

No. 630

April 2020

**Enriched Galerkin method for
the shallow-water equations**

**M. Hauck, V. Aizinger, F. Frank,
H. Hajduk, A. Rupp**

ISSN: 2190-1767

Enriched Galerkin method for the shallow-water equations

Moritz Hauck^a, Vadym Aizinger^b, Florian Frank^a, Hennes Hajduk^c, Andreas Rupp^d

^aDepartment of Mathematics, Friedrich-Alexander University Erlangen-Nürnberg, Cauerstraße 11, 91058 Erlangen, Germany

^bChair of Scientific Computing, University of Bayreuth, Universitätsstraße 30, 95447 Bayreuth, Germany

^cInstitute of Applied Mathematics (LS III), TU Dortmund University, Vogelpothsweg 87, 44227 Dortmund, Germany

^dInterdisciplinary Center for Scientific Computing (IWR), Heidelberg University, Im Neuenheimer Feld 205, 69120 Heidelberg, Germany

Abstract

This work presents an enriched Galerkin (EG) discretization for the two-dimensional shallow-water equations. The EG finite element spaces are obtained by extending the approximation spaces of the classical finite elements by discontinuous functions supported on elements. The simplest EG space is constructed by enriching the piecewise linear continuous Galerkin space with discontinuous, element-wise constant functions. Similarly to discontinuous Galerkin (DG) discretizations, the EG scheme is locally conservative, while, in multiple space dimensions, the EG space is significantly smaller than that of the DG method. This implies a lower number of degrees of freedom compared to the DG method. The EG discretization presented for the shallow-water equations is well-balanced, in the sense that it preserves lake-at-rest configurations. We evaluate the method's robustness and accuracy using various analytical and realistic benchmarks and compare the results to those obtained using the DG method. Finally, we briefly discuss implementation aspects of the EG method within our MATLAB/GNU Octave framework FESTUNG.

Keywords: Enriched Galerkin, Finite elements, Shallow-water equations, discontinuous Galerkin, Local conservation, Ocean modeling

1. Introduction

The two dimensional shallow-water equations (SWE) are used for a wide range of applications in environmental and hydraulic engineering, oceanography, and many other areas. They are discretized on computational domains that can be very large and often feature complex geometries; therefore, the numerical schemes must be computationally efficient and robust. The nonlinearity and hyperbolic character of the SWE system constitute additional challenges for designing discretizations and solution algorithms, while other application-specific aspects such as local conservation of unknown quantities and well-balancedness represent further desirable properties (see [1] for a brief overview of key requirements for SWE models).

The aforementioned issues led to a large number of studies dedicated to the development, analysis, and practical evaluation of various numerical techniques for solving the SWE. Historically, the first models used finite differences on structured grids, but, with the emergence of unstructured-mesh models (e.g. TELEMAC [2] or ADCIRC [3]), finite elements and finite volumes became the de-facto standard. A big advantage of the finite element approach is its potential to naturally accommodate higher-order approximations on unstructured meshes; in this vein, various discretizations based on the continuous (CG) and discontinuous Galerkin (DG) approximation spaces (or mixtures of both) have been described and compared in the literature [4, 5]. The results of these comparisons can be summarized (in a somewhat oversimplified fashion) as follows:

- Using continuous (e.g. piecewise linear) finite elements for elevation and velocity is computationally very efficient (at least the low-order schemes) but tends to have stability issues usually represented by spurious elevation or velocity modes that arise from the LBB condition or from too large elevation spaces;

*Corresponding author

Email address: vadym.aizinger@uni-bayreuth.de (Vadym Aizinger)

- Using discontinuous Galerkin (e.g. piecewise linear) spaces for both unknowns is robust and needs no additional stabilization but is computationally more expensive (up to a factor of four to five in serial execution [6]). In addition, discontinuous Galerkin schemes are locally conservative, better suited for adaptive and non-conforming meshes, and can partially offset their higher computational costs by more efficient parallel scaling [7];
- some combinations of continuous and discontinuous spaces such as the lowest order Raviart-Thomas spaces tend to perform better [4], but are rather complicated to extend to higher orders on triangular meshes.

The idea of the enriched Galerkin (EG) method is to enhance the approximation space of the continuous finite elements by adding element-local discontinuous functions and, by using the solution procedure nearly identical to that of the DG method (i.e. integrals of edge fluxes, Riemann solvers, etc.), produce a robust, locally conservative discretization with significantly fewer unknowns than a DG discretization of the same order.

In its original form, the EG method (also considered in this paper) adds a piecewise constant DG component to a continuous piecewise linear or multilinear CG discretization. This method was introduced for the linear advection–diffusion–reaction equation in [8] and was proved there to be stable and converge at the same rate as the discontinuous Galerkin method for piecewise linear discretization. The piecewise constant enrichment of the linear continuous Galerkin approximation is intrinsically stable and provides the local conservation property as was shown in [9].

Since then, the EG method has been generalized to the classical continuous Galerkin function space of the polynomial order k augmented by discontinuous, element-wise constant functions [9, 10] and even to arbitrary enrichment with polynomials of degree m with $-1 \leq m \leq k$ (where enrichment of degree $m = -1$ represents no enrichment at all) as discussed in [11]. Arbitrary enrichment allows to consider EG as generalization of both CG and DG since EG uses the same bilinear and linear forms as DG.

Thus, EG inherits many advantages of DG. However, in multiple dimensions, EG has fewer degrees of freedom (DOF) than the DG method if $m < k$. The standard EG method (with $m = 0$) has been developed to solve general elliptic and parabolic problems with dynamic mesh adaptivity [12, 13, 14, 15] and was extended to address multiphase fluid flow problems [15]. Recently, the EG method has been applied to solve the nonlinear poroelastic problem [16, 17], and its performance has been compared to other two- and three-field formulation methods [18].

In the context of convection-dominated problems, additional stabilization techniques are often required. Classical stabilizations include the streamline upwind Petrov-Galerkin (SUPG) method, weighted essentially non oscillatory (WENO) schemes, as well as provably bound-preserving alternatives (e.g. [19, 20, 21, 22, 23]). Popular methods designed particularly for DG discretizations include the edge-based Barth-Jespersen limiter [24], and its vertex-based counterparts [25, 26]. Limiters for an EG discretization of the linear advection equation have recently been proposed in [27]. The approach therein deviates from classical DG slope limiters but rather fits in the framework of *algebraic flux correction* [28], which only recently has been extended to the DG setting [29, 30]. Numerical solutions based on the methods in [27] can be proven to satisfy discrete maximum principles under CFL-like time step restrictions, which makes the approach superior to geometrical slope limiting.

The main focus of the present work is to formulate and evaluate an EG scheme for the SWE and to compare the quality of the EG and DG discretizations. The method is implemented in the FESTUNG framework [31, 32, 33, 34] by modifying our DG implementation for the SWE presented in [35, 36]. The same scheme was initially introduced in our UTBEST solver [37, 38] and later extended to three dimensions in UTBEST3D [39, 40].

The paper is structured as follows. The mathematical model and its discretization using the EG method are presented in Sec. 2, a brief description of the implementation using our MATLAB/GNU Octave framework FESTUNG is the subject of Sec. 3. In Sec 4, we demonstrate the accuracy and robustness of our EG scheme using an analytical convergence test, a supercritical flow example with discontinuous solution, and a realistic tidal flow scenario for Bahamas islands. A short Conclusions section wraps up this work.

2. EG formulation for the SWE

2.1. Governing equations

The SWE in conservative form are given by

$$\partial_t \xi + \nabla \cdot \mathbf{q} = 0, \quad (1a)$$

$$\partial_t \mathbf{q} + \nabla \cdot (\mathbf{q}\mathbf{q}^\top / H) + \tau_{\text{bf}} \mathbf{q} + \begin{pmatrix} 0 & -f_c \\ f_c & 0 \end{pmatrix} \mathbf{q} + gH \nabla \xi = \mathbf{F}. \quad (1b)$$

They are considered on a two-dimensional, polygonally-bounded domain Ω and finite time interval (t_0, t_{end}) . By ξ , we denote the free surface elevation of the water body with respect to a certain zero level (e.g., the mean sea level). The quantity $H = \xi - z_b$ represents the total fluid depth with z_b denoting the bathymetry. $\mathbf{q} := (U, V)^\top$ is the depth integrated horizontal velocity field, f_c the Coriolis coefficient, g the gravitational acceleration, and τ_{bf} the bottom friction coefficient. Wind stress, the atmospheric pressure gradient, and tidal potential are combined in the body force term $\mathbf{F} := (F_x, F_y)^\top$.

Defining $\mathbf{c} := (\xi, U, V)^\top$, system (1) can be rewritten in the following compact form:

$$\partial_t \mathbf{c} + \nabla \cdot \mathbf{A}(\mathbf{c}) = \mathbf{r}(\mathbf{c}) \quad (2)$$

with

$$\mathbf{A}(\mathbf{c}) = \begin{pmatrix} U & V \\ \frac{U^2}{\xi - z_b} + g\xi \left(\frac{\xi}{2} - z_b\right) & \frac{UV}{\xi - z_b} \\ \frac{UV}{\xi - z_b} & \frac{V^2}{\xi - z_b} + g\xi \left(\frac{\xi}{2} - z_b\right) \end{pmatrix} \quad \text{and} \quad \mathbf{r}(\mathbf{c}) = \begin{pmatrix} 0 \\ -\tau_{\text{bf}}U + f_c V - g\xi \partial_x z_b + F_x \\ -\tau_{\text{bf}}V - f_c U - g\xi \partial_y z_b + F_y \end{pmatrix}. \quad (3)$$

65 In this work, we use several types of boundary conditions for the SWE (1). Following the standard approach, interior values are used as boundary values at parts of the boundary, on which the corresponding unknowns are not prescribed. By $\hat{\cdot}$, we denote prescribed boundary values of the respective unknowns. The following types of boundary conditions are used in this work:

Dirichlet boundary: Here, all unknowns are specified:

$$\xi = \hat{\xi}, \quad \mathbf{q} = \hat{\mathbf{q}}. \quad (4)$$

Land boundary: Denoting by \mathbf{n} the exterior unit normal to $\partial\Omega$, we set the normal flux to zero:

$$\mathbf{q} \cdot \mathbf{n} = 0. \quad (5)$$

Open sea boundary: Only the free surface elevation is prescribed for this boundary type:

$$\xi = \hat{\xi}. \quad (6)$$

Radiation boundary: No unknowns are specified, which corresponds to a free outflow condition.

Finally, we need to prescribe initial conditions for the elevation and depth integrated velocity

$$\xi(\mathbf{x}, 0) = \xi_0(\mathbf{x}), \quad \mathbf{q}(\mathbf{x}, 0) = \mathbf{q}_0(\mathbf{x}). \quad (7)$$

70 2.2. Enriched Galerkin discretization

Let $\{\mathcal{T}_\Delta\}_{\Delta>0}$ be a simplicial, shape-regular, quasi uniform, geometrically conformal family of triangulations of $\Omega \subset \mathbb{R}^2$ with $\#T$ denoting the total number of elements of \mathcal{T}_Δ . We obtain the local variational formulation of system (2) by multiplying with smooth test functions $\phi \in C^\infty(\Omega)^3$ and integrating by parts on each element $T \in \mathcal{T}_\Delta$ yielding

$$(\partial_t \mathbf{c}, \phi)_T - (\mathbf{A}(\mathbf{c}), \nabla \phi)_T + \langle \mathbf{A}(\mathbf{c}) \cdot \mathbf{n}, \phi \rangle_{\partial T} = (\mathbf{r}(\mathbf{c}), \phi)_T, \quad (8)$$

where we write $(\cdot, \cdot)_T$ and $\langle \cdot, \cdot \rangle_{\partial T}$ for the L^2 -scalar products on elements and their boundaries, and denote by $\mathbf{n} = (n_x, n_y)^\top$ an exterior unit normal to ∂T .

Defining the *broken polynomial spaces* of order $m \in \mathbb{N}_0$ as

$$\mathbb{P}_m(\mathcal{T}_\Delta) := \left\{ v \in L^2(\Omega) : v|_T \text{ is a polynomial of degree at most } m, \forall T \in \mathcal{T}_\Delta \right\} \quad \text{with} \quad \mathbb{P}_{-1}(\mathcal{T}_\Delta) := \{0\}$$

we can modify them to obtain the EG *test* and *trial spaces*

$$\mathbb{P}_{k,m}(\mathcal{T}_\Delta) := \left(\mathbb{P}_k(\mathcal{T}_\Delta) \cap C(\Omega) \right) + \mathbb{P}_m(\mathcal{T}_\Delta) \quad (9)$$

for integers $-1 \leq m \leq k$, $k > 0$. Thus, obviously, $\mathbb{P}_m(\mathcal{T}_\Delta) \subset \mathbb{P}_{k,m}(\mathcal{T}_\Delta) \subset \mathbb{P}_k(\mathcal{T}_\Delta)$. Here, '+' denotes the sum of subspaces which is not a direct sum if $m \neq -1$. Examples of spaces are given in Fig. 1. From (9) it follows

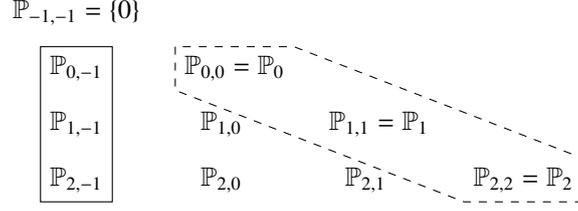


Figure 1: Example of EG spaces: CG spaces are marked by a solid line, DG spaces by a dashed line. EG spaces of interest are those lying in-between.

immediately that

$$\begin{aligned} \dim \mathbb{P}_{k,m}(\mathcal{T}_\Delta) &= \dim(\mathbb{P}_k(\mathcal{T}_\Delta) \cap C(\Omega)) + \dim \mathbb{P}_m(\mathcal{T}_\Delta) - \dim((\mathbb{P}_k(\mathcal{T}_\Delta) \cap C(\Omega)) \cap \mathbb{P}_m(\mathcal{T}_\Delta)) \\ &= \dim \mathbb{P}_{k,-1}(\mathcal{T}_\Delta) + \dim \mathbb{P}_{m,m}(\mathcal{T}_\Delta) - \dim \mathbb{P}_{m,-1}(\mathcal{T}_\Delta). \end{aligned} \quad (10)$$

This dimension formula will become handy in Sec. 3 for determining EG shape functions. For more details about these spaces and their properties, the reader is referred to [11, Sect. 3].

To obtain the semi-discrete EG formulation of (8), \mathbf{c} and ϕ are replaced by their discrete counterparts $\mathbf{c}_\Delta, \phi_\Delta \in \mathbb{P}_{k,m}(\mathcal{T}_\Delta)^3$. Since the values of a discontinuous function are not unique on element edges, we replace the boundary term $\mathbf{A}(\mathbf{c}) \cdot \mathbf{n}$ by a numerical flux $\hat{\mathbf{A}}(\mathbf{c}_\Delta, \mathbf{c}_\Delta^+, \mathbf{n})$ that depends on the discontinuous values of the solution on element T (without superscript) and its edge neighbor (superscript $+$). On domain boundaries, the specified boundary values of free surface elevation and velocity are utilized in place of \mathbf{c}_Δ^+ for the flux computation.

In conclusion, summing up over all elements $T \in \mathcal{T}_\Delta$, yields

$$\sum_{T \in \mathcal{T}_\Delta} (\partial_t \mathbf{c}_\Delta, \phi_\Delta)_T - \sum_{T \in \mathcal{T}_\Delta} ((\mathbf{A}(\mathbf{c}_\Delta), \nabla \phi_\Delta)_T - \langle \hat{\mathbf{A}}(\mathbf{c}_\Delta, \mathbf{c}_\Delta^+, \mathbf{n}), \phi_\Delta \rangle_{\partial T}) = \sum_{T \in \mathcal{T}_\Delta} (\mathbf{r}(\mathbf{c}_\Delta), \phi_\Delta)_T. \quad (11)$$

In this work, we use the Lax–Friedrichs flux combined with the Roe–Pike averaging [38, 41, 42] defined as

$$\hat{\mathbf{A}}(\mathbf{c}_\Delta, \mathbf{c}_\Delta^+, \mathbf{n}) = \frac{1}{2} (\mathbf{A}(\mathbf{c}_\Delta) + \mathbf{A}(\mathbf{c}_\Delta^+)) \cdot \mathbf{n} + \frac{\lambda}{2} (\mathbf{c}_\Delta - \mathbf{c}_\Delta^+), \quad (12)$$

where $\lambda = \lambda(\mathbf{c}_\Delta, \mathbf{c}_\Delta^+, \mathbf{n})$ is given by

$$\lambda(\mathbf{c}_\Delta, \mathbf{c}_\Delta^+, \mathbf{n}) = \left| \frac{(U_\Delta \sqrt{H_\Delta^+} + U_\Delta^+ \sqrt{H_\Delta}) n_x + (V_\Delta \sqrt{H_\Delta^+} + V_\Delta^+ \sqrt{H_\Delta}) n_y}{H_\Delta \sqrt{H_\Delta^+} + H_\Delta^+ \sqrt{H_\Delta}} \right| + \sqrt{\frac{g}{2} (H_\Delta + H_\Delta^+)}$$

with $H_\Delta^+ := \xi_\Delta^+ - z_b$, where we exploit the fact that the bathymetry z_b is assumed to be continuous.

Discrete initial conditions are obtained using suitable projections of ξ_0 , and \mathbf{q}_0 into the discrete ansatz space (9) (cf. [11, Sect. 5] for a comparison of reasonable choices). For the temporal discretization, we use strong stability preserving (SSP) explicit Runge–Kutta methods [43].

3. Implementation aspects

Our implementation of the EG method is based on the DG code for the SWE realized within the FESTUNG framework [35, 44]. To obtain the required EG operators, we use a simple strategy of modifying the DG code. This trick is by no means restricted to the SWE system; it can be just as easily applied to any linear or nonlinear PDE. The main idea of our approach is to exploit the fact that EG (or, for that matter also CG) approximation spaces are embedded in the DG spaces of the same order. Therefore, all EG or CG basis functions can be represented as linear combinations of DG basis functions; this representation can be written in the form of a (generally rectangular) system of linear equations, which can then be used to convert an available DG operator into the corresponding one for EG or

CG spaces. The downside of this approach is that a full DG discretization must be assembled first—thus limiting the efficiency gain due to smaller approximation spaces of the EG method. In the solution procedure, a linear system of equations with the mass matrix needs to be solved. We observed some speedup in the solution procedure for the EG method when choosing a diagonal mass matrix (e.g., by lumping) or semi-implicit time discretizations. This speedup is due to a smaller size of the linear equation system compared to DG.

In this work, we utilize finite element spaces of polynomial degree $k \leq 2$. Let us first consider the DG space $\mathbb{P}_k(\mathcal{T}_\mathcal{A})$ and denote by $N := \dim \mathbb{P}_k(\mathcal{T}_\mathcal{A})$ the number DG unknowns for a scalar quantity. We have $\dim \mathbb{P}_1(\mathcal{T}_\mathcal{A}) = 3\#T$ and $\dim \mathbb{P}_2(\mathcal{T}_\mathcal{A}) = 6\#T$. For fixed k , let $\{\varphi_i : i \in \{1, \dots, N\}\}$, be the element-wise continuous nodal basis of the DG method, where the nodal property is local to every element $T \in \mathcal{T}_\mathcal{A}$. Moreover, we denote by $\{\phi_i : i \in \{1, \dots, M\}\}$ with $M := \dim \mathbb{P}_{k,m}(\mathcal{T}_\mathcal{A})$ a basis of the EG space.

In the following, we construct bases for various EG spaces from CG and DG basis functions. Here, one has to be somewhat careful since, in general, simply combining a CG basis with element-wise DG basis functions produces a linearly-dependent set. For the admissible combinations of k and m with $-1 \leq m < k \leq 2$ considered in our work, the EG bases can be constructed as follows:

- $k \in \{1, 2\}$, $m = -1$: The space $\mathbb{P}_{k,-1}(\mathcal{T}_\mathcal{A})$ coincides with the CG ansatz space of order at most k . Therefore, we can simply choose the CG basis functions.
- $k = 1$, $m = 0$: The union of characteristic functions of all elements $T \in \mathcal{T}_\mathcal{A}$ and the continuous, piecewise linear interpolatory functions for all but one mesh vertex form a basis.
- $k = 2$, $m = 0$: We obtain a basis from the union of characteristic functions of all elements and all but one of the shape functions for the quadratic CG space.
- $k = 2$, $m = 1$: We use the standard linear DG basis and extend it by the nodal quadratic shape functions equal to 1 at one edge midpoint, but omit the ones corresponding to cell vertices. The functions are linearly independent and are contained in the space $\mathbb{P}_{2,1}(\mathcal{T}_\mathcal{A})$. As the number of basis functions equals $\dim \mathbb{P}_{2,1}(\mathcal{T}_\mathcal{A}) = 3\#T + \#E$, it must be a basis of $\mathbb{P}_{2,1}(\mathcal{T}_\mathcal{A})$. Here, $\#T$ and $\#E$ denote the number of triangles and edges of $\mathcal{T}_\mathcal{A}$, respectively. The dimension formula (10) ensures that this, in fact, constitutes a basis of $\mathbb{P}_{2,1}(\mathcal{T}_\mathcal{A})$.

3.1. Assembling nonlinear EG operators from DG operators

Next, we show how to obtain an EG from the corresponding DG operator. To simplify the presentation, we formulate our approach here for the scalar case noting that the generalization to vector fields is straightforward. DG discretizations of nonlinear PDEs such as the SWE feature nonlinear operators of the form

$$\mathbf{B}^{\text{DG}} : \mathbb{R}^N \rightarrow \mathbb{R}^N, \quad \mathbf{x} \mapsto \left(b \left(\sum_{j=1}^N x_j \varphi_j, \varphi_i \right) \right)_{i=1, \dots, N}, \quad (13)$$

where $b : \mathbb{P}_k(\mathcal{T}_\mathcal{A}) \times \mathbb{P}_k(\mathcal{T}_\mathcal{A}) \rightarrow \mathbb{R}$ is linear in the second argument. Our goal is to form the EG operator

$$\mathbf{B}^{\text{EG}} : \mathbb{R}^M \rightarrow \mathbb{R}^M, \quad \mathbf{y} \mapsto \left(b \left(\sum_{j=1}^M y_j \phi_j, \phi_i \right) \right)_{i=1, \dots, M}, \quad (14)$$

by making use of (13). Since $\mathbb{P}_{k,m}(\mathcal{T}_\mathcal{A}) \subset \mathbb{P}_k(\mathcal{T}_\mathcal{A})$, it is possible to express the EG basis functions as linear combinations of the DG basis

$$\phi_i = \sum_{k=1}^N C_{ik} \varphi_k, \quad C_{ik} \in \mathbb{R}, \quad i \in \{1, \dots, M\}. \quad (15)$$

Next, we insert (15) into (14), obtaining

$$b \left(\sum_{j=1}^M y_j \phi_j, \phi_i \right) = b \left(\sum_{j=1}^M y_j \sum_{l=1}^N C_{jl} \varphi_l, \sum_{k=1}^N C_{ik} \varphi_k \right) = \sum_{k=1}^N C_{ik} b \left(\sum_{l=1}^N \left(\sum_{j=1}^M C_{jl} y_j \right) \varphi_l, \varphi_k \right) \quad \forall i \in \{1, \dots, M\}.$$

By defining $\mathbf{C} := (C_{kl})_{kl}$ for $k \in \{1, \dots, M\}$, $l \in \{1, \dots, N\}$, we can thus write

$$\mathbf{B}^{\text{EG}}(\mathbf{y}) = \left(b \left(\sum_{j=1}^M y_j \phi_j, \phi_i \right) \right)_{i=1, \dots, M} = \mathbf{C} \left(b \left(\sum_{l=1}^N (\mathbf{C}^\top \mathbf{y})_l \varphi_l, \varphi_k \right) \right)_{k=1, \dots, N} = \mathbf{C} \mathbf{B}^{\text{DG}}(\mathbf{C}^\top \mathbf{y}) \quad (16)$$

Employing (16), we assemble the terms corresponding to (14) by modifying an existing DG discretization of the operator (13) and computing the matrix \mathbf{C} . Due to the above choice of EG and DG basis functions, the matrix \mathbf{C} can be determined from simple geometric considerations during preprocessing.

3.2. Assembly of the EG mass matrix

We consider the operator b_M defined by $b_M(u_A, v_A) := (u_A, v_A)_\Omega$. Since b_M is a bilinear form, we can write its induced operator (mass matrix) as

$$\mathbf{B}_M^{\text{DG}} : \mathbb{R}^N \rightarrow \mathbb{R}^N, \quad \mathbf{x} \mapsto \mathbf{B}_M^{\text{DG}} \mathbf{x}$$

with $(\mathbf{B}_M^{\text{DG}})_{ij} = \int_\Omega \varphi_j \varphi_i dx$ for $i, j \in \{1, \dots, N\}$. The corresponding EG mass matrix

$$\mathbf{B}_M^{\text{EG}} \in \mathbb{R}^{M \times M}, \quad (\mathbf{B}_M^{\text{EG}})_{ij} = \int_\Omega \phi_j \phi_i dx$$

and can be obtained from (16), and, in operator form, is given by

$$\mathbf{B}_M^{\text{EG}} : \mathbb{R}^M \rightarrow \mathbb{R}^M, \quad \mathbf{y} \mapsto \mathbf{C} \mathbf{B}_M^{\text{DG}} \mathbf{C}^\top \mathbf{y}.$$

That is, for operators that are induced by bilinear forms, the assembly can be preprocessed.

4. Numerical results

In this section, we investigate the performance of the EG method using artificial and realistic test problems for the SWE. The main goals of these numerical studies can be summarized as follows:

- Verify the expected rate of convergence against a manufactured analytical solution;
- evaluate the solution quality for realistic benchmarks;
- compare the EG and DG methods in terms of accuracy, stability, and robustness.

We denote the results obtained for different combinations of k and m by the corresponding finite element spaces. To simplify notation, we omit their dependency on \mathcal{T}_A . Hence, we write $\mathbb{P}_{k,m}$ instead of $\mathbb{P}_{k,m}(\mathcal{T}_A)$ and use the convention that $\mathbb{P}_{k,k}$ is the DG space of polynomial degree k .

As temporal discretization, we use an SSP Runge–Kutta time stepping scheme described in [45] with $s = k + 1$ stages. The time step size depends on the specific test problem.

4.1. Analytical convergence test

For our first numerical experiment, we approximate a smooth solution of the SWE on the square domain $\Omega := (0, 1000) \times (0, 1000)$. The coarsest mesh (corresponding to level one) consists of 16 triangular elements and is shown in Fig. 2 (left). To investigate the convergence behavior, we consider a total of five meshes obtained from the coarsest mesh by uniform refinement via edge bisection.

The bathymetry is prescribed as

$$z_b(x, y) := -4 + \frac{1}{1000}x + \frac{2}{1000}y.$$

Parameters τ_{bf} and f_c are set to zero. Substituting the following analytical solution

$$\begin{aligned}\xi(x, y, t) &:= 2 + C_1 - 2 C_2 \sin\left(\frac{\pi(x + y + C_3 t)}{600}\right), \\ U(x, y, t) &:= 2 C_1 + C_2 C_3 \sin\left(\frac{\pi(x + y + C_3 t)}{600}\right), \\ V(x, y, t) &:= C_1 + C_2 C_3 \sin\left(\frac{\pi(x + y + C_3 t)}{600}\right).\end{aligned}$$

into (2),(3) we use the method of manufactured solutions with the corresponding right hand side and Dirichlet boundary conditions for each unknown imposed on all boundaries. We solve the SWE for the time interval $(0, 1000)$ using the time step size $\Delta t = 1/4$ in all considered scenarios. This value of Δt is sufficiently small to make temporal discretization errors negligible compared to spatial approximation errors. Solution parameters are chosen as $C_1 = 0.3$, $C_2 = 0.2$, $C_3 = 0.2$.

The projected initial surface elevation for $\mathbb{P}_{1,0}$ on the coarsest mesh is shown in Fig. 2 (middle). Fig. 2 (right) displays the numerical solution for the surface elevation using the same approximation ($k = 1$, $m = 0$) on the finest mesh at the final time.

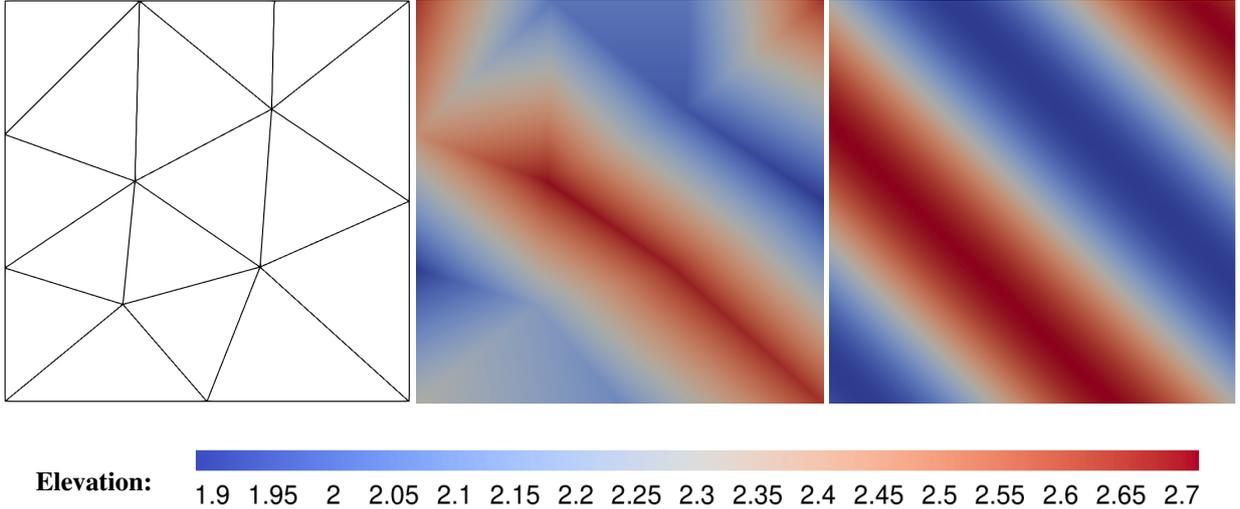


Figure 2: Analytical convergence test: Domain with coarsest mesh (left), initial surface elevation of linear EG method ($k = 1$, $m = 0$) on coarsest mesh (middle), linear EG approximation ($k = 1$, $m = 0$) of the surface elevation on the finest mesh (level 5) at the final time $t = 1000$ (right).

For all considered CG, EG, and DG methods ($-1 \leq m \leq k \leq 2$), we list the $L^2(\Omega)$ discretization errors at the final time along with the corresponding convergence rates in Tab. 1 and plot them in Fig. 3. The results indicate that we obtain at least second order convergence for all methods. Third order convergence can be observed for $k = 2$ and $m \in \{1, 2\}$, but not for the CG approximation $k = 2$, $m = -1$. The results for $k = 2$ and $m = 0$ are slightly better than without the piecewise constant enrichment, but do not exhibit third order accuracy. In conclusion, the EG scheme converges almost exactly as well as the DG method if $m = k - 1$, $k \in \{1, 2\}$, although the absolute errors tend to be somewhat larger than for their DG counterparts. This is to be expected because EG has fewer unknowns, and even the projected exact solution in general becomes less accurate if the number of unknowns is reduced.

In Tab. 2, we list the total numbers of degrees of freedom for all configurations. Note that, the DOF for the EG method with element-wise constant enrichment ($k \in \{1, 2\}$, $m = 0$) has approximately half as many DOF as the DG method of order k . EG with $k = 2$, $m = 1$ has approximately three quarters of the DOF for the quadratic DG method, thus indicating a potential gain in computational cost for EG compared to DG. In conclusion, the EG method performs similarly to DG for smooth solutions while having significantly fewer DOF.

Method	Level	Δ	Err(ξ)	EOC(ξ)	Err(U)	EOC(U)	Err(V)	EOC(V)
$\mathbb{P}_{1,-1}$	1	500	1.47E+02	–	4.98E+02	–	4.49E+02	–
	2	250	2.12E+01	2.80	1.05E+02	2.24	8.06E+01	2.48
	3	125	4.15E+00	2.36	2.82E+01	1.90	2.22E+01	1.86
	4	62.5	9.55E–01	2.12	8.42E+00	1.75	6.43E+00	1.79
	5	31.25	2.35E–01	2.02	2.23E+00	1.92	1.65E+00	1.96
$\mathbb{P}_{1,0}$	1	500	4.94E+01	–	1.35E+02	–	1.20E+02	–
	2	250	1.22E+01	2.01	3.07E+01	2.13	3.45E+01	1.80
	3	125	3.10E+00	1.98	7.38E+00	2.06	8.76E+00	1.98
	4	62.5	7.80E–01	1.99	1.83E+00	2.01	2.21E+00	1.99
	5	31.25	1.95E–01	2.00	4.59E–01	2.00	5.56E–01	1.99
$\mathbb{P}_{1,1}$	1	500	3.02E+01	–	1.11E+02	–	9.21E+01	–
	2	250	6.91E+00	2.13	2.34E+01	2.25	2.54E+01	1.86
	3	125	1.65E+00	2.07	5.62E+00	2.06	6.25E+00	2.02
	4	62.5	4.02E–01	2.04	1.40E+00	2.01	1.57E+00	2.00
	5	31.25	9.98E–02	2.01	3.48E–01	2.01	3.92E–01	2.00
$\mathbb{P}_{2,-1}$	1	500	1.68E+01	–	8.51E+01	–	8.18E+01	–
	2	250	4.29E+00	1.97	3.59E+01	1.25	3.20E+01	1.35
	3	125	1.07E+00	2.00	1.11E+01	1.69	9.87E+00	1.70
	4	62.5	2.66E–01	2.01	3.01E+00	1.89	2.64E+00	1.90
	5	31.25	6.67E–02	2.00	7.65E–01	1.98	6.68E–01	1.98
$\mathbb{P}_{2,0}$	1	500	1.19E+01	–	5.96E+01	–	6.88E+01	–
	2	250	1.79E+00	2.74	1.20E+01	2.31	1.11E+01	2.63
	3	125	2.59E–01	2.79	2.81E+00	2.09	2.43E+00	2.19
	4	62.5	4.49E–02	2.53	7.21E–01	1.96	6.41E–01	1.92
	5	31.25	9.67E–03	2.21	2.03E–01	1.83	1.88E–01	1.77
$\mathbb{P}_{2,1}$	1	500	9.60E+00	–	3.30E+01	–	3.59E+01	–
	2	250	1.41E+00	2.77	4.88E+00	2.76	4.76E+00	2.92
	3	125	1.80E–01	2.97	5.72E–01	3.09	6.02E–01	2.98
	4	62.5	2.28E–02	2.98	6.80E–02	3.07	7.96E–02	2.92
	5	31.25	2.88E–03	2.99	8.94E–03	2.93	1.12E–02	2.84
$\mathbb{P}_{2,2}$	1	500	6.85E+00	–	3.20E+01	–	3.37E+01	–
	2	250	7.56E–01	3.18	4.43E+00	2.85	4.02E+00	3.07
	3	125	8.86E–02	3.09	5.10E–01	3.12	4.77E–01	3.07
	4	62.5	1.09E–02	3.03	5.76E–02	3.15	6.11E–02	2.96
	5	31.25	1.35E–03	3.01	7.26E–03	2.99	8.31E–03	2.88

Table 1: Analytical convergence test: $L^2(\Omega)$ errors Err(\cdot) at the final time and experimental orders of convergence EOC(\cdot) for the surface elevation and depth integrated velocity components.

Level	Δ	# T	# V	# E	Degrees of freedom						
					$\mathbb{P}_{1,-1}$	$\mathbb{P}_{1,0}$	$\mathbb{P}_{1,1}$	$\mathbb{P}_{2,-1}$	$\mathbb{P}_{2,0}$	$\mathbb{P}_{2,1}$	$\mathbb{P}_{2,2}$
1	500	16	14	29	14	29	48	43	58	77	96
2	250	64	43	106	43	106	192	149	212	298	384
3	125	256	149	404	149	404	768	553	808	1172	1536
4	62.5	1024	553	1576	553	1576	3072	2129	3152	4648	6144
5	31.25	4096	2129	6224	2129	6224	12228	8353	12448	18512	24576

Table 2: Series of refined triangular meshes: Number of triangles (# T), vertices (# V), edges (# E), and degrees of freedom for CG, EG, and DG for all considered orders.

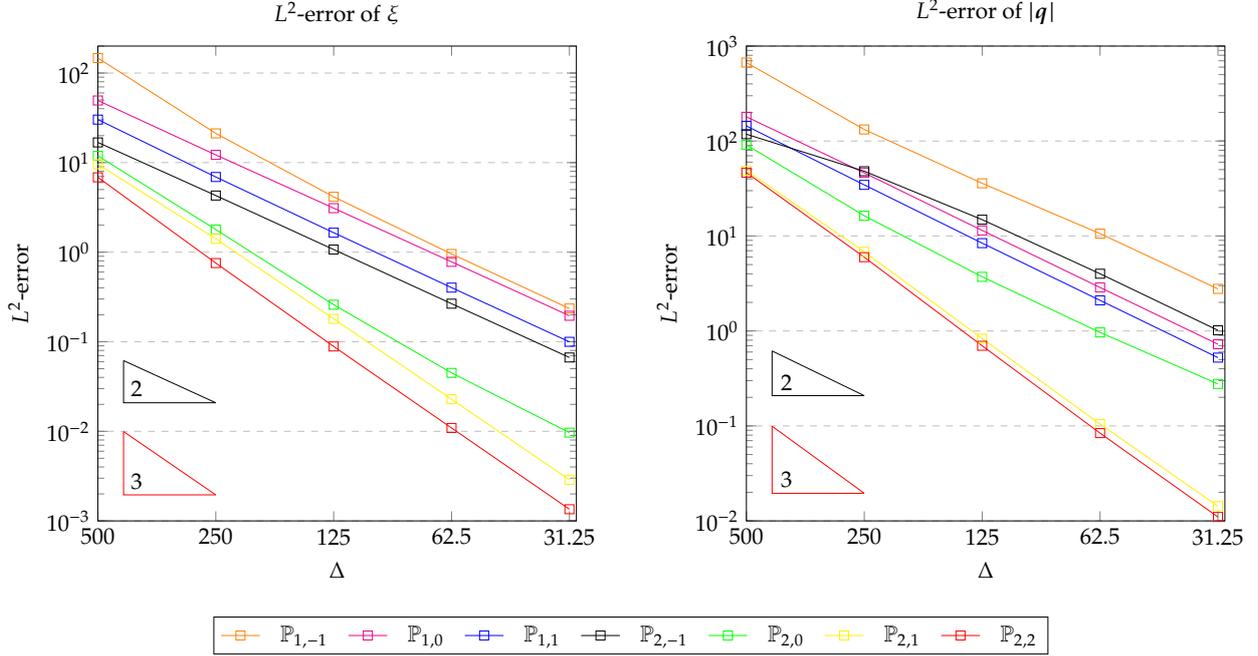


Figure 3: Analytical convergence test: Logarithmic plots of the $L^2(\Omega)$ errors at the final time for the surface elevation ξ (left) and the depth integrated velocity magnitude $|q|$ (right) for CG, EG, and DG discretizations.

160 4.2. Supercritical flow in a constricted channel

In order to demonstrate the stability and robustness of the EG method, we solve the supercritical flow problem proposed by Zienkiewicz and Ortiz in [46]. The computational domain is a channel whose lateral boundary walls are constricted on both sides with an angle of five degrees (cf. Fig. 4). This benchmark uses constant bathymetry $z_b \equiv -1$ while parameters τ_{bf} , and f_c are once more set to zero. The following initial and boundary conditions are prescribed for this problem: Initially, we set the surface elevation and momentum to $\xi_0 \equiv 0$, $\mathbf{q}_0 = (1, 0)^T$, respectively. Land boundary conditions are imposed at the lateral (wall) boundaries, while at the inlet (left), free surface elevation as well as velocity are for all times specified to be identical to their corresponding initial values. Finally, at the outlet (right) radiation boundary conditions are used. Denoting by u and H the axial velocity and water depth at the inlet, respectively, the flow regime is set to be supercritical by choosing the inlet Froude number Fr

$$\text{Fr} := \frac{u}{\sqrt{gH}} = 2.5,$$

which is equivalent to setting the gravitational constant $g = 0.16 \frac{m}{s^2}$ in this artificial example.

The solution to this problem converges to a steady-state, for which an analytical solution is available (see [47]). Fig. 4 illustrates the computational domain along with the unstructured mesh used in all computations.

165 We run all simulations to a steady-state using pseudo time stepping with a time step of $\Delta t = 1/10$ and present the results for all schemes in Fig. 5. The steady-state surface elevation shown in Fig. 5 (top left) is discontinuous and displays interactions of waves reflected from the channel constrictions. Fig. 5 (top right to bottom right), display the steady-state solutions for all considered EG and DG methods ($0 \leq m \leq k \leq 2$). Since no limiter was used, all approximations exhibit spurious oscillations close to the discontinuities. We expect the results to improve and the numerical solutions to become bound-preserving if a limiter, such as the one developed in [27], is utilized (at least)
170 for the free surface elevation.

The type of enrichment appears to play a major role for the stability of EG schemes. Thus the results for enrichments with $m = k - 1$, $k \in \{1, 2\}$ are almost indistinguishable from the corresponding DG results. In particular, the characteristic wave features such as positions and magnitudes of discontinuities are in good agreement with the analytical solution. On the other hand, the EG solution for $k = 2$, $m = 0$ exhibits severe oscillations not only in the vicinity

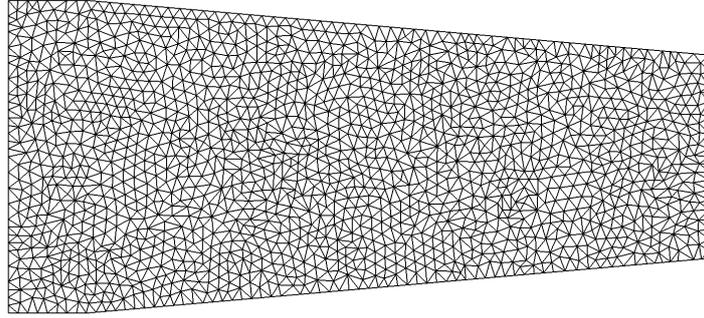


Figure 4: Supercritical flow in a constricted channel: Unstructured mesh with 3155 elements.

175 of the discontinuities but also in the remainder of the computational domain. This phenomenon raises the concern
a piecewise constant enrichment of the quadratic CG space is not sufficient to obtain an intrinsically stable scheme.
However, an enrichment by piecewise linear discontinuous functions seems to remedy this issue, which confirms the
previous findings in 4.1 and [11]. Our study of this benchmark suggests that optimal EG schemes ($m = k - 1$) possess
similar stability properties to their DG counterparts while offering potential advantages in computational efficiency.

180 4.3. Tidal flow at Bahamas Islands

The realistic benchmark considered in this work involves a tidal flow scenario around the Bahamas Islands. The
domain geometry, bathymetry as well as boundary types are depicted in Fig. 6 (left). Furthermore, four recording
stations are placed at the following locations (38 667, 49 333), (56 098, 9 613), (41 263, 29 776), and (59 594, 41 149)
(coordinates in meters). These stations monitor the temporal evolution of the numerical solution at their respective
185 locations and can also be seen in Fig. 6 (left). Fig. 6 (right) shows the unstructured mesh used in all simulations.

For the bottom friction coefficient we use the standard quadratic friction law $\tau_{\text{bf}} = C_f |\mathbf{q}|/H^2$ (see e.g. [48]) with
coefficient $C_f = 0.009$. The (constant) Coriolis parameter is set to $3.19 \times 10^{-5} \text{ s}^{-1}$. The following tidal forcing is
prescribed at the open sea boundary:

$$\begin{aligned} \hat{\xi}(t) = & 0.075 \cos\left(\frac{t}{25.82} + 3.40\right) + 0.095 \cos\left(\frac{t}{23.94} + 3.60\right) + 0.100 \cos\left(\frac{t}{12.66} + 5.93\right) \\ & + 0.395 \cos\left(\frac{t}{12.42} + 0.00\right) + 0.060 \cos\left(\frac{t}{12.00} + 0.75\right) \end{aligned} \quad (17)$$

with time t in hours. In real-life ocean simulations, the initial conditions are often unknown or very difficult to
obtain, therefore a cold start initialization is performed: The flow domain is assumed to be at rest initially ($\xi_0 \equiv 0$,
 $\mathbf{q}_0 \equiv (0, 0)^T$). Then, starting immediately at initial time $t = 0$, boundary condition (17) is imposed as A tidal forcing
driving the flow.

190 The simulations are run for a total of twelve days using constant time step $\Delta t = 15$ seconds for EG and DG
discretizations corresponding to $0 \leq m \leq k \leq 2$. In Fig. 7–9, we compare the numerical solutions for all considered
EG and DG methods at the four recording stations.

Fig. 7 demonstrates excellent agreement for the surface elevation for all EG and DG methods. The curves lie
nearly on top of each other and no differences can be visually detected at this resolution. The differences in the
surface elevation between the EG and DG approximations at the recording stations are on the order of 10^{-4} meters.
195

In Fig. 8 and Fig. 9, we plot both velocity components at the recording stations. For the recording station two (Fig.
8 and Fig. 9 top right), one can observe slight differences between the approximations of orders one and two. Such
behavior is fully consistent with the station comparisons performed in [38]. The effect of approximation order has
greater magnitude than the differences between the EG and DG discretizations. Small differences between EG and
200 DG discretizations of the same order can be observed in the plots at some locations. The deviations at the recording
stations are on the order of 10^{-3} to 10^{-4} m s^{-1} .

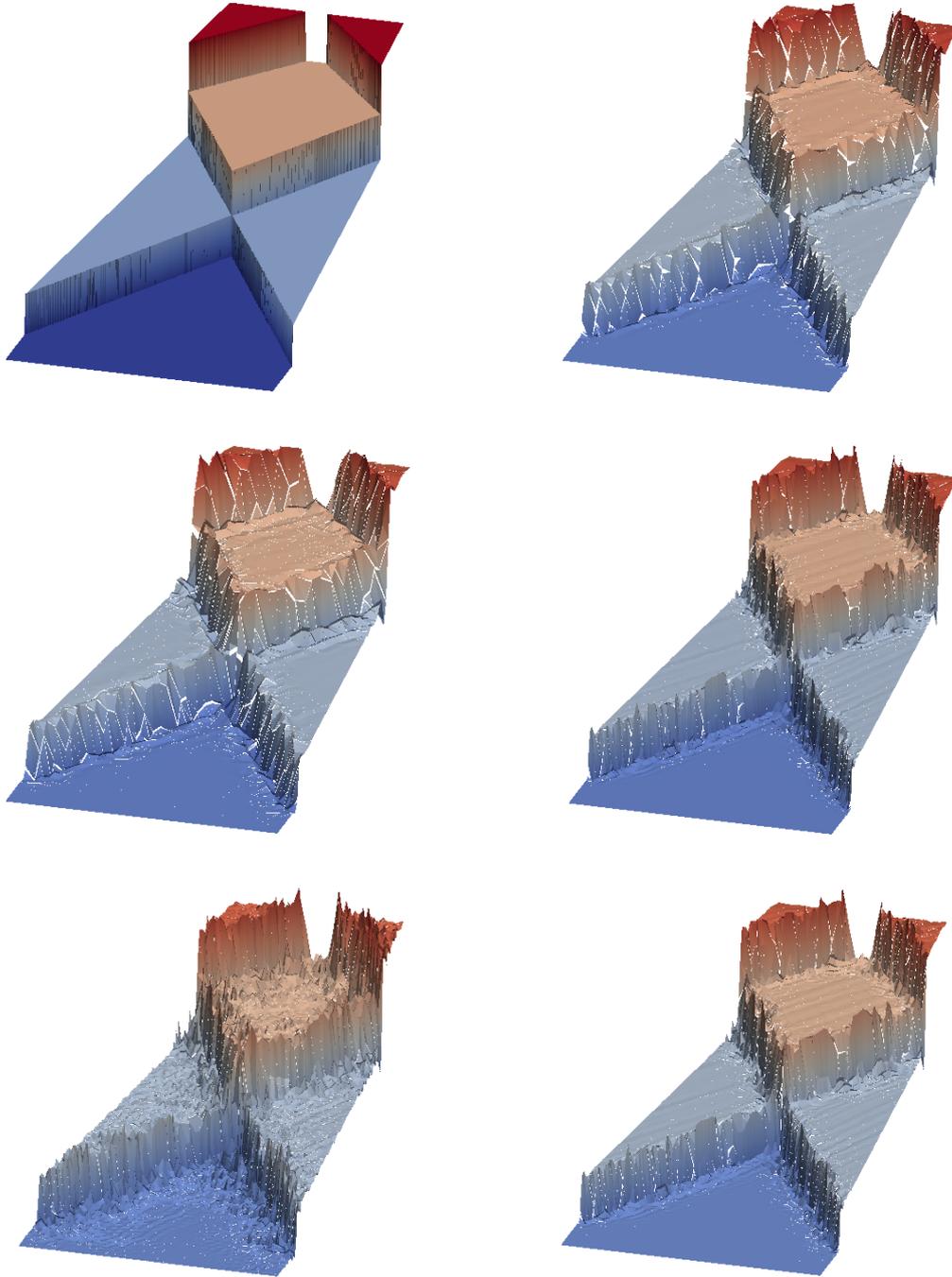


Figure 5: Supercritical flow in a constricted channel, free surface elevation, analytical solution (top left) and approximations at steady-state for: $\mathbb{P}_{1,1}$ (top right), $\mathbb{P}_{1,0}$ (middle left), $\mathbb{P}_{2,2}$ (middle right), $\mathbb{P}_{2,0}$ (bottom left), $\mathbb{P}_{2,1}$ (bottom right).

In addition to testing the accuracy and robustness of the EG method, we also used the Bahamas example to verify the well-balancedness of the method. For this purpose, we ran our simulation for the lake-at-rest configuration with open sea boundary condition set to zero instead of using (17). Just as expected, no spurious circulation emerged for any of the EG or DG configurations.

205

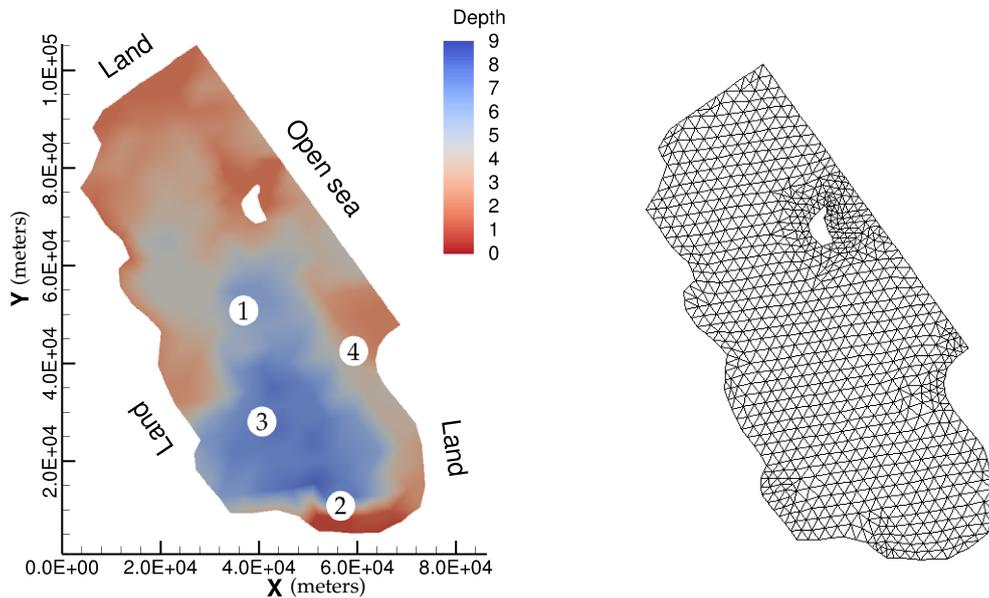


Figure 6: Tidal flow around Bahamas Islands: Bathymetry, boundary type and positions of recording stations (left), unstructured triangular mesh with 1696 elements (right).

5. Conclusions

In this work, we present the first enriched Galerkin discretization for the system of 2D shallow-water equations, evaluated its performance in analytical and realistic test problems, and compared the numerical results to those obtained using our discontinuous Galerkin solver. The results of our studies demonstrate that EG schemes with enrichments using discontinuous spaces of order one less than the order of the continuous space display similar accuracy and robustness as the corresponding DG discretizations. Similarly to DG, EG discretizations guarantee local conservation of all conserved unknowns while the total number of degrees of freedom is substantially lower than that for the DG space of the same order. This makes the enriched Galerkin method a very attractive candidate for solving the shallow-water equations and other nonlinear hyperbolic PDE systems.

Several interesting avenues of future research concerning the development of the EG solver for the SWE present themselves at this time. Thus one could employ the quadrature-free methodology to improve the computational efficiency of the method—similarly to our recent work for the DG SWE solver in [49]. Also EG discretizations appears to hold promise (perhaps even more so than the DG method) for bathymetry reconstruction using modified shallow-water equations [50]. Finally, stabilization techniques, such as limiters, e.g., the one designed specifically for EG in [27], will be required for certain applications of the SWE such as dam break problems.

Acknowledgments

This work is supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy EXC 2181/1 – 390900948 (the Heidelberg STRUCTURES Excellence Cluster).

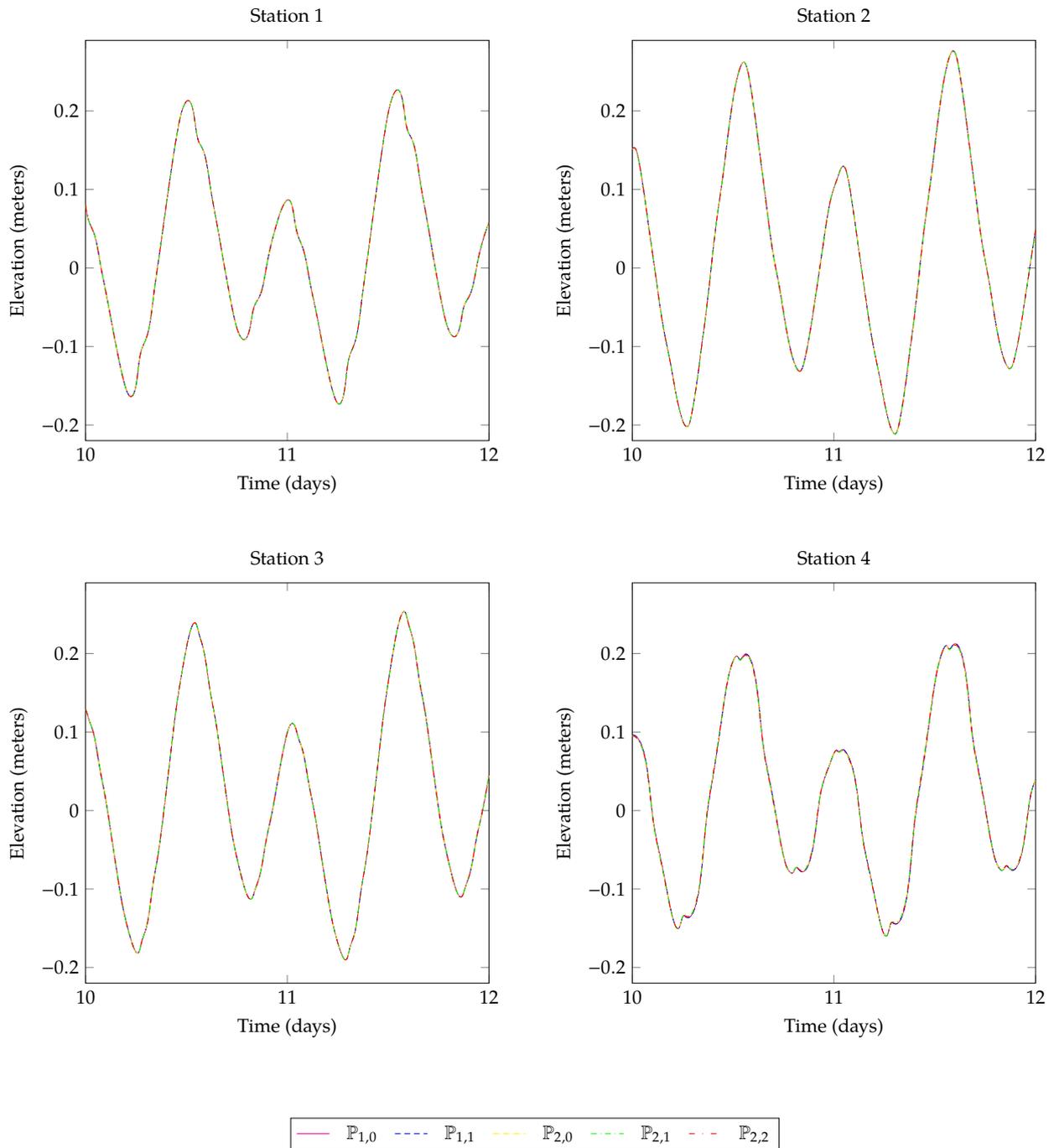


Figure 7: Tidal flow at Bahamas Islands: Free surface elevation for days eleven and twelve at recording stations one to four (left to right, top to bottom).

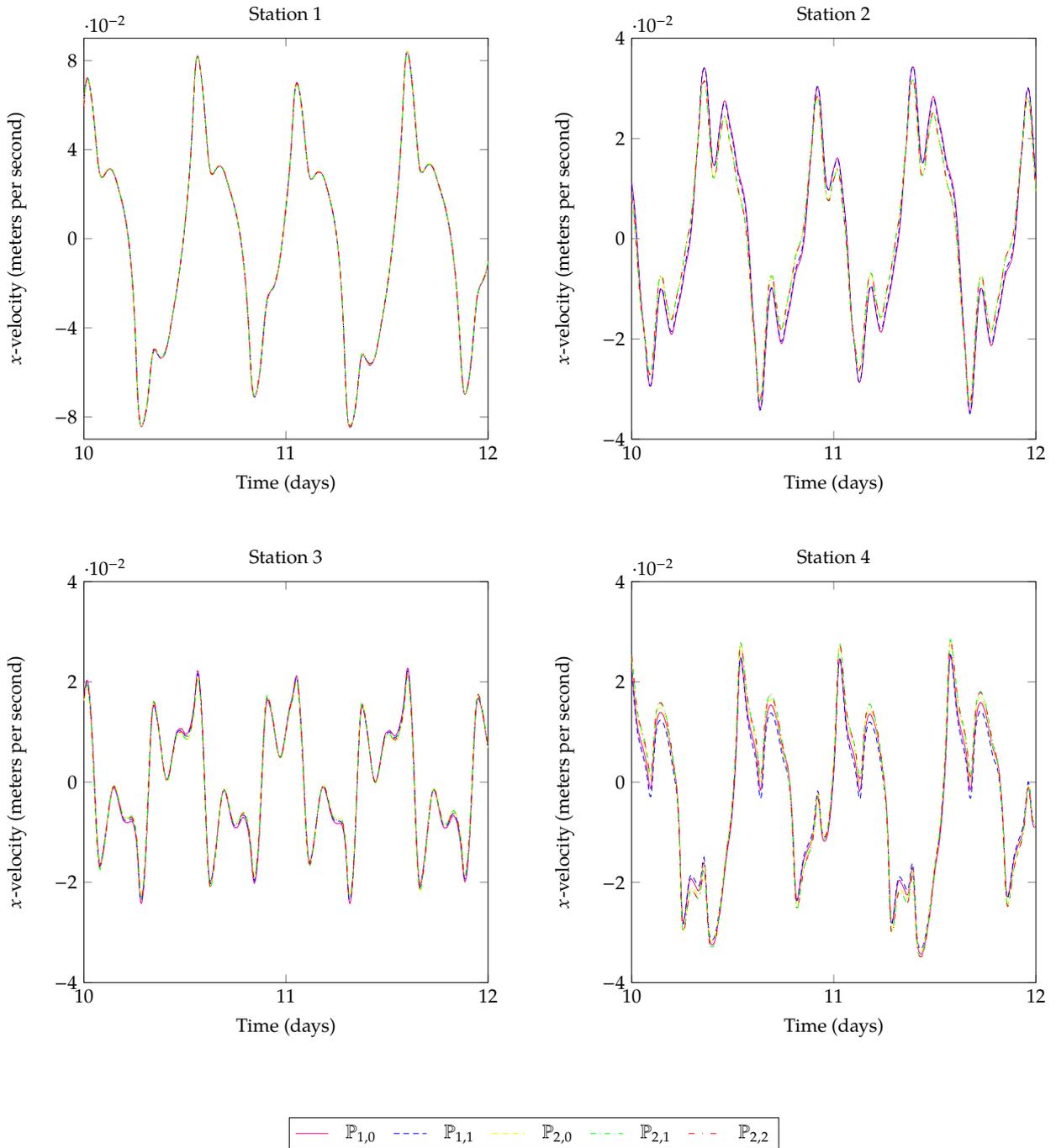


Figure 8: Tidal flow at Bahamas Islands: x -velocity component for days eleven and twelve at recording stations one to four (left to right, top to bottom).

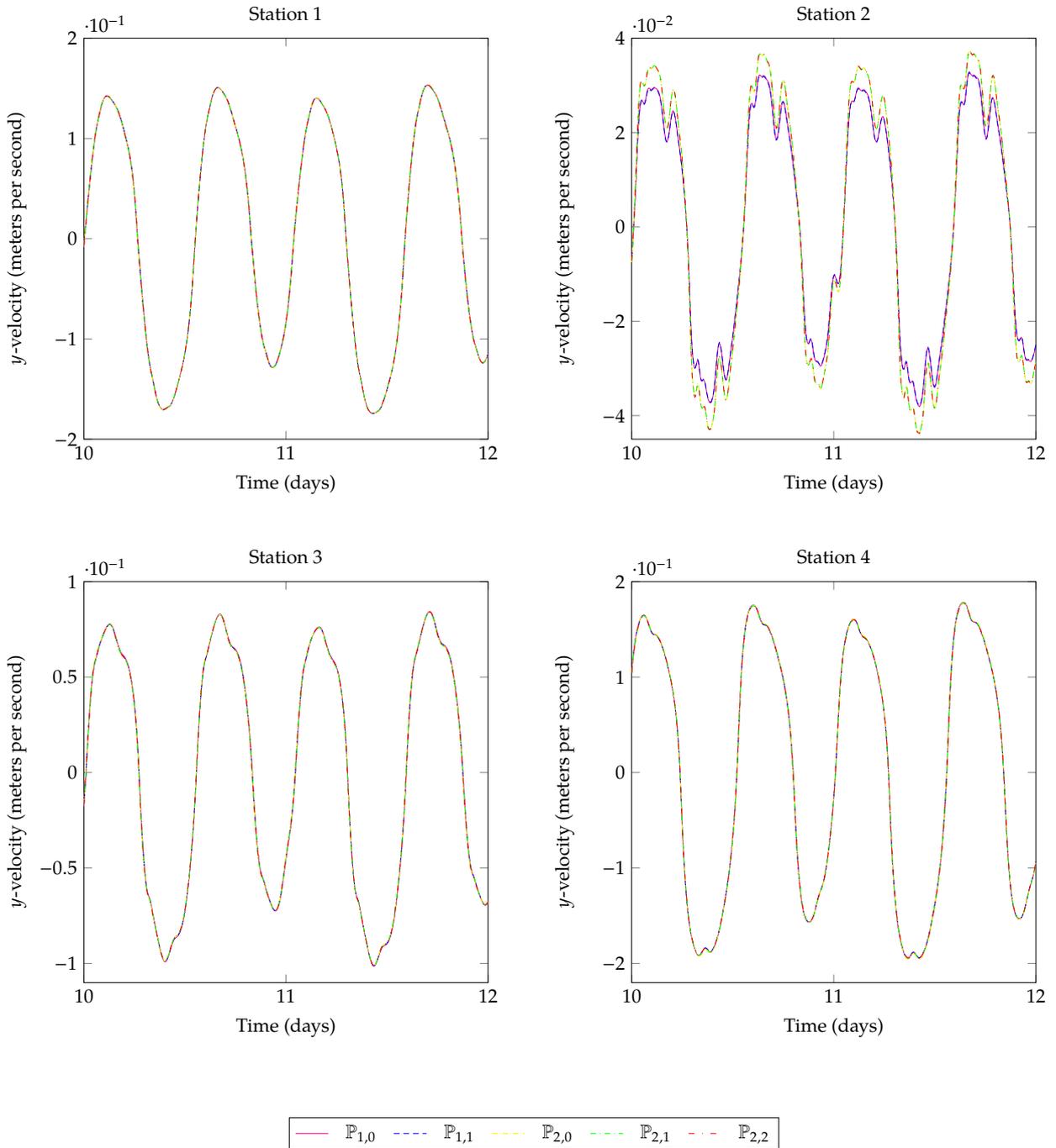


Figure 9: Tidal flow at Bahamas Islands: y -velocity component for days eleven and twelve at recording stations one to four (left to right, top to bottom).

References

- 225 [1] R. Hinkelmann, Q. Liang, V. Aizinger, C. Dawson, **Robust shallow water models**, *Environmental Earth Sciences* 74 (11) (2015) 7273–7274, (editorial). doi:10.1007/s12665-015-4764-1.
URL <https://eref.uni-bayreuth.de/52288/>
- [2] J.-C. Galland, N. Goutal, J.-M. Hervouet, **TELEMAC: A new numerical model for solving shallow water equations**, *Advances in Water Resources* 14 (3) (1991) 138–148. doi:10.1016/0309-1708(91)90006-A.
- 230 [3] R. Luettich, J. Westerink, N. Scheffner, **ADCIRC: An advanced three-dimensional circulation model for shelves, coasts and estuaries, Report 1: Theory and methodology of ADCIRC-2DDI and ADCIRC-3DL**, Tech. Rep. Dredging Research Program Technical Report DRP-92-6, US Army Engineers Waterways Experiment Station, Vicksburg, MS (1992).
- [4] E. Hanert, V. Legat, E. Deleersnijder, **A comparison of three finite elements to solve the linear shallow water equations**, *Ocean Modelling* 5 (1) (2003) 17–35. doi:10.1016/S1463-5003(02)00012-4.
- 235 [5] R. Comblen, J. Lambrechts, J.-F. Remacle, V. Legat, **Practical evaluation of five partly discontinuous finite element pairs for the non-conservative shallow water equations**, *International Journal for Numerical Methods in Fluids* 63 (6) (2010) 701–724. doi:10.1002/flid.2094.
- [6] C. Dawson, J. Westerink, J. Feyen, D. Pothina, **Continuous, discontinuous and coupled discontinuous–continuous Galerkin finite element methods for the shallow water equations**, *International Journal for Numerical Methods in Fluids* 52 (1) (2006) 63–88. doi:10.1002/flid.1156.
- 240 [7] E. Kubatko, S. Bunya, C. Dawson, J. Westerink, C. Mirabito, **A performance comparison of continuous and discontinuous finite element shallow water models**, *Journal of Scientific Computing* 40 (1) (2009) 315–339. doi:10.1007/s10915-009-9268-2.
- [8] R. Becker, E. Burman, P. Hansbo, M. G. Larson, **A reduced P1-discontinuous Galerkin method**, *Chalmers Finite Element Center Preprint* 2003-13.
- 245 [9] S. Sun, J. Liu, **A locally conservative finite element method based on piecewise constant enrichment of the continuous Galerkin method**, *SIAM Journal on Scientific Computing* 31 (4) (2009) 2528–2548. doi:10.1137/080722953.
- [10] S. Lee, Y.-J. Lee, M. F. Wheeler, **A locally conservative enriched Galerkin approximation and efficient solver for elliptic and parabolic problems**, *SIAM Journal on Scientific Computing* 38 (3) (2016) A1404–A1429. doi:10.1137/15M1041109.
- [11] A. Rupp, S. Lee, **Continuous Galerkin and enriched Galerkin methods with arbitrary order discontinuous trial functions for the elliptic and parabolic problems with jump conditions**, submitted to *Journal of Scientific Computing*.
- 250 [12] W. Choi, S. Lee, **Optimal error estimate of elliptic problems with Dirac sources for discontinuous and enriched Galerkin methods**, *Applied Numerical Mathematics* 150 (2019) 76–104. doi:10.1016/j.apnum.2019.09.010.
- [13] S. Lee, A. Mikelic, M. F. Wheeler, T. Wick, **Phase-field modeling of two phase fluid filled fractures in a poroelastic medium**, *Multiscale Modeling & Simulation* 16 (4) (2018) 1542–1580. doi:10.1137/17M1145239.
- 255 [14] S. Lee, M. F. Wheeler, **Adaptive enriched Galerkin methods for miscible displacement problems with entropy residual stabilization**, *Journal of Computational Physics* 331 (2017) 19–37. doi:10.1016/j.jcp.2016.10.072.
- [15] S. Lee, M. F. Wheeler, **Enriched Galerkin methods for two-phase flow in porous media with capillary pressure**, *Journal of Computational Physics* 367 (2018) 65–86. doi:10.1016/j.jcp.2018.03.031.
- [16] J. Choo, S. Lee, **Enriched Galerkin finite elements for coupled poromechanics with local mass conservation**, *Computer Methods in Applied Mechanics and Engineering* 341 (2018) 311–332. doi:10.1016/j.cma.2018.06.022.
- 260 [17] T. Kadeethum, H. M. Nick, S. Lee, C. N. Richardson, S. Salimzadeh, F. Ballarin, **A novel enriched Galerkin method for modelling coupled flow and mechanical deformation in heterogeneous porous media**, in: *53rd US Rock Mechanics/Geomechanics Symposium*, American Rock Mechanics Association, New York, NY, USA, 2019, ARMA-2019-0228.
- [18] T. Kadeethum, H. Nick, S. Lee, **Comparison of two- and three-field formulation discretizations for flow and solid deformation in heterogeneous porous media**, in: *20th Annual Conference of the International Association for Mathematical Geosciences*, 2019.
- 265 [19] A. N. Brooks, T. J. Hughes, **Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations**, *Computer methods in applied mechanics and engineering* 32 (1982) 199–259. doi:https://doi.org/10.1016/0045-7825(82)90071-8.
- [20] C.-W. Shu, **High order weighted essentially nonoscillatory schemes for convection dominated problems**, *SIAM review* 51 (2009) 82–126. doi:https://doi.org/10.1137/070679065.
- 270 [21] B. Cockburn, C.-W. Shu, **The Runge–Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems**, *J. Comput. Phys.* 141 (2) (1998) 199–224. doi:10.1006/jcph.1998.5892.
- [22] X. Zhang, C.-W. Shu, **Maximum-principle-satisfying and positivity-preserving high-order schemes for conservation laws: survey and new developments**, *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 467 (2011) 2752–2776. doi:https://doi.org/10.1098/rspa.2011.0153.
- 275 [23] M. Dumbser, O. Zanotti, R. Loubère, S. Diot, **A posteriori subcell limiting of the discontinuous Galerkin finite element method for hyperbolic conservation laws**, *Journal of Computational Physics* 278 (2014) 47–75. doi:https://doi.org/10.1016/j.jcp.2014.08.009.
- [24] T. Barth, D. Jespersen, **The design and application of upwind schemes on unstructured meshes**, in: *Proc. AIAA 27th Aerospace Sciences Meeting*, Reno, 1989.
- 280 [25] D. Kuzmin, **A vertex-based hierarchical slope limiter for adaptive discontinuous Galerkin methods**, *Journal of Computational and Applied Mathematics* 233 (12) (2010) 3077–3085, *Finite Element Methods in Engineering and Science (FEMTEC 2009)*. doi:10.1016/j.cam.2009.05.028.
- [26] V. Aizinger, **A geometry independent slope limiter for the discontinuous Galerkin method**, in: E. Krause, Y. Shokin, M. Resch, D. Kröner, N. Shokina (Eds.), *Computational Science and High Performance Computing IV*, Vol. 115 of *Notes on Numerical Fluid Mechanics and Multidisciplinary Design*, Springer Berlin Heidelberg, 2011, pp. 207–217. doi:10.1007/978-3-642-17770-5_16.
- 285 [27] D. Kuzmin, H. Hajduk, A. Rupp, **Locally bound-preserving enriched Galerkin methods for the linear advection equation**, *Computers & Fluids (to appear)* (2020) 104525doi:https://doi.org/10.1016/j.compfluid.2020.104525.

- [28] D. Kuzmin, Algebraic flux correction I. Scalar conservation laws, in: R. L. D. Kuzmin, S. Turek (Eds.), *Flux-Corrected Transport: Principles, Algorithms, and Applications*, Vol. 2, Springer, 2012, pp. 145–192.
- 290 [29] R. Anderson, V. Dobrev, T. Kolev, D. Kuzmin, M. Quezada de Luna, R. Rieben, V. Tomov, High-order local maximum principle preserving (MPP) discontinuous Galerkin finite element method for the transport equation, *J. Comput. Phys.* 334 (2017) 102–124. doi:10.1016/j.jcp.2016.12.031.
- [30] H. Hajduk, D. Kuzmin, T. Kolev, R. Abgrall, Matrix-free subcell residual distribution for Bernstein finite element discretizations of linear advection equations, *Comput. Method. Appl. M.* 359 (2020) 112658. doi:10.1016/j.cma.2019.112658.
- 295 [31] F. Frank, B. Reuter, V. Aizinger, P. Knabner, FESTUNG: A MATLAB/GNU Octave toolbox for the discontinuous Galerkin method, Part I: Diffusion operator, *Computers and Mathematics with Applications* 70 (1) (2015) 11–46. doi:10.1016/j.camwa.2015.04.013.
- [32] B. Reuter, V. Aizinger, M. Wieland, F. Frank, P. Knabner, FESTUNG: A MATLAB/GNU Octave toolbox for the discontinuous Galerkin method, Part II: Advection operator and slope limiting, *Computers and Mathematics with Applications* 72 (7) (2016) 1896–1925. doi:10.1016/j.camwa.2016.08.006.
- 300 [33] A. Jaust, B. Reuter, V. Aizinger, J. Schütz, P. Knabner, FESTUNG: A MATLAB/GNU Octave toolbox for the discontinuous Galerkin method, part III: Hybridized discontinuous Galerkin (HDG) formulation, *Computers and Mathematics with Applications* 75 (12) (2018) 4505–4533. doi:10.1016/j.camwa.2018.03.045.
- [34] B. Reuter, A. Rupp, V. Aizinger, F. Frank, P. Knabner, FESTUNG: A MATLAB / GNU Octave toolbox for the discontinuous Galerkin method. Part IV: Generic problem framework and model-coupling interface, *Communications in Computational Physics* (to appear).
- 305 [35] B. Reuter, H. Hajduk, A. Rupp, F. Frank, V. Aizinger, P. Knabner, FESTUNG 1.0: Overview, usage, and example applications of the MATLAB / GNU Octave toolbox for discontinuous Galerkin methods, submitted to *Computers and Mathematics with Applications*.
- [36] H. Hajduk, B. R. Hodges, V. Aizinger, B. Reuter, Locally filtered transport for computational efficiency in multi-component advection-reaction models, *Environmental Modelling & Software* 102 (2018) 185–198. doi:10.1016/j.envsoft.2018.01.003.
- [37] C. Dawson, V. Aizinger, The local discontinuous Galerkin method for advection-diffusion equations arising in groundwater and surface water applications, in: J. Chadam, A. Cunningham, R. E. Ewing, P. Ortoleva, M. F. Wheeler (Eds.), *Resource Recovery, Confinement, and Remediation of Environmental Hazards*, Springer, 2002, pp. 231–245. doi:10.1007/978-1-4613-0037-3_13.
- 310 [38] V. Aizinger, C. Dawson, A discontinuous Galerkin method for two-dimensional flow and transport in shallow water, *Advances in Water Resources* 25 (1) (2002) 67–84. doi:10.1016/S0309-1708(01)00019-7.
- [39] C. Dawson, V. Aizinger, A discontinuous Galerkin method for three-dimensional shallow water equations, *Journal of Scientific Computing* 22 (1–3) (2005) 245–267. doi:10.1007/s10915-004-4139-3.
- 315 [40] V. Aizinger, J. Proft, C. Dawson, D. Pothina, S. Negusse, A three-dimensional discontinuous Galerkin model applied to the baroclinic simulation of Corpus Christi Bay, *Ocean Dynamics* 63 (1) (2013) 89–113. doi:10.1007/s10236-012-0579-8.
- [41] P. Roe, Approximate riemann solvers, parameter vectors, and difference schemes, *J. Comput. Phys.* 43 (1981) 357–372.
- [42] P. Roe, J. Pike, Efficient construction and utilisation of approximate riemann solutions, in: R. Glowinski, L. JL (Eds.), *Computing Methods in Applied Sciences and Engineering*, 1984, pp. 499–518.
- 320 [43] S. Gottlieb, C.-W. Shu, E. Tadmor, Strong stability-preserving high-order time discretization methods, *SIAM Review* 43 (2001) 89–112. doi:10.1137/S003614450036757X.
- [44] F. Frank, B. Reuter, *FESTUNG: The Finite Element Simulation Toolbox for UNstructured Grids* (2020). URL <https://github.com/FESTUNG>
- 325 [45] V. Aizinger, C. Dawson, B. Cockburn, P. Castillo, *The local discontinuous Galerkin method for contaminant transport*, *Advances in Water Resources* 24 (1) (2000) 73–87. doi:10.1016/S0309-1708(00)00022-1. URL <http://www.sciencedirect.com/science/article/pii/S0309170800000221>
- [46] O. C. Zienkiewicz, P. Ortiz, A split-characteristic based finite element model for the shallow water equations, *International Journal for Numerical Methods in Fluids* 20 (8–9) (1995) 1061–1080. doi:10.1002/flid.1650200823.
- 330 [47] A. Ippen, High-velocity flow in open channels: A symposium: Mechanics of supercritical flow, *Transactions of the American society of Civil Engineers* 116 (1) (1951) 268–295.
- [48] C. B. Vreugdenhil, *Numerical Methods for Shallow-Water Flow*, Springer Science & Business Media, 1994. doi:10.1007/978-94-015-8354-1.
- [49] S. Faghih-Naini, S. Kuckuk, V. Aizinger, D. Zint, R. Grosso, H. Köstler, Quadrature-free discontinuous Galerkin method with code generation features for shallow water equations on automatically generated block-structured meshes, *Advances in Water Resources* 138 (2020) 103552. doi:10.1016/j.advwatres.2020.103552.
- 335 [50] H. Hajduk, D. Kuzmin, V. Aizinger, Bathymetry reconstruction using inverse shallow water models: Finite element discretization and regularization, in: H. van Brummelen, A. Corsini, S. Perotto, G. Rozza (Eds.), *Numerical Methods for Flows: FEF 2017 Selected Contributions*, Springer International Publishing, Cham, 2020, pp. 223–230. doi:10.1007/978-3-030-30705-9_20.