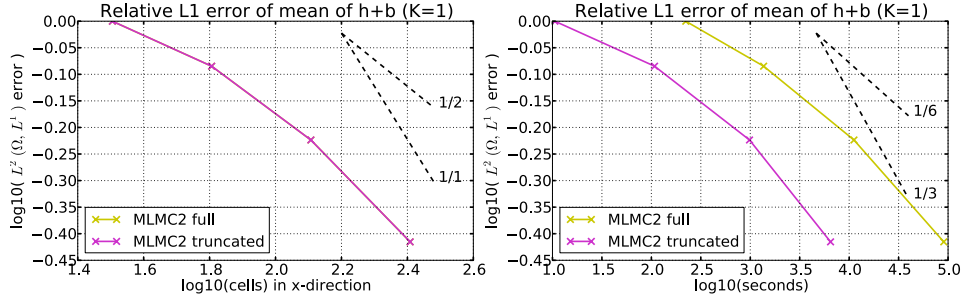

FIGURE 12. Convergence of estimated mean for 2-D steady-state (6.7) with full ($\bar{L} = 8$) and truncated ($\ell + 1$) number of levels in the hierarchical representation (6.5) of bottom topography. For a given mesh resolution, both estimators coincide, verifying statements in Proposition 5.2. The implementation with the truncated number of levels is more than 10 times faster on a mesh of 256×256 cells.

6.2.3. *Efficiency of parallelization.* The parallelization for the MLMC algorithm is performed using the static load balancing procedure described in [24]. In Figure 13 we show the efficiency of parallelization defined as a fraction of *simulation time* (excluding time spent for `MPI` communications and idling) over *wall clock time*,

$$(6.9) \qquad \text{efficiency} := 1 - \frac{(\text{total clock time of all MPI routines})}{(\#\text{cores}) \times (\text{wall clock time})}.$$

We observe that runs for convergence analysis in Figures 10 and 11 were quite efficient and most of the time is spent computing rather than communicating or waiting. Furthermore, the strong scaling up to around 4000 cores is verified in Figure 14 for this problem. Similarly, Figure 15 verifies a weak scaling up to a similar number of cores. The algorithm is expected to scale up to a much larger number of cores.
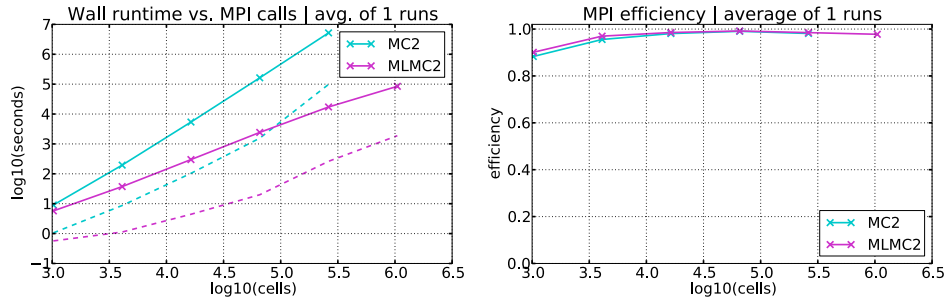
FIGURE 13. MPI overhead. For large problems (more than 64 cells in each dimension) efficiency of parallelization is $\approx 98\%$.
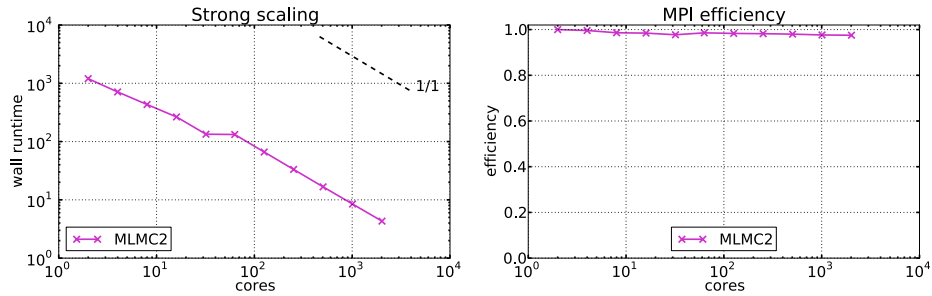

FIGURE 14. Strong scaling of MLMC-FVM for steady-state perturbation problem (6.5) - (6.7) is verified up to ~ 4000 cores.
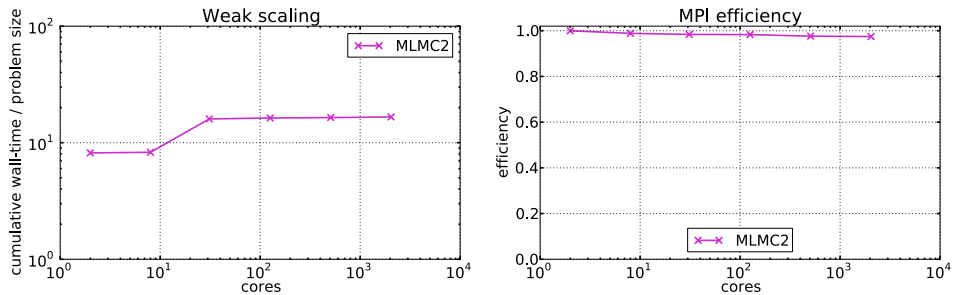

FIGURE 15. Weak scaling of MLMC-FVM for steady-state perturbation problem (6.5) - (6.7) is verified up to ~ 4000 cores.

7. CONCLUSION

The shallow water equations model many phenomena in geophysics and oceanography. Inputs to these equations such as the initial data and the bottom topography are *uncertain* due to inherent uncertainties in the measurement process. Here, we model the uncertainty in terms of random fields and design efficient statistical sampling algorithms to efficiently compute the uncertainty in the solution.

In particular, we extend our Multi-Level Monte Carlo (MLMC) algorithm of [20, 21] to the case of systems of balance laws like the shallow water equations with

random field source terms. A key advantage of the MLMC algorithm is that it is *non-intrusive.* Any deterministic solver can be used in the algorithm. In this paper, we employ the *well-balanced energy stable* FVM solver of [14] together with the MLMC method. This solver is good at performing long time integration and allows stable simulation of small perturbations of the ocean at rest steady state.

A key challenge in designing efficient MLMC algorithms for the shallow water equations lies in the representation of the bottom topography. In practical applications [13], the uncertainty in the bottom topography is represented as (possibly uncorrelated) nodal values. Apart from increasing the number of sources of uncertainty, this representation makes the MLMC algorithm slow as a large number of samples (from the underlying topography mesh) have to be drawn, even at the coarsest FVM meshes. Here, we propose a *novel multi-level hierarchical representation* of the bottom topography and show that only samples of bottom topography from (even) coarser meshes have to drawn, when the flow is being simulated on a FVM mesh at a certain level of resolution.

Extensive numerical experiments (in one and two space dimensions) show that the MLMC algorithm is consistently (by 2 orders of magnitude on meshes appearing in engineering practice) faster than the standard MC method. Furthermore, the multi-level hierarchical bottom topography representation leads to an order of magnitude speed up over the standard MLMC algorithm, at least in two space dimensions and for engineering accuracies of point-wise relative errors of 1% in estimates of mean and variance. Coupled with these desirable properties, a robust FVM solver and an efficient parallelization strategy such as the one described in [24], the MLMC-FVM method allows to simulate realistic uncertain flows in the presence of spatially heterogeneous random field source terms. At the same time, the proposed multi-level representation methodology for the uncertain bottom topography allows MLMC-FVM treatment of general balance laws with random source terms.

References

[1] ALSVID. Available from http://folk.uio.no/mcmurry/amhd.
[2] ALSVID-UQ. Available from http://mlmc.origo.ethz.ch/.
[3] R. Abgrall. *A simple, flexible and generic deterministic approach to uncertainty quantification in non-linear problems.* Rapport de Recherche, INRIA, 2007.
[4] A. Barth, Ch. Schwab and N. Zollinger. *Multilevel MC Method for Elliptic PDEs with Stochastic Coefficients.* Num. Math., 2011 (to appear).
[5] P. D. Bates, S. N. Lane and R. I. Ferguson. *Parametrization, Validation and Uncertainty analysis of CFD models of fluvial and flood hydraulics in natural environments.* in Computational Fluid Dynamics: Applications in environmental hydraulics, John Wiley and sons, 2005, 193-212.
[6] Q. Y. Chen, D. Gottlieb and J. S. Hesthaven. *Uncertainty analysis for steady flow in a dual throat nozzle.* J. Comput. Phys, **204**:378-398, 2005.
[7] Constantine M. Dafermos. *Hyperbolic Conservation Laws in Continuum Physics (2nd Ed.).* Springer Verlag (2005).
[8] G. DalMaso, P. LeFloch and F. Murat. *Definition and weak stability of nonconservative products.* J. Math. Pures. Appl., **74**:483-548, 1995.
[9] M. Giles. *Improved multilevel Monte Carlo convergence using the Milstein scheme.* Preprint NA-06/22, Oxford computing lab, Oxford, U.K, 2006.
[10] M. Giles. *Multilevel Monte Carlo path simulation.* Oper. Res., **56**:607-617, 2008.
[11] Edwige Godlewski and Pierre A. Raviart. *Hyperbolic Systems of Conservation Laws.* Mathematiques et Applications, Ellipses Publ., Paris (1991).

[12] S. Heinrich. *Multilevel Monte Carlo methods.* Large-scale scientific computing, Third international conference LSSC 2001, Sozopol, Bulgaria, 2001, Lecture Notes in Computer Science, Vol **2170**, Springer Verlag (2001), pp. 58-67.

[13] P. F. Fisher and N. J. Tate. *Causes and consequences of error in digital elevation models.* Prog. in Phy. Geography, 30 (4), 2006, 467-489.

[14] U.S. Fjordholm, S. Mishra, and E. Tadmor. *Well-balanced, energy stable schemes for the shallow water equations with varying topology. J. Comput. Phys,* 230, 5587-5609, 2011.

[15] U. S. Fjordholm, S. Mishra and E. Tadmor. Arbitrary order accurate essentially non-oscillatory entropy stable schemes for systems of conservation laws. *Research Report N. 2011–39*, Seminar für Angewandte Mathmatik ETH Zürich, 2011.

[16] R.A. LeVeque. *Numerical Solution of Hyperbolic Conservation Laws.* Cambridge Univ. Press 2005.

[17] R. LeVeque, D. George and M. Berger. Tsunami modeling with adaptively refined finite volume methods. *Acta Numerica*, 20, 211-289, 2011.

[18] G. Lin, C.H. Su and G. E. Karniadakis. *The stochastic piston problem.* PNAS **101**:15840-15845, 2004.

[19] X. Ma and N. Zabaras. *An adaptive hierarchical sparse grid collocation algorithm for the solution of stochastic differential equations.* J. Comp. Phys, 228, 3084-3113, 2009.

[20] S. Mishra and Ch. Schwab. *Sparse tensor multi-level Monte Carlo Finite Volume Methods for hyperbolic conservation laws with random initial data.* Math. Comp., 2011 (to appear). Also available from `http://www.sam.math.ethz.ch/reports/2010/24`.

[21] S. Mishra, Ch. Schwab and J. Šukys. *Multi-level Monte Carlo finite volume methods for non-linear systems of conservation laws in multi-dimensions.* J. Comp. Phys., 2011 (in review). Available from: `http://www.sam.math.ethz.ch/reports/2011/02`.

[22] G. Poette, B . Després and D. Lucor. *Uncertainty quantification for systems of conservation laws.* J. Comput. Phys. **228**:2443-2467, 2009.

[23] P. L. Roe. *Approximate Riemann solvers, parameter vectors, and difference schemes.* J. Comput. Phys., **226**:250-258, 1981.

[24] J. Šukys, S. Mishra, and Ch. Schwab. *Static load balancing for multi-level Monte Carlo finite volume solvers.* Parallel Processing and Applied Mathematics 9th International Conference, PPAM 2011, Torun, Poland, 2011 (to appear). Available from: `http://www.sam.math.ethz.ch/reports/2011/32`.

[25] J. Tryoen, O. Le Maitre, M. Ndjinga and A. Ern. *Intrusive projection methods with upwinding for uncertain non-linear hyperbolic systems.* Preprint, 2010.

[26] X. Wan and G. E. Karniadakis. *Long-term behaviour of polynomial chaos in stochastic flow simulations.* Comput. Meth. Appl. Mech. Engg. **195**:5582-5596, 2006.

[27] G.B. Whitham *Linear and nonlinear waves.* John Wiley and Sons., New York, 1999, 636 pp.

[28] J. A. S. Witteveen, A. Loeven, H. Bijl *An adaptive stochastic finite element approach based on Newton-Cotes quadrature in simplex elements.* Comput. Fluids, 38, 1270-1288, 2009.

[29] D. Xiu and J. S. Hesthaven. *High-order collocation methods for differential equations with random inputs.* SIAM J. Sci. Comput., 27, 1118-1139, 2005.

[30] *Brutus,* ETH Zürich, `de.wikipedia.org/wiki/Brutus_(Cluster)`.

[31] *Cray XE6,* Swiss National Supercomputing Center (CSCS), Manno, `www.cscs.ch`.

[32] *MPI: A Message-Passing Interface Standard.* Version 2.2, 2009, available from: `http://www.mpi-forum.org/docs/mpi-2.2/mpi22-report.pdf`.

[33] *Open MPI: Open Source High Performance Computing.* Available from `http://www.open-mpi.org/`.

(Siddhartha Mishra)
SEMINAR FOR APPLIED MATHEMATICS
ETH
HG G. 57.2,
RÄMISTRASSE 101, ZÜRICH, SWITZERLAND AND
CENTER OF MATHEMATICS FOR APPLICATIONS,
UNIVERSITY OF OSLO,
P.O.BOX - 0316, BLINDERN, OSLO, NORWAY.
    E-mail address: smishra@sam.math.ethz.ch

(Christoph Schwab)
SEMINAR FOR APPLIED MATHEMATICS
ETH
HG G. 57.1,
RÄMISTRASSE 101, ZÜRICH, SWITZERLAND.
    E-mail address: christoph.schwab@sam.math.ethz.ch

(Jonas Šukys)
SEMINAR FOR APPLIED MATHEMATICS
ETH
HG G. 62.1,
RÄMISTRASSE 101, ZÜRICH, SWITZERLAND.
    E-mail address: jonas.sukys@sam.math.ethz.ch

Research Reports