

# WAVEx: Stabilized finite elements for spectral wind wave models using FEniCSx

Mark Loveland <sup>a,\*</sup>, Eirik Valseth <sup>a,b,c</sup>, Jessica Meixner <sup>d</sup>, Clint Dawson <sup>a</sup>

<sup>a</sup> Oden Institute for Computational Engineering and Sciences, University of Texas at Austin, 201 E. 24th St., Austin, TX, 78712, USA

<sup>b</sup> The Department of Data Science, The Norwegian University of Life Science, Drøbakveien 31, Ås, 1433, Norway

<sup>c</sup> Department of Scientific Computing and Numerical Analysis, Simula Research Laboratory, Kristian Augusts gate 23, Oslo, 0164, Norway

<sup>d</sup> National Oceanic and Atmospheric Administration (NOAA), NWS, NCEP, EMC, 830 University Research Court, College Park, MD, 20740, USA



## ARTICLE INFO

### Keywords:

Spectral wind wave model  
Stabilized finite element methods  
FEniCSx  
Python

## ABSTRACT

The prediction of the wind wave spectrum of the ocean using numerical models are an important tool for researchers, engineers, and communities living in coastal areas. The governing equation of the wind wave models, the Wave Action Balance Equation, presents unique challenges for implementing reliable numerical models because it is highly advective, highly nonlinear and high dimensional. Historically, most operational models have utilized finite difference methods, others have used finite volume methods but relatively few attempts at using finite element methods. In this work, we seek to fill this gap by investigating several different finite element discretizations of the Wave Action Balance Equation. The methods, which include streamline upwind Petrov-Galerkin (SUPG), least squares, and discontinuous Galerkin, are implemented and convergence properties are examined for some simplified 2-D test cases. Then, a new spectral wind wave model, WAVEx, is formulated and implemented for the full problem setting. WAVEx uses continuous finite elements along with SUPG stabilization in geographic/spectral space that allows for fully unstructured triangular meshes in both geographic and spectral space. For propagation in time, a second order fully implicit finite difference method is used. When source terms are active, a second order operator splitting scheme is used to linearize the problem. In the splitting scheme, propagation is solved using the implicit method and the nonlinear source terms are treated explicitly. Several test cases, including analytic tests and laboratory experiments, are demonstrated and results are compared to analytic solutions, observations, as well as output from another model that is used operationally.

## 1. Introduction

Wind waves are those that typically range in period from 0.25–30 s, or wave lengths of 0.1–1500 m, and are the product of disturbances of the ocean surface due to locally generated winds (Holthuijsen, 2007). Wind wave models take into consideration the local winds, bathymetry, and currents in order to approximate how wind waves evolve over time.

This research focuses specifically on spectral wind wave models that are defined through the Wave Action Balance Equation (WAE). These spectral wind wave models work by estimating the transport of wave action distributed among wind waves of various frequencies and directions as defined through the WAE (Komen et al., 1994; Monbaliu, 2003; Khandekar, 1989; Janssen, 2008; Young, 1999). The WAE presents challenges for numerical discretization techniques because it is a first order transient hyperbolic equation whose domain lies in 4 dimensions (2 in space, frequency, direction) with a non-divergent

free velocity field and possibly nonlinear source/sink terms. Numerical spectral wind wave models have been in operation since the 1970s and are used daily as a tool in many fields including engineering, shipping, and meteorology (Group, 1988; Tolman, 1991; Komen et al., 1994). They are also important tools for both improving our understanding of ocean processes and for predicting the impacts of severe weather events. The wind wave models can produce results regarding the state of the sea surface (sea state) such as average wave height, significant wave height, mean wave direction, peak period, and wave radiation stresses.

Currently, there are many well-known spectral wind wave models used in practice such as ECWAM (Group, 1988; Janssen, 2004), which is supported by the European Centre for Medium-Range Weather Forecasts (ECMWF), Simulating Waves Nearshore (SWAN) (Booij et al., 1999), which is supported by Delft University of Technology, and

\* Corresponding author.

E-mail address: [markloveland@utexas.edu](mailto:markloveland@utexas.edu) (M. Loveland).

URL: <https://www.researchgate.net/profile/Mark-Loveland-2> (M. Loveland).

WAVEWATCH III (Tolman, 1991), which is supported by the United States National Oceanic and Atmospheric Administration (NOAA). A brief summary of the methods employed in contemporary numerical spectral follows but we note that there are many great historical reviews on numerical spectral wind wave models see e.g., Roland (2008), Meixner (2013), Komen et al. (1994), Alves et al. (2007), Monbaliu (2003), Janssen (2008), Khandekar (1989) and Roland and Ardhuin (2014).

All numerical spectral wind wave models rely on numerical approximations to the governing partial differential equation (PDE), i.e., the WAE. Historically, spectral wind wave models used finite difference approximations along with splitting schemes to separate the propagation and source terms of the WAE (Alves et al., 2007). For instance, the ECWAM model contains both a first order explicit upwind scheme and a second order leapfrog method for the propagation terms with a semi-implicit second order centered difference for the source terms (Group, 1988). The original WAVEWATCH III model uses an explicit predictor-corrector method that approximates Crank–Nicholson and an explicit Euler scheme for the source terms (Tolman, 1991). The scheme of SWAN was unique at its time of inception in that it used a fully implicit solver without splitting nonlinear and linear terms using the ideas of a nonlinear solver by Patankar (1980). In SWAN, a finite difference formulation based on an implicit upwind scheme is used (Booij et al., 1999). This idea of using an implicit time step scheme is relatively uncommon in fluid problems such as the WAE but show great promise in applications where the CFL condition is particularly restrictive (Roland and Ardhuin, 2014).

It is also common when splitting the WAE into propagation and source to employ a method of characteristics for the propagation portion (Roland and Ardhuin, 2014). This is an intriguing method because without source terms the WAE is just an advection problem and the characteristics should give the exact solution. However, it can be difficult to use these methods for complex bathymetries and complicated source terms and thus numerical approximations are necessary in practice. Two popular models, STWAVE and TOMAWAC, were developed beginning in the 1990s use this characteristic paradigm and in many cases with great success (Smith et al., 2001; Benoit et al., 1997).

As the reliability and usage of wave models increased, there was a push for spectral wind wave models to be used in more shallow waters. Historically, structured grids are common because they are easier to implement and may be computationally less expensive than unstructured grids. However, in order for structured grids to work for coastal applications, very high levels of refinement are needed to capture often highly irregular changes in bathymetry (Roland and Ardhuin, 2014) as well as coastline. Thus, spectral wind wave models capable of unstructured grids became more popular because they are often computationally cheaper than refining a structured grid multiple levels.

The first known model capable of computations on unstructured meshes is the aforementioned TOMAWAC (Benoit et al., 1997), which was released in 1997 and was able to represent the geographic domain with unstructured triangles using what is often known as a characteristic-Galerkin or a Lagrange–Galerkin approach. That is, a method that uses finite element interpolation in combination with a method of characteristics. In Hsu et al. (2005), Hsu et al. developed a finite element method (FEM) version of SWAN that used a Taylor–Galerkin scheme in geographic space while splitting the spectral frequency, direction, and source terms. This model was further developed, focusing on improvements in eliminating oscillations around steep gradients, into wind wave model (WWM) I and WWM II, which added an implicit and explicit fluctuation splitting schemes (also known as residual distribution schemes) to the geographic space (Roland et al., 2006, 2009; Roland, 2012; Abgrall, 2006). This same approach was added into the WAVEWATCH III model, extending WAVEWATCH III to allow for unstructured meshes and implicit time steps (Abdolali et al., 2020). Around the same time, SWAN became available on unstructured

meshes made of triangular elements by extending its finite difference upwinding scheme (Zijlema, 2010).

In addition to the model of Hsu, there have been a couple more finite element methods applied to the WAE. In 2012, Yildirim et al. used a hybrid approach with discontinuous Galerkin in geographic space and Fourier collocation in spectral space to create a third generation model (Yildirim and Karniadakis, 2012). In 2014, Meixner used a discontinuous Galerkin FEM in spectral and geographic space (Meixner et al., 2014). Both of the models used Runge–Kutta type explicit time stepping schemes.

In addition to models that use finite difference methods, and FEMs, there are models that use finite volume methods. FVCOM-SWAVE is an implementation of a finite volume method that allows for unstructured grids (Qi et al., 2009). A summary of all of the aforementioned schemes are contained in Table 1. As can be seen in Table 1, as these operational wind wave models have increased capabilities for capturing the wave spectrum in coastal areas, there are a set of two common features amongst numerical schemes. The first is a capability of fully unstructured triangular grids in geographic space and the second is implicit time stepping. These are important features because unstructured meshes allow for refinement of the irregular bathymetry in the coast and the implicit time step allows for violation of the highly restrictive CFL condition that is inherent in the WAE. This research aims to adhere to these principles but from a finite element perspective.

This work focuses on the development and implementation of a new spectral wind wave model, called WAVEx, that is capable of using unstructured meshes via the FEM along with implicit time stepping using the open source finite element library, FENiCSx (McRae et al., 2016; Habera et al., 2020). FEMs gained popularity starting in the mid twentieth century as a means to numerically approximate differential equations see e.g., Zienkiewicz et al. (2005), Becker et al. (1981), Carey and Oden (1983) and Reddy (2019). In general a FEM consists of recasting a differential equation into an equivalent weak form, partitioning the associated domain into a finite number of subdomains (i.e., elements), approximating the solution on each element with a finite dimensional basis, and subsequently solving the associated linear algebraic system of equations to achieve an approximate solution. The FEM has many benefits in that it is backed by rigorous mathematical theory that readily extends to high order approximations, unstructured meshes, and adaptive meshing see e.g., Ern and Guermond (2013), Brenner and Scott (2008) and Zienkiewicz et al. (2005) for thorough introductions and complete discussions on advantages and weaknesses.

FEMs were originally developed for elliptic type problems, which often arise in structural mechanics (Brenner and Scott, 2008; Ern and Guermond, 2013). However, the standard (Bubnov–Galerkin) FEMs have issues when applied to many common fluid problems, which are often hyperbolic in nature, and has historically led to unstable solutions and solutions that had spurious oscillations (Johnson, 2012; Ern and Guermond, 2013; Reddy, 2019; Oden and Demkowicz, 2017). These problems have been thoroughly analyzed and many stabilized finite element techniques were developed (Johnson et al., 1984; Pironneau and Pironneau, 1989; Zienkiewicz et al., 2013) to address the issues. Some of the methods developed that became popular include artificial viscosity (Ern and Guermond, 2013), Taylor–Galerkin (Donea and Quartapelle, 1992), Streamlined Upwind Petrov–Galerkin (SUPG) (Brooks and Hughes, 1982), least squares FEM (Bochev and Gunzburger, 1998), discontinuous Galerkin (DG) (Dawson and Proft, 2001; Cockburn et al., 2000; Brezzi et al., 2006, 2004), and Lagrange–Galerkin or characteristic–Galerkin (Douglas and Russell, 1982; Bermejo et al., 2023; Morton et al., 1988; Dawson et al., 1994).

In the following study, a set of four finite element formulations of the WAE: SUPG, DG, least squares, and standard Bubnov–Galerkin, will be considered and implemented using the FENiCSx library in a simplified two-dimensional domain (1 in space and 1 in frequency) along with implicit finite difference approximation in time for all FEM schemes. Two test cases will be demonstrated, one of a propagating

**Table 1**  
Summary of numerical schemes in contemporary spectral wind wave models.

Model name	Year	Numerical method	Method description	Grid type
WAM	1988	Finite difference	Advection: Explicit 1st order upwind, leapfrog Source: Centered difference (semi-implicit)	Structured
WAVEWATCH III	1990	Finite difference	Advection: QUICKEST (Explicit 3rd order upwind) Source: Semi-implicit	Structured
STWAVE	1990s	Method of characteristics/Finite difference	Advection: Method of characteristics Source: Finite difference	Structured
TOMAWAC	1997	Method of characteristics/FEM	Advection: Method of characteristics Source: semi-implicit (linear finite elements for interpolation)	Unstructured
SWAN	1999, 2008	Finite difference	Space: Upwinding Time: First order implicit	Unstructured
WWMII/WW3	2009	Contour residual scheme	Space: Contour residual distribution Time: Implicit upwind	Unstructured
FVCOM-SWAVE	2009	Finite volume	FVCOM finite volume in space, centered difference in directional, flux corrected transport algorithm for frequency	Unstructured
Yildirim	2011	FEM/Spectral	Geographic space: Discontinuous Galerkin Spectral space: Fourier collocation Time: TVD Runge-Kutta	Unstructured
Meixner	2014	FEM	Space: Discontinuous Galerkin Time: SSP Runge-Kutta	Unstructured

sinusoidal signal and one with a developing shock. For each test case, the  $L^2$  and  $l^\infty$  errors are calculated and studied with respect to  $h$  refinement. Then, employing the strategies of Loveland et al. in Loveland et al. (2022), WAVEEx is implemented for the full four dimensional WAE. Thus, WAVEEx is a finite element model that employs SUPG stabilization with a second-order implicit time-stepping scheme (Crank–Nicolson) along with a Strang splitting approach to handle source terms in the WAE implemented using FEniCSx. The open-source framework FEniCSx was chosen, as its powerful open-source framework allows for automation of coding finite element solvers and in hopes that it will allow for users to more easily experiment with different numerical techniques or with different source terms via its Python interface.

Several test cases from the US Office of Naval Research (ONR) testbed (Ris et al., 2003) are reproduced in order to demonstrate the validity and applicability of WAVEEx. The collection of tests include theoretical problems with analytic solutions as well as one lab experiment that has physical measurements. To measure accuracy, the solution generated by WAVEEx is compared to the tabulated results at the given points. From these comparison points, both the root mean square error (RMSE), i.e.,  $l^2$  and the  $l^\infty$  error are computed. For each test case, the parameters include significant wave height and mean wave direction and are computed in WAVEEx. The set of test cases were selected to investigate convergence properties, accuracy, and stability of our numerical schemes as well as demonstrate the scalability of the schemes. To the knowledge of the authors, no finite element methods have been used for the WAE that use SUPG stabilization, least squares FEM, or discontinuous Galerkin approach along with implicit time stepping. Furthermore, to the knowledge of the authors, no full scale wave model has been implemented completely within Python. The use of Python we believe will increase the code's readability as well as ease future collaborations and further development.

## 2. Methods

### 2.1. Problem definition

The Wave Action Balance Equation (WAE) is derived as a conservation law of wave action,  $N$ , which is a scalar-valued function of horizontal geographical space ( $x, y$ ) and spectral space ( $\sigma, \theta$ ), i.e., frequency and direction, and can be found in e.g., Holthuijsen (2007),

Young (1999), Komen et al. (1994) and LeBlond and Mysak (1981). Assuming source terms are linear with respect to  $N$ , the WAE is a linear, scalar-valued, hyperbolic equation in 4 dimensions with a varying (both in 4-D space and time), non-divergence free velocity field,  $(\mathbf{c}(x, y, \sigma, \theta))$ , which can be determined independently of the unknown  $N(x, y, \sigma, \theta)$ . This is different than most conservation laws such as Navier–Stokes and related transport equations because typically the propagation velocity is just a function of the unknown i.e.  $\mathbf{c}(N)$ .

The proper boundary and initial conditions are as follows, assuming we have a bounded domain in four-dimensional space  $\Omega \subset \mathbb{R}^4$  that is sufficiently regular. Similar to other advection problems, the boundary is split up into 2 segments, inflow and outflow:

$$\Gamma_- = x \in \partial\Omega : \mathbf{c} \cdot \mathbf{n} < 0 = \text{inflow} \quad (1)$$

$$\Gamma_+ = x \in \partial\Omega : \mathbf{c} \cdot \mathbf{n} \geq 0 = \text{outflow},$$

where  $\mathbf{n}$  is the outwards unit normal vector to the boundary. Then it has been shown that the following problem possesses a unique solution (LeVeque and Leveque, 1992; Donea and Quartapelle, 1992; Pironneau and Pironneau, 1989):

$$N_t + \nabla \cdot (\mathbf{c}N) = \frac{S(N, x, y, \sigma, \theta, t)}{\sigma} \quad \text{on } \Omega \times (0, T), \\ N = N_- \quad \text{on } \Gamma_-, \\ N = N_0 \quad \text{on } \Omega \quad \text{at } t = 0, \quad (2)$$

where  $N_t$  denotes the (partial) time derivative of  $N$ ,  $S$  the source/sink terms,  $\Omega$  the computational domain, and  $N_-$  the specified essential boundary condition on the outflow boundary. In the case when  $S$  is either 0 or is independent of  $N$ , a fairly straightforward analytic solution to the above problem can be obtained via the method of characteristics (Donea and Quartapelle, 1992). However, in practice  $S$  is quite complex, often nonlinear, and realistic problems become too complex to analyze with pencil and paper and thus the need for numerical methods to approximately solve (2) arises.

The propagation velocities,  $\mathbf{c}$ , are all determined using the constitutive relation called the dispersion relation, which is derived under assumptions of Airy Wave Theory. The dispersion relation relates relative radial frequency,  $\sigma$ , to wavenumber magnitude,  $k$ :

$$\sigma^2 = gk \tanh(kd). \quad (3)$$

Here  $g$  is the constant of gravitational acceleration and  $d$  is total water depth. The velocity  $\mathbf{c}$  is a non-constant, non-divergence free vector

quantity defined as:

$$\mathbf{c} = (c_g \cos\theta + u, c_g \sin\theta + v, c_\sigma, c_\theta), \quad (4)$$

where  $u, v$  are the mean water velocities in  $x, y$  directions respectively,  $c_g$  the relative group velocity (defined as  $\frac{\partial\sigma}{\partial k}$ , whereas the absolute group velocity is  $\frac{\partial\omega}{\partial k} = c_g + \|(u, v)\|$ ), and  $c_\sigma, c_\theta$  define the advection of the spectra with respect to direction and relative frequency, respectively. The group velocity,  $c_g$ , can be directly obtained by differentiating the dispersion relation in (3) for relative frequency,  $\sigma$ , with respect to wavenumber magnitude  $k$ . The explicit expression is:

$$c_g = \frac{1}{2} \left( 1 + \frac{2kd}{\sinh 2kd} \right) \sqrt{\frac{g}{k} \tanh(kd)}. \quad (5)$$

The propagation velocity  $c_\sigma$ , represents frequency shifting due to changes in depths and currents. By applying the chain rule, we can find an expression of  $c_\sigma$ :

$$\begin{aligned} c_\sigma = \frac{d\sigma}{dt} &= \frac{k\sigma}{\sinh(2kd)} \left( \frac{\partial d}{\partial t} + u \frac{\partial d}{\partial x} + v \frac{\partial d}{\partial y} \right) - \\ &c_g k \left( \frac{\partial u}{\partial x} \cos^2(\theta) + \frac{\partial u}{\partial y} \cos(\theta) \sin(\theta) + \frac{\partial v}{\partial x} \sin(\theta) \cos(\theta) \right. \\ &\left. + \frac{\partial v}{\partial y} \cos^2(\theta) \right). \end{aligned} \quad (6)$$

The full derivation is quite lengthy but can be found in Appendix D in Holthuijsen (2007). Now the propagation velocity  $c_\theta$ , with respect to  $\theta$ , represents the shift in the spectrum due to refraction and diffraction. Similarly to  $c_\sigma$ , the derivation is quite lengthy but can be obtained by applying the chain rule:

$$\begin{aligned} c_\theta = \frac{d\theta}{dt} &= \frac{\sigma}{\sinh(2kd)} \left( \frac{\partial d}{\partial x} \sin(\theta) - \frac{\partial d}{\partial y} \cos(\theta) \right) + \\ &\frac{\partial u}{\partial x} \cos(\theta) \sin(\theta) - \frac{\partial u}{\partial y} \cos^2(\theta) + \frac{\partial v}{\partial x} \sin^2(\theta) - \frac{\partial v}{\partial y} \cos(\theta) \sin(\theta). \end{aligned} \quad (7)$$

## 2.2. Source terms

In the WAE (2), the source term  $S(N, x, y, \sigma, \theta, t)$  was left arbitrary. However, in practice, it can take many forms but in general it can be thought of as a sum of three key sources/sinks:

$$S = S_{in} + S_{diss} + S_{nl}. \quad (8)$$

where  $S_{in}$  represents any contribution to the spectrum due to wind input,  $S_{diss}$  represents any change in the energy spectrum due to dissipation that can include things like whitecapping and surf zone breaking, and  $S_{nl}$  represents changes in the spectrum due to nonlinear wave interactions. When source terms are active, the WAVEx model in this study will use the default 3rd generation source term package used in SWAN cycle III Version 41.41, see details in Team et al. (2022). The default wind input term is that from WAM Cycle III and first outlined in Komen et al. (1984), the dissipation from whitecapping, bottom friction, and depth-induced breaking are included and the DIA approximation is used to account for nonlinear interactions. More details on these source terms can be found in Chapter 3 of Loveland (2023).

## 2.3. Weak formulations for the wave action balance equation

In this section, several weak forms of the WAE will be presented. These weak forms will eventually lead to fully discrete systems that are then solved with FEniCSx. First, for comparison purposes, two forms of the standard Bubnov–Galerkin approach will be derived for the WAE and the problems with the standard Bubnov–Galerkin discretizations will be discussed. Three alternatives to the standard Bubnov–Galerkin approach: the least squares FEM, the SUPG method, and the DG method will be derived and discussed. All of these weak forms will then be run on two simplified test cases using FEniCSx in Section 3 in order to investigate convergence properties of all the stabilized finite element

methods with respect to  $h$  refinement. The full discretization of an SUPG method will be described for the full four-dimensional setting in the subsequent section. Note that we utilize inner product notation for integrals, i.e.,  $(u, v)_\Omega = \int_\Omega uv \, d\Omega$ .

### 2.3.1. Bubnov–Galerkin

Taking the strong form of the WAE as in (2), we obtain an equivalent problem but in the weak form by multiplying both sides by a test function  $w \in L^2(\Omega)$  and integrating over a bounded and Lipschitz domain  $\Omega \subset \mathbb{R}^4$ . Note that we do not integrate over time because we will eventually employ finite difference approximation in this dimension. Thus we have the following weak form that we will call CG Strong because there are more regularity requirements on the trial space. Find  $N \in U$  such that:

$$\left( \frac{\partial N}{\partial t}, w \right)_\Omega + (\nabla \cdot (\mathbf{c}N), w)_\Omega = \left( \frac{S}{\sigma}, w \right)_\Omega \quad \forall w \in L^2(\Omega). \quad (9)$$

Thus, it is required that  $U = \{N \in H^1(\Omega, t); N(\Gamma_-, t) = N_-; \frac{\partial N}{\partial t} \in L^2(\Omega, t)\}$ , recalling that  $\Gamma_-$  is the inflow boundary and  $N_-$  is the Dirichlet condition on that part of the boundary. We will also assume that velocity vector  $\mathbf{c}$  and source term  $S$  is sufficiently well behaved for the weak form to have meaning. It can be shown that (9) is well-posed (in the sense that a solution exists, it is unique, and continuously depends on the data) see e.g., the book of Ern and Guermond (Ern and Guermond, 2013). However, it is also known that if (9) is discretized using the standard Bubnov–Galerkin approach then the discrete inf-sup condition depends on mesh size,  $h$ , and therefore does not provide optimal error estimates (Ern and Guermond, 2013). By optimal, we mean for polynomial approximations of order  $k$  the convergence of the error in the  $L^2$  sense to of the same order of convergence as the underlying FE interpolants, usually  $h^{k+1}$  (this classical result can be found in e.g., Babuska et al. (2010), Brenner and Scott (2008), Reddy (2019) and Oden and Demkowicz (2017)). In practice this leads to oscillatory solutions because only  $L^2$  stability can be shown (Johnson et al., 1984; Zienkiewicz et al., 2005). Also, the standard Bubnov–Galerkin approximation in this setting often leads to ill-conditioned systems of equations and can even lead to divergence in the finite element solution especially if the exact solution is non-smooth (Zienkiewicz et al., 2005; Johnson, 2012).

We can also derive a second weak form if we integrate by parts, which we will call the CG form, find  $N \in U$ :

$$\left( \frac{\partial N}{\partial t}, w \right)_\Omega - (\mathbf{c}N, \nabla w)_\Omega + (\mathbf{c}N \cdot \mathbf{n}, w)_{\Gamma_+} = \left( \frac{S}{\sigma}, w \right)_\Omega \quad \forall w \in H^1(\Omega). \quad (10)$$

In order for the term on the boundary to have meaning we must require additional regularity than just  $N \in L^2(\Omega)$  and so the analysis is not so straightforward but has been shown to be well posed for cases of homogeneous boundary conditions and in cases with lower dimensionality (Ern et al., 2021). When discretized, similar problems can be expected to that of the strong CG from in (9). In order to mitigate the problems inherent in these standard Bubnov–Galerkin discretizations, we will investigate a least squares method, an SUPG method, and a DG method.

### 2.3.2. Least squares

The least squares approach is fundamentally different to the Galerkin approach in that the least squares method is posed as a minimization problem of a linear differential operator. A good reference on the properties of least squares can be found in the text of Bochev and Gunzburger (2006). To derive the least squares form for the WAE, we start with the strong form as in (2), we define the differential operator  $\mathcal{L}$  and right hand side as in Bochev and Gunzburger (2006) so that the above equation  $\mathcal{L}$  would be defined:

$$\mathcal{L} \bullet = \frac{\partial \bullet}{\partial t} + \nabla \cdot (\mathbf{c} \bullet). \quad (11)$$

Then the least squares variational form then is defined as follows. Find  $N \in U$ :

$$(\mathcal{L}N, \mathcal{L}w)_\Omega = \left(\frac{S}{\sigma}, \mathcal{L}w\right)_\Omega \quad \forall w \in U. \quad (12)$$

This problem is equivalent to minimizing the functional  $J(V) = \frac{1}{2}(\mathcal{L}(V), \mathcal{L}(V))_\Omega - (\mathcal{L}V, S/\sigma)_\Omega$ . Explicitly writing out the definition of  $\mathcal{L}$  gives the formulation for the WAE will be, find  $N \in U$ :

$$\left(\frac{\partial N}{\partial t} + \nabla \cdot (\mathbf{c}N), \nabla \cdot (\mathbf{c}w)\right)_\Omega = \left(\frac{S}{\sigma}, \nabla \cdot (\mathbf{c}w)\right)_\Omega \quad \forall w \in U, \quad (13)$$

where  $U = \{N \in H^1(\Omega, t); N(\Gamma_-, t) = N_-; \frac{\partial N}{\partial t} \in L^2(\Omega)\}$ , and again  $\mathbf{c}, S$  are assumed sufficiently regular to give the integrals meaning. We note that the time derivative on the test space will be omitted because the time derivative will be discretized via finite difference and not finite elements.

The least squares problem generates a symmetric problem (trial and test spaces are the same) and thus when discretized exhibits different properties than the Bubnov–Galerkin approximations. It has been shown that for similar problems, such as the advection–reaction equation, the least squares method can reduce oscillations and provide stability (Ern and Guermond, 2013). It can be shown that least squares produces optimal error convergence in the streamline directions however it has yet to be shown that the least squares method produces optimal errors with respect to  $h$  refinement in the  $L^2$  norm (Ern and Guermond, 2013).

### 2.3.3. SUPG

The next stabilized form that will be investigated is the SUPG method. Because the WAE is an advection dominated equation, the method is a natural choice to employ. The method originally from Brooks and Hughes (1982) was designed for fluid problems and has had much success in stabilizing advection dominated problems for fluids by adding diffusion in the upwind direction in the weak formulation. The SUPG scheme is formulated by adding a term with a vanishing residual to the standard Bubnov–Galerkin form in the discrete section. This in turn modifies the discrete test space, which results in the change from a standard Bubnov–Galerkin form to a so called Petrov–Galerkin form. We can add this residual term to either CG strong as in (9) or the CG form as in (10). We will demonstrate both possible SUPG formulations now. We start with CG strong form and taking it into the discrete setting where the test and trial functions are piecewise smooth and globally continuous. We will call the resulting form SUPG strong, which is find  $N_h \in U_h(\Omega)$ :

$$\begin{aligned} & \left(\frac{\partial N_h}{\partial t} + \nabla \cdot (\mathbf{c}N_h), w_h\right)_\Omega + \left(\frac{\partial N_h}{\partial t} + \nabla \cdot (\mathbf{c}N_h) - \frac{S}{\sigma}, \tau \mathbf{c} \cdot \nabla(w_h)\right)_{\Omega_e} \\ &= \left(\frac{S}{\sigma}, w_h\right)_\Omega \quad \forall w_h \in V_h(\Omega). \end{aligned} \quad (14)$$

Here  $\Omega_e$  is the interior over each finite element. Because in the analytic setting the residual term is identically 0, then this problem inherits the well-posedness from (9). The weak form can be simplified by adding the integrals together:

$$\begin{aligned} & \left(\frac{\partial N_h}{\partial t} + \nabla \cdot (\mathbf{c}N_h), w_h + \tau \mathbf{c} \cdot \nabla(w_h)\right)_{\Omega_e} = \left(\frac{S}{\sigma}, w_h + \tau \mathbf{c} \cdot \nabla(w_h)\right)_{\Omega_e} \\ & \forall w_h \in V_h(\Omega). \end{aligned} \quad (15)$$

This will be called the SUPG strong form that will be discretized with continuous finite elements in FEniCSx. It has been shown that adding this diffusion in the streamline direction improves the stability of the discrete problem by improving the discrete inf-sup condition (see Johnson et al. (1984) and Brooks and Hughes (1982)) and allowing for discrete stability in the  $H^1$  sense. Furthermore, error estimates for very similar problems to the WAE, such as the advection–reaction equation, show nearly optimal rates at  $O(h^{k+0.5})$  (Johnson et al., 1984; Burman, 2010; Burman and Smith, 2011).

We can also arrive at an SUPG form by taking (10) into the discrete setting and adding the residual to get, find  $N_h \in U_h(\Omega)$ :

$$\begin{aligned} & \left(\frac{\partial N_h}{\partial t}, w_h\right)_\Omega - (\mathbf{c}N_h, \nabla w_h)_\Omega + (\mathbf{c}N_h \cdot \mathbf{n}, w_h)_{\Gamma_+} + \\ & \left(\frac{\partial N_h}{\partial t} + \nabla \cdot (\mathbf{c}N_h) - \frac{S}{\sigma}, \tau \mathbf{c} \cdot \nabla(w_h)\right)_{\Omega_e} = \left(\frac{S}{\sigma}, w_h\right)_\Omega \quad \forall w_h \in V_h(\Omega). \end{aligned} \quad (16)$$

After discretization of both the SUPG strong and SUPG forms,  $\tau$  is taken from Codina (2000), Burman (2010) and is chosen to be:

$$\tau = \frac{h}{\|\mathbf{c}\|}, \quad (17)$$

where  $h$  is the characteristic length of an element.

### 2.3.4. Discontinuous Galerkin

The next stabilized form that will be examined is the DG method as in the work of Meixner et al. (2014). The DG method is different to any of the other methods discussed up to this point in that it allows for discontinuities between element interfaces. The DG formulation is obtained by taking the strong form of the WAE, multiplying by a test function, and integrating by parts over each element. Find  $N_h \in U_h$ :

$$\begin{aligned} & \left(\frac{\partial N_h}{\partial t}, w_h\right)_{\Omega_e} - (\mathbf{c}N_h, \nabla w_h)_{\Omega_e} + (\mathbf{c}\hat{N}_h \cdot \mathbf{n}, w_h)_{\partial\Omega_e} = \left(\frac{S}{\sigma}, w_h\right)_{\Omega_e} \\ & \forall w_h \in W_h. \end{aligned} \quad (18)$$

The test and trial spaces are virtually the same with  $W_h = \{v \in L^2(\Omega) : v|_{\Omega_e} \in P^k(\Omega_e) \forall \Omega_e \in T_h\}$  and  $U_h$  being  $W_h$  but with a finite energy lift on the inflow boundary.  $P^k$  are polynomials of degree  $k$ , and  $\Omega_e$  corresponds to the interior of an element while  $T_h$  is a finite element partition of the domain  $\Omega$ .  $\hat{N}_h$  is a uniquely defined flux on the boundary, which is required to form a closed system of equations (see Meixner et al. (2014), Cockburn et al. (2000) and Dawson and Proft (2001)). In this implementation, the fluxes on the element boundaries are uniquely defined in this case using the Lax–Friedrichs flux, which is described in detail in the work of Meixner et al. (2014). The DG method has well documented analysis on stability and error estimation for hyperbolic problems such as the advection reaction equation, which is essentially what the Wave Action Balance Equation is Cockburn et al. (2000), Brezzi et al. (2004) and Brezzi et al. (2006). It has been shown that for a DG scheme like the one above, stability in the  $H^1$  sense exists and that almost optimal convergence,  $O(h^{k+\frac{1}{2}})$  can be shown (Brezzi et al., 2004).

## 2.4. WAVEx: Implementation of wave action balance equation solver with FEniCSx

We now introduce WAVEx, which is capable of solving the WAE in its full 4-dimensional spatial domains using the techniques described in Loveland et al. (2022), and is implemented with the finite element library FEniCSx. WAVEx in its current state allows for SUPG stabilization along with Strang splitting to deal with the nonlinear source terms. The codebase for WAVEx is publicly available and on Github at <https://github.com/Markloveland/WAVEx>. This section will explain in detail the numerics currently available in WAVEx including the finite element schemes as well as the time stepping schemes.

### 2.4.1. Semi-discrete form

This particular implementation in WAVEx will use a continuous FEM approach along with SUPG stabilization. We first limit ourselves to cases where the global domain is a Cartesian product between two subdomains (geographic space, and spectral space). This means we can construct the product basis:

$$N(x, y, \sigma, \theta) \approx \sum_{i=1}^N \sum_{j=1}^M N_{i,j} \phi_i(x, y) \psi_j(\sigma, \theta), \quad (19)$$

where  $N_{i,j}$  are the scalar weights of the interpolating functions.

As we know, the standard Bubnov–Galerkin approximation of advective equations leads to ill conditioned discrete problems and as in Section 2.3, we will use SUPG stabilization. SUPG was chosen because it performed well in our preliminary 2D studies (see Section 3), and the fact that it is also computationally cheapest. We begin by first recalling the two possible weak forms as described in Section 2.3, SUPG strong and SUPG. The latter being when integration by parts is conducted and the former is when integrating by parts is omitted. Both options are included in the WAVEEx code repository. For the form without integration by parts, SUPG strong, we begin with (15) and substitute in the product basis to get the semi-discrete (i.e., discrete in space, infinite dimensional in time) system:

$$\left(\frac{\partial(\phi_i \psi_j N_{i,j})}{\partial t} + \nabla \cdot (\mathbf{c} \phi_i \psi_j N_{i,j}) - \phi_i \psi_j S_{i,j}, \gamma_k \beta_l + \tau \mathbf{c} \cdot \nabla(\gamma_k \beta_l)\right)_\Omega = 0. \quad (20)$$

The source terms,  $S$  in general are quite complex, so we will only seek the  $L^2$  projections of their exact form as shown in (20). Analogously, the SUPG weak form of the WAE (16) in the semidiscrete setting is defined by substituting in our product basis into (16) to get:

$$\begin{aligned} \left(\frac{\partial(\phi_i \psi_j N_{i,j})}{\partial t}, \gamma_k \beta_l\right)_\Omega - (\mathbf{c} \phi_i \psi_j N_{i,j}, \nabla(\gamma_k \beta_l))_\Omega + (\mathbf{c} \phi_i \psi_j \cdot \mathbf{n} N_{i,j}, \gamma_k \beta_l)_{\partial\Omega} + \\ \left(\frac{\partial(\phi_i \psi_j N_{i,j})}{\partial t} + \nabla \cdot (\mathbf{c} \phi_i \psi_j N_{i,j}) - \phi_i \psi_j S_{i,j}, \tau \mathbf{c} \cdot \nabla(\gamma_k \beta_l)\right)_\Omega \\ = (\phi_i \psi_j S_{i,j}, \gamma_k \beta_l)_\Omega. \end{aligned} \quad (21)$$

This takes care of the terms not related to the time step, the choice of finite difference approximation in time will dictate the full discretization.

#### 2.4.2. Time step choice and fully discrete form

WAVEEx uses a generalized one step finite difference approximation for the time derivative in (20) and (21):

$$\begin{aligned} \left(\frac{\partial(\phi_i \psi_j N_{i,j})}{\partial t}, \gamma_k \beta_l + \tau \mathbf{c} \cdot \nabla(\gamma_k \beta_l)\right)_\Omega = F(N, t) \rightarrow \\ \left(\frac{\phi_i^{n+1} \psi_j^{n+1} N_{i,j}^{n+1} - \phi_i^n \psi_j^n N_{i,j}^n}{\Delta t}, \gamma_k \beta_l + \tau \mathbf{c} \cdot \nabla(\gamma_k \beta_l)\right)_\Omega = \alpha_t F(N, t^{n+1}) \\ + (1 - \alpha_t) F(N, t^n). \end{aligned} \quad (22)$$

where term  $F(N, t)$  is shorthand from (20) and equals:

$$F(N, t) = -(\nabla \cdot (\mathbf{c} \phi_i \psi_j N_{i,j}) - \phi_i \psi_j S_{i,j}, \gamma_k \beta_l + \tau \mathbf{c} \cdot \nabla(\gamma_k \beta_l))_\Omega, \quad (23)$$

and for (21):

$$\begin{aligned} F(N, t) = (\mathbf{c} \phi_i \psi_j N_{i,j}, \nabla(\gamma_k \beta_l))_\Omega - (\mathbf{c} \phi_i \psi_j \cdot \mathbf{n} N_{i,j}, \gamma_k \beta_l)_{\partial\Omega} \\ + (\phi_i \psi_j S_{i,j}, \gamma_k \beta_l)_\Omega \\ - (\nabla \cdot (\mathbf{c} \phi_i \psi_j N_{i,j}) - \phi_i \psi_j S_{i,j}, \tau \mathbf{c} \cdot \nabla(\gamma_k \beta_l))_\Omega, \end{aligned} \quad (24)$$

the superscripts represent the discrete level in time,  $n$  will be the previous time step and  $n+1$  is the point in time to be solved for, and  $\alpha_t$  is a time step parameter between 0 to 1. The scheme is implicit unless  $\alpha_t = 0$ , as  $\alpha_t = 0$  would be the first order explicit Euler approximation,  $\alpha_t = 1$  would be the first order implicit Euler approximation, and  $\theta = 0.5$  would be the second order Crank–Nicolson scheme. The terms with superscript  $n$  are known and move to the RHS while the terms  $n+1$  will be part of the global stiffness matrix at each time step.

The definition given for the finite difference approximation to the time derivative leads to a fully discrete system described by (22) depending on the choice of  $F(N, t)$ . Choosing (23) will yield the fully discrete system for SUPG strong (no integration by parts) and choosing (24) will give the fully discrete system for SUPG weak. Once a choice of finite basis is made for trial and test spaces, the forms above will yield a system of equations that in turn can be solved for the action balance at the next time step,  $N_{i,j}^{n+1}$ . However, there are two difficulties that need to be addressed in order to reach a final implementation: (i),

FEniCS(x) cannot evaluate these integrals on the entire 4-dimensional domain  $\Omega$ . In the next section, the discrete equations in (22) will be broken down into products of the geographic and spectral subdomains using the product rule as in Loveland et al. (2022) into a discrete form that is implementable in FEniCSx. (ii), in general the source term  $S$  is nonlinear with respect to  $N$ . This means that the discrete form of (22) will not yield a system of linear equations as long as we are using an implicit time step,  $\alpha_t > 0$ . To avoid this, an operator splitting method will be described in Section 2.4.4 in order to avoid having to solve a nonlinear system. Although in the future a fully implicit solver for the nonlinear problem is of great interest.

#### 2.4.3. FEniCSx compatible format

Now because we want to only evaluate functions in the 2-dimensional subdomains with FEniCSx we need to take advantage of the product rule. As an example, starting with the second term from (21) we get:

$$-(\mathbf{c} \phi_i \psi_j, \nabla(\gamma_k \beta_l))_\Omega = -(\mathbf{c} \phi_i \psi_j, \gamma_k \nabla(\beta_l) + \beta_l \nabla(\gamma_k))_\Omega. \quad (25)$$

Similar to the previous examples, using the property that the basis functions are only non-constant in their respective subdomains we get the following simplifications ( $\nabla_1$  denotes gradient operator in the geographic subdomain, and  $\nabla_2$  denotes the gradient operator in the spectral subdomain, and same for velocity):

$$-(\mathbf{c} \phi_i \psi_j, \gamma_k \nabla(\beta_l) + \beta_l \nabla(\gamma_k))_\Omega = -(\mathbf{c}_1 \phi_i \psi_j, \beta_l \nabla_1(\gamma_k))_\Omega - (\mathbf{c}_2 \phi_i \psi_j, \gamma_k \nabla_2(\beta_l))_\Omega. \quad (26)$$

This importantly allows us to pull integrands apart in the following fashion so that we can use FEniCSx:

$$\begin{aligned} -(\mathbf{c}_1 \phi_i \psi_j, \beta_l \nabla_1(\gamma_k))_\Omega - (\mathbf{c}_2 \phi_i \psi_j, \gamma_k \nabla_2(\beta_l))_\Omega = - \int_{\Omega_2} \psi_j \beta_l \\ \times \int_{\Omega_1} \phi_i \mathbf{c}_1 \cdot \nabla_1 \gamma_k dx dy \\ - \int_{\Omega_2} \psi_j \nabla_2 \beta_l \cdot \int_{\Omega_1} \mathbf{c}_2 \phi_i \gamma_k dx dy. \end{aligned} \quad (27)$$

Where (27) represents a term that can be evaluated using FEniCSx. Integrands inside  $\Omega_1$  each create a  $N_{dof1} \times N_{dof1}$  sparse matrix and to form the full discrete system we will need a matrix at each quadrature point in the second subdomain. Each matrix will be different because of the changes in  $\mathbf{c}$  in both subdomains. The remainder of the terms in (22) can be rewritten in this way via the product rule. The full process is the same can be seen in detail in Appendix.

#### 2.4.4. Operator splitting

The source terms are in general non-linear, in the interest of saving computational complexity we will employ a splitting scheme to handle the source terms of the WAE. More extensive notes on operator splitting can be found in Glowinski et al. (2016). We will be employing the second order Strang splitting (Strang, 1968) scheme, which can be defined on a system of ODEs as follows:

$$\begin{aligned} \frac{dX}{dt} + A(X, t) = 0, \in (0, T), \\ X(0) = X_0. \end{aligned} \quad (28)$$

In our case  $A$  will be the discrete finite element operator in 4D space and  $X$  will be the action balance at all of the degrees of freedom at a particular point in time. The Strang splitting scheme splits up  $A$  as a sum of two operators  $A_1$  and  $A_2$  for  $n \geq 0$ ,  $X^n \rightarrow X^{n+\frac{1}{2}} \rightarrow \hat{X}^{n+\frac{1}{2}} \rightarrow X^{n+1}$ . Solve:

$$\begin{aligned} \frac{dX_1}{dt} + A_1(X_1, t) = 0, t \in (t^n, t^n + \frac{1}{2}), \\ X_1(t^n) = X^n, \end{aligned} \quad (29)$$

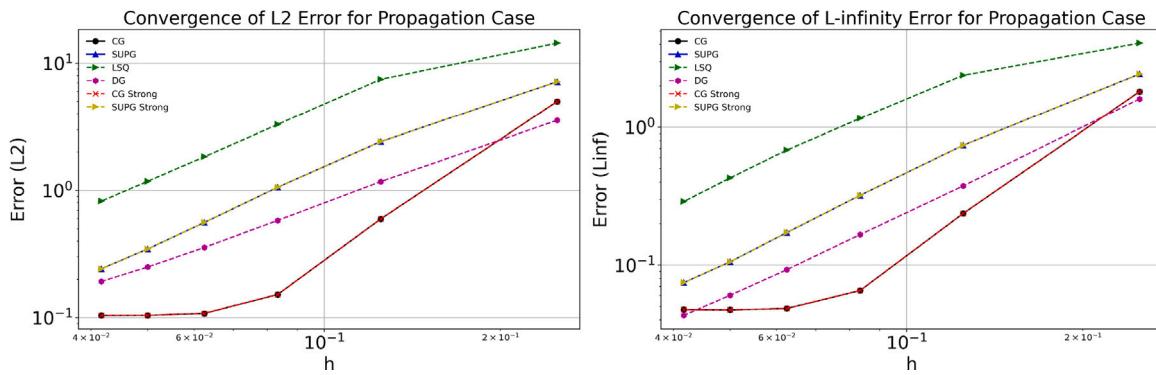


Fig. 1.  $L^2$  and  $l^\infty$  errors at  $T = 5$  for all schemes for case 1.

then set  $X^{n+\frac{1}{2}} = X_1(t^{n+\frac{1}{2}})$  and solve:

$$\frac{dX_2}{dt} + A_2(X_2, t) = 0, t \in (t^n, t^n + \Delta t), \quad (30)$$

$$X_2(t^n) = X^{n+\frac{1}{2}},$$

then set  $\hat{X}^{n+\frac{1}{2}} = X_2(\Delta t)$  and solve:

$$\frac{dX_1}{dt} + A_1(X_1, t) = 0, t \in (t^{n+\frac{1}{2}}, t^{n+1}), \quad (31)$$

$$X_1(t^{n+\frac{1}{2}}) = \hat{X}^{n+\frac{1}{2}}.$$

This gives  $X^{n+1} = X_1(t^{n+1})$ . We will apply this second order Strang splitting to the WAE by setting  $A_1$  to be the source terms and  $A_2$  as all remaining terms. In particular, for the case of the semi-discretization in (20), the operators will be:

$$A_1 = -(\phi_i \psi_j S_{i,j}, \gamma_k \beta_l + \tau \mathbf{c} \cdot \nabla(\gamma_k \beta_l))_{\Omega}, \quad (32)$$

$$A_2 = (\nabla \cdot (\mathbf{c} \phi_i \psi_j N_{i,j}), \gamma_k \beta_l + \tau \mathbf{c} \cdot \nabla(\gamma_k \beta_l))_{\Omega}.$$

We will implement the Strang splitting with an implicit-explicit scheme. The time step associated with operator  $A_2$  will be the scheme from Section 2.4.2, whereas the two time steps associated with operator  $A_1$  will be an explicit second order Runge-Kutta (RK2) scheme (Gottlieb et al., 2017). The RK2 used in WAVEx is defined as:

$$N^{(1)} = N^n + \Delta t A_1(N^n, t^n), \quad (33)$$

$$N^{n+1} = \frac{1}{2} N^n + \frac{1}{2} (N^{(1)} + \Delta t A_1(N^{(1)}, t^{n+1})).$$

To summarize, WAVEx utilizes SUPG stabilization in 4 dimensions and employs Strang splitting to separate advection from the source terms. The source terms are integrated in time with an explicit RK2 scheme while the advection part is integrated in time with an implicit generalized one step finite difference. Because WAVEx uses the FEniCSx framework, it is automatically parallelized via MPI and the linear algebra backend that is used is the Portable, Extensible Toolkit for Scientific Computation (PETSc) (Balay et al., 2019). The linear solver that is used in all the following cases is the default GMRES solver with block Jacobi preconditioning. Also, WAVEx is designed to allow for the addition of different discretizations in both time and space besides just the specific choices made in this case.

### 3. Convergence test case results

To investigate the asymptotic convergence properties of the stabilized methods introduced in Section 2.3, two test cases will be considered. These test cases will be run with reduced dimension (2 instead of 4 in space) and the velocity fields will be simplified. For each case the error in the finite element solution,  $N_h$ , relative to the exact solution,  $N_{exact}$ , will be computed both in the  $L^2(\Omega)$  and the  $l^\infty(\Omega)$

norm that are defined as:

$$\|N_h - N_{exact}\|_{L^2} = \sqrt{\int_{\Omega} (N_h - N_{exact})^2 dx}, \quad (34)$$

$$\|N_h - N_{exact}\|_{l^\infty} = \max_{i \in n_{dof}} (|N_h(x_i) - N_{exact}(x_i)|).$$

#### 3.1. Propagation of a sinusoidal wave

The first test case is a harmonic wave propagating through a square domain over time. The goal of this test case is to evaluate the capabilities of the different FEM schemes in a “pure advection” scenario by setting source,  $S$ , to 0. Harmonic waves are propagating across 2D domain in the direction of the velocity vector,  $\mathbf{c} = (1, 1)$ , which is constant in space and time. The domain is a square,  $\Omega = (0, 10) \times (0, 10)$ . The exact solution is:

$$N_{analytic} = \sin(x - c_x t) + \cos(y - c_y t). \quad (35)$$

This dictates the initial and boundary conditions:

$$N_0(x, y) = \sin(x) + \cos(y), \quad (36)$$

$$N(x, y, t) = N_{analytic} \quad \text{for } (x, y) \in \Gamma_-,$$

and as a consequence of the velocity field the Dirichlet boundary,  $\partial\Omega_d$ , is the bottom and left sides of the square domain. The test case has a total simulation time of 5 units, which is split up into 1000 time steps. The time step is set small to minimize error from the approximation in time and focus on FEM error convergence with respect to  $h$  refinement.

Each scheme introduced in Section 2.3 is discretized with uniform triangular elements in space and order 1 Lagrange finite element functions for both trial and test space. Each scheme is fully implicit in time with order 1 implicit Euler finite difference approximation. The computational grid is refined in space with a uniform spacings of  $L/4$ ,  $L/8$ ,  $L/12$ ,  $L/16$ ,  $L/20$ ,  $L/24$ . For each finite element scheme, the global  $L^2$  and  $l^\infty$  errors are computed. The errors at the final time  $T = 5$  are plotted for both norms and for all schemes in Fig. 1.

For the smooth solution present in case 1, nearly optimal convergence rates of  $O(h^{k+1})$  were shown for the SUPG cases. The CG methods without stabilization converged at a rate of  $h^2$  until machine precision was met at  $L/20$ , which brings down the average shown in Table 2 to 1.559. The errors in the SUPG converged at a better rate than its theoretical rate of  $O(h^{1.5})$  most likely due to the minimal amount of residual present in the smooth solution. Least squares had the highest in magnitude of  $L^2$  error and converged at a rate between  $h^{1.5}$  and  $h^2$ . Lastly, we observe that the DG method had the lowest magnitude in errors for stabilized schemes and converged near the optimal theoretical rate of  $h^{1.5}$ .

**Table 2**  
Error convergence rates for case 1.

Method	Avg. L2 rate	Avg. Linf rate
CG strong	1.559	1.445
CG	1.559	1.445
SUPG strong	1.988	2.01
SUPG	1.988	2.01
Least squares	1.790	1.730
DG	1.611	1.981

**Table 3**  
Error convergence rates for case 2.

Method	Avg. L2 rate	Avg. Linf rate
CG strong	1.287	1.270
CG	1.287	1.270
SUPG strong	1.948	1.461
SUPG	1.795	1.461
Least squares	1.907	1.440
DG	1.999	1.460

### 3.2. Formation of a shock near the boundary

This test case is an adaptation of a common numerical test for advection dominated PDEs and is adapted from the work of [Egger and Schöberl \(2010\)](#). This case has an analytic solution that begins as a smooth bump then shifts into a shock towards the right boundary. The aim is to see how the stabilized schemes handle ill-conditioned problems and formation of discontinuities. Stability in this situation is critical for an operational model because a shock in the solution may form from sharp changes in bathymetry, currents, or wind forcing.

The domain is the unit square  $\Omega = (0, 1) \times (0, 1)$  with a uniform velocity field  $\mathbf{c} = \{1, 0\}$ . The analytic solution is:

$$N_{exact} = (-4(y - 0.5)^2 + 1) \left[ x + \frac{e^{\gamma c_x x} - 1}{1 - e^{\gamma c_x}} \right], \quad (37)$$

where  $\gamma = 100t$ ,  $c_x = 1$ , and  $t \in [0.01, 1]$ . The number of time steps for each run is set to 1000. A consequence of this analytic solution is that  $S$  should be:

$$(-4(y - 0.5)^2 + 1) \left[ \frac{100e^{100t}(e^{100tx} - 1)}{(1 - e^{100t})^2} + \frac{100xe^{100tx}}{1 - e^{100t}} \right] + (-4(y - 0.5)^2 + 1) \times \left[ 1 + \frac{100te^{\gamma c_x x}}{1 - e^{\gamma c_x}} \right]. \quad (38)$$

The domains are the same structure as in case 1, and the levels of h refinement for this case are  $L/4$ ,  $L/8$ ,  $L/12$ ,  $L/16$ ,  $L/20$ ,  $L/24$ ,  $L/28$ ,  $L/32$ ,  $L/36$ . The errors at final time  $T = 1$  are plotted for both  $L^2$  and  $l^\infty$  norms are displayed in [Fig. 2](#). Even with a small time step that is guaranteed stable, the solutions from the CG weak forms both produced the infamous oscillations in the final solution. A comparison in the solution at  $T = 1$  of CG and SUPG implementations are shown in [Fig. 3](#). Consequently, the resulting error convergence rates in the CG implementations were not optimal as in the former case. Interestingly, the SUPG and SUPG strong results did have differences as presented in [Fig. 2](#) with the SUPG strong version performing best. All the stabilized schemes did not produce visible oscillations at any level of refinement and all schemes converged at equal to or higher than the best theoretical rates as presented in [Table 3](#). The DG method had the lowest error of all the stabilized schemes, which is not surprising because for the same polynomial order the DG method has nearly double the degrees of freedom as the continuous implementations. We now move onto the test cases for the full WAE problem. For the full implementation, SUPG was chosen because it performed well in our preliminary 2D studies, it is most simple to implement and computationally the cheapest. In the future we would like to also implement the Least Squares and DG schemes.

### 4. Full scale test cases

In this section, 6 test cases from the ONR testbed will be implemented. For each of the cases the RMSE and  $l^\infty$  errors will be computed with respect to the WAVEx output and analytic solution or lab data depending on the case. The RMSE is computed using the definition:

$$RMSE(x, y) = \sqrt{\frac{\sum_{i=1}^{N_d} (x_i - y_i)^2}{N_d}}, \quad (39)$$

where  $x$  represents the set of parameters generated by the wave model,  $y$  is the set of parameters that come from either the analytic solution or observation, and  $N_d$  is the number of points the ONR test bed has data for. Using the same nomenclature as in [\(39\)](#), the  $l^\infty$  measure is defined as:

$$l^\infty(x, y) = \max_{1 \leq i \leq N_d} |x_i - y_i|. \quad (40)$$

In all cases, the data that is compared to the WAVEx model is either significant wave height ( $H_s$ ), or both  $H_s$  and mean wave direction ( $\theta_{mean}$ ). Significant wave height is defined as the highest 1/3 of recorded waves at a point, and mean wave direction is the mean direction of the spectrum. In WAVEx, the aforementioned statistical values must be estimated. The significant wave height is estimated by first calculating the zeroth moment of the spectrum,  $m_0$ . In general the  $i$ th moment of the spectrum is defined as:

$$m_i(x, y) = \int_{-\pi}^{\pi} \int_{\sigma_{min}}^{\sigma_{max}} \sigma^i E(x, y, \sigma, \theta) d\sigma d\theta, \quad (41)$$

and recall  $E = N\sigma$ . The significant wave height is subsequently estimated by:

$$H_s(x, y) \approx 4\sqrt{m_0}. \quad (42)$$

The reasoning behind this being a decent approximation to the highest one third of waves is not trivial but its assumptions and derivation can be found in several books such as the ([Holthuijsen, 2007](#); [LeBlond and Mysak, 1981](#)). The mean wave direction in degrees is computed as:

$$\theta_{mean}(x, y) = \frac{180}{\pi} \frac{\int_{-\pi}^{\pi} \int_{\sigma_{min}}^{\sigma_{max}} \cos(\theta) E(x, y, \sigma, \theta) d\sigma d\theta}{\int_{-\pi}^{\pi} \int_{\sigma_{min}}^{\sigma_{max}} \sin(\theta) E(x, y, \sigma, \theta) d\sigma d\theta}. \quad (43)$$

In all the presented test cases, WAVEx is run using the SUPG strong scheme as in [\(20\)](#) and the implicit time stepping uses the Crank–Nicolson scheme, which is when  $\alpha_t = 0.5$  from [\(22\)](#) for second order accuracy in time.

#### 4.1. A21: Shoaling

The first test case is an idealized situation where some waves are propagating from the deep ocean directly into the coastline. The water depth in the domain is uniformly decreasing from 20 m to 0 m with a slope of 0.005. The waves are propagating perpendicular to the coastline and are approximately monochromatic. The waves at the inflow boundary have a significant wave height of 1 m with a mean period of 10 s. All source terms are zero and there are no currents in this case. Because all source terms are zero, the analytic solution for the significant wave height can easily be computed because the WAE becomes a pure advection equation with nonuniform velocity field. The result is that the significant wave height should be:

$$H_s(x) = H_s(0) \sqrt{\frac{c_g(0)}{c_g(x)}}. \quad (44)$$

An unstructured mesh is used in the WAVEx simulations for cases A11 and A21 illustrated in [Fig. 4](#). The mesh has 7458 nodes comprising 13 841 triangular elements and has variable resolution with element size ranging from 800 m near the bottom to 20 m near the top. In WAVEx, the spectral grid is capable of being unstructured but for all

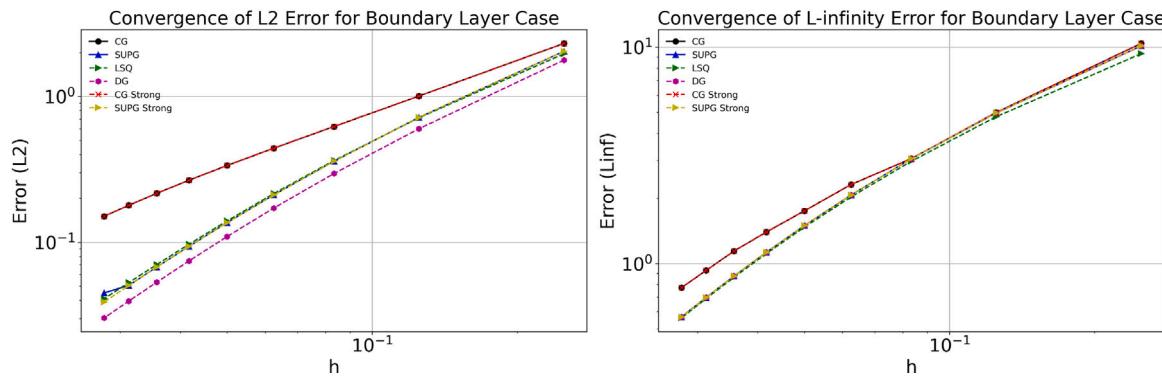


Fig. 2.  $L^2$  and  $l^\infty$  errors at  $T = 1$  for all schemes for case 2.

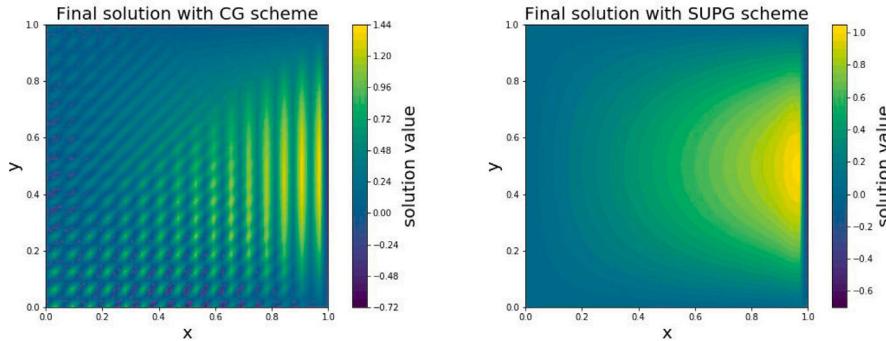


Fig. 3. Solution for case 2 on a  $32 \times 32$  node grid, SUPG right CG left.

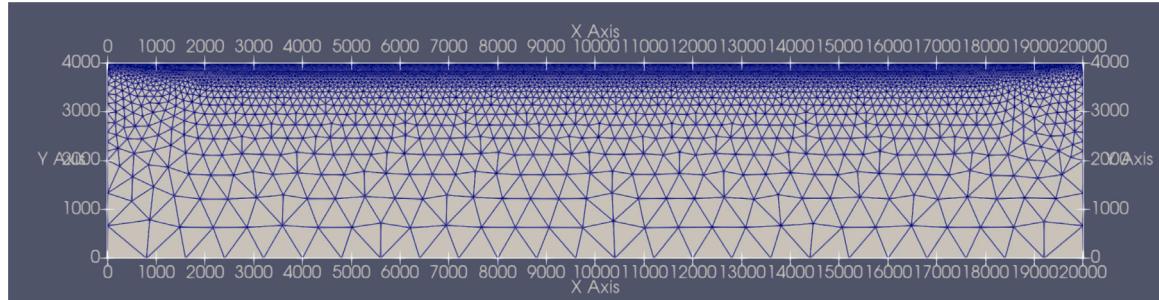


Fig. 4. The unstructured geographic mesh used for test cases A11, A21.

of the ONR cases it is structured with logarithmic spacing in frequency and uniform in direction. An example is shown in Fig. 5. The time step is set to 50 s for all the cases, and WAVEx solves until the iterative solver reaches a steady state. The significant wave height is computed along the propagation direction and results are plotted in Fig. 6. The results in this figure also clearly demonstrate the effect of instability in the Bubnov–Galerkin solution as its oscillations makes the solution very poor. The computed RMSE error vs the analytic solution is 0.000695 m for WAVEx with SUPG stabilization and 0.313 m without, while  $l^\infty$  error is 0.0018 m with SUPG and .9916 m without.

#### 4.2. A11: Refraction

The second test case is similar to case A21 in that the bathymetry is the same and the domain is identical. The only difference is that the waves on the inflow boundary now approach the shoreline at a direction 30 degrees above the direction perpendicular to the shoreline. This too has an analytic solution. Again, no source terms are present

and the currents are all 0. In addition to significant wave height, the mean wave direction is measured and the analytic solutions are:

$$\begin{aligned} \theta(x) &= \arcsin(c(x) \frac{\sin(30)}{c(0)}), \\ H_s(x) &= H_s(0) \sqrt{\frac{c_g(0)\cos(30)}{c_g(x)\cos(\theta(x))}}. \end{aligned} \quad (45)$$

The significant wave height is plotted against the analytic solution in Fig. 7 while the mean wave direction is plotted in Fig. 8. In both figures, the WAVEx and analytic results agree very well. The observed RMSE error in  $H_s$  is 0.00261 m and RMSE error in degrees is 0.119 degrees while the  $l^\infty$  error is 0.00865 m and 0.1946 degrees.

#### 4.3. A31–34: Currents

Cases A31 through A34 are situations in deep water where there are interacting currents but no source terms. All simulations are performed

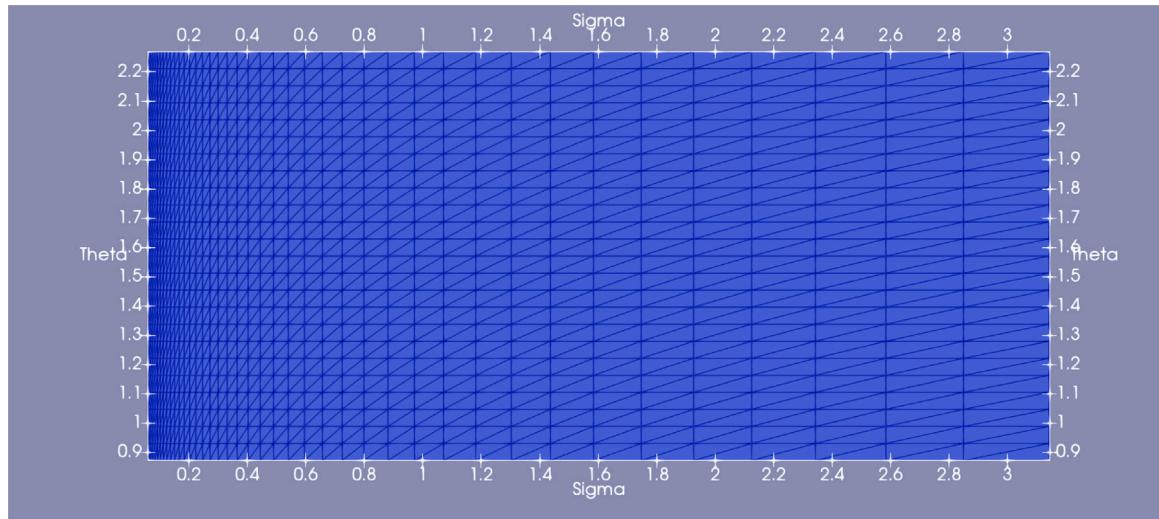


Fig. 5. Logarithmic spaced spectral grid as used in cases A11, A21.

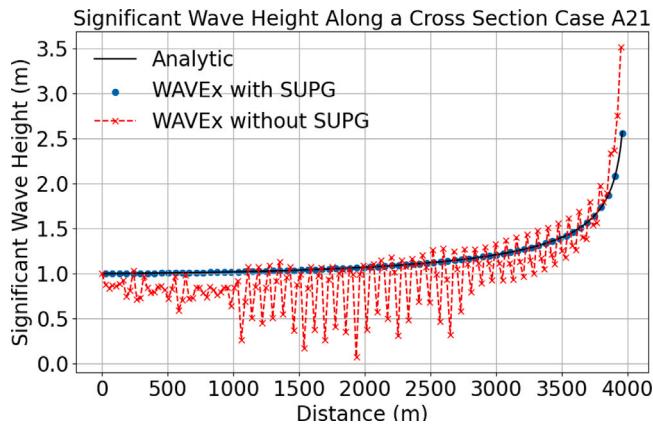


Fig. 6. Significant wave heights from case A21.

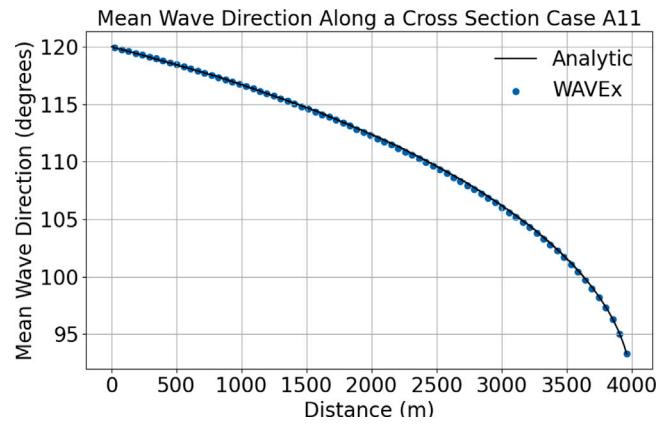


Fig. 8. Mean wave direction for case A11.

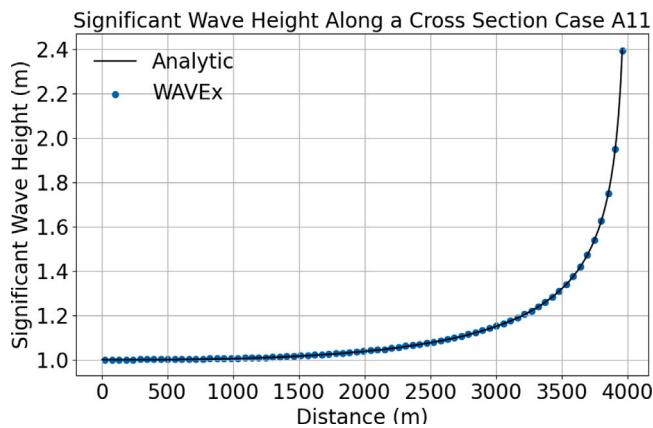


Fig. 7. Significant wave height for case A11.

on the same domain, and is a rectangle shape of constant arbitrarily deep water (set to 10 000 m) and is 4000 m long in the principal direction. To avoid any noise from the side boundaries, the width of the domain is set to 10 000 m. Each case has either different currents or different mean wave directions on the inflow boundary. Case A31 has an inflow boundary where the mean wave direction is parallel with the principle axis with a current that is also parallel with the principle axis and increases from 0 m/s at the inflow boundary to 2 m/s at the

outflow boundary. Case A32 has the same boundary conditions as A31 but the currents in this case decrease from 0 m/s at the inflow boundary to -2 m/s at the outflow boundary. Case A33 has the same current profile as case A31 but the inflow boundary condition now has a mean wave direction of 120 degrees relative to the positive direction of the principle axis. Finally, case A34 is the same as A33 but instead of 120 degrees, it is set to 60 degrees. All waves at the inflow boundary have a significant wave height of 1 m and peak period of 10 s. Similar to cases A11 and A21, an analytic expression for the directions and significant wave heights may be derived. These are more complex due to the nonzero currents and are different for cases A31, A32 and the A33, A34 pair. The solutions are different because the former have incoming spectra parallel to the currents while the latter are at an angle. The analytic solution for significant wave height in cases A31 and A32 is:

$$H_s(x) = H_s(0) \sqrt{\frac{c(0)^2}{c(x)(c(x) + 2u(x))}}, \quad (46)$$

$$c(x) = c(0)(0.5 + 0.5 \sqrt{1 + 4 \frac{u(x)}{c(0)}}),$$

while the solution for cases A33 and A34 along with the analytic solution for mean wave direction is:

$$H_s(x) = H_s(0) \sqrt{\frac{\sin(2\theta(0))}{\sin(2\theta(x))}}, \quad (47)$$

$$\theta(x) = \arccos\left(\frac{gk(0)\cos(\theta(0))}{\left(\frac{2\pi}{P_{kper}} - u(x)k(0)\cos(\theta(0))\right)^2}\right).$$

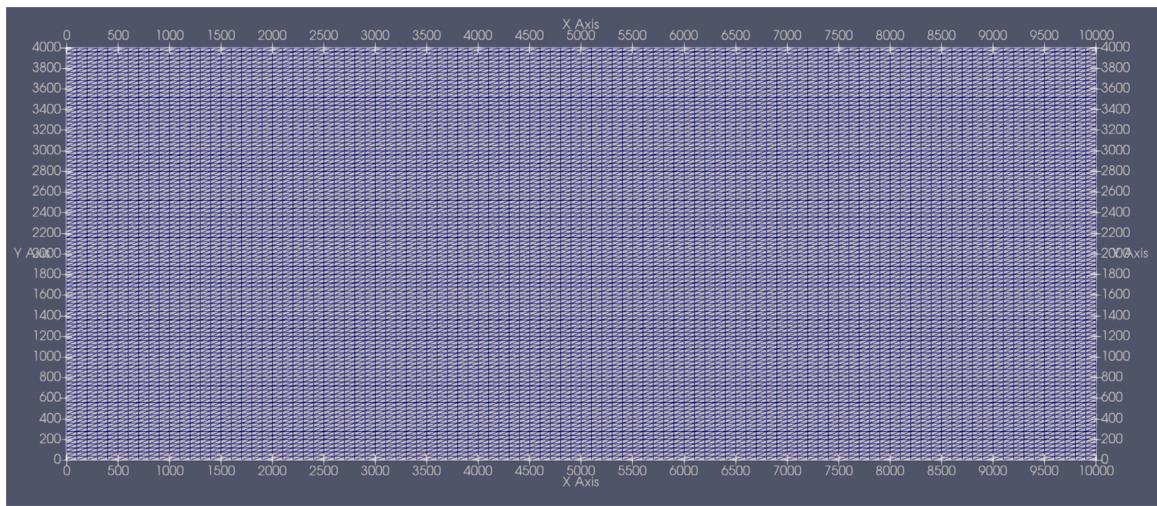


Fig. 9. Geographic mesh for cases A31–A34.

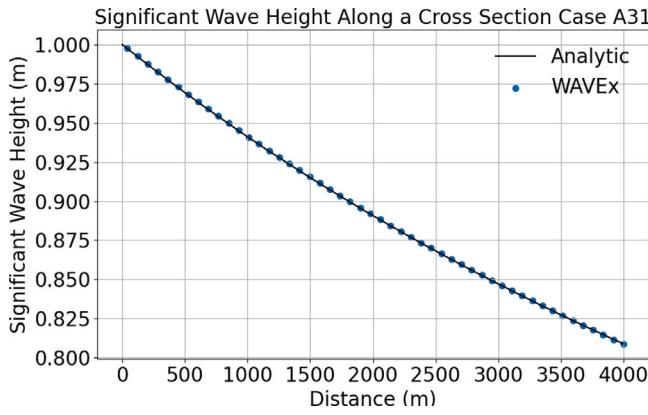


Fig. 10. Significant wave height from case A31.

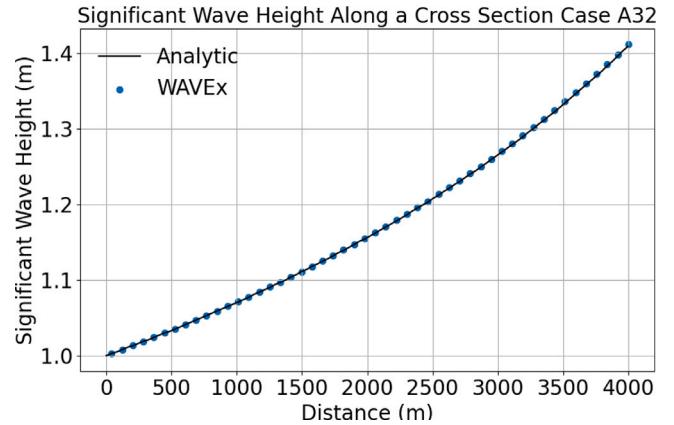


Fig. 11. Significant wave height from case A32.

For these cases, the mesh is uniformly spaced with triangular elements. The mesh contains 10,201 nodes that comprise 20,000 elements. This mesh is 10 000 m in the  $x$  direction and 4000 m in the  $y$  direction. Each element has a size of around 110 m. The mesh is 100 elements wide and 100 elements tall. The mesh is illustrated in Fig. 9. The resulting significant wave heights from case A31 and A32 are shown in Figs. 10, 11, and the computed RMSE are 0.000255 m, and 0.00109 m respectively while  $l^\infty$  error is 0.0003737 m and 0.00257 m. Significant wave height for cases A33 and A34 are plotted in Figs. 12 and 14. The RMSE for significant wave height for A33 is 0.000315 m, while A34 is 0.000899 m. The  $l^\infty$  for case A33 is 0.000659 m while A34 is 0.00128 m. The resulting mean wave direction for cases A33 and A34 are shown in Figs. 13 and 15. The RMSE for mean wave direction in case A33 is 0.1793 degrees while for case A34 it is 0.436. The  $l^\infty$  for case A33 is 0.028 degrees while case A34 is 0.0436 degrees. We observe close agreement of WAVEx and the analytic solutions in all cases. This agreement is further witnessed in the low errors in both mean wave direction and significant wave height.

#### 4.4. L11: Wave breaking on a beach

The final test case is the L11 case, a laboratory experiment intended to simulate groups of waves breaking on a beach. The bathymetry varies in slope as it approaches the beach but is uniform in the along-shore direction. The input spectrum is a JONSWAP spectrum with peakedness parameters specified in the manual (Ris et al., 2003). The

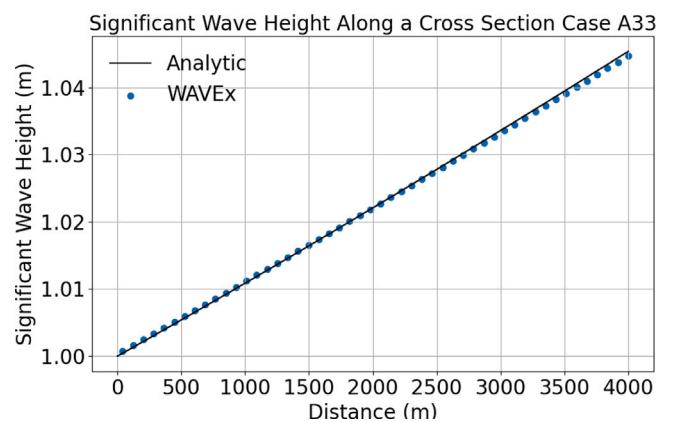


Fig. 12. Significant wave height from case A33.

mesh that is used is in this case and is set as the rectangle  $[0, 113] \times [7.4, 30]$ , it is a structured, uniform mesh with triangular elements. Each element is approximately 1 m wide and there are 4681 total nodes comprising 9000 elements. The mesh is shown in Fig. 16. In addition to comparison to the measured results from the ONR testbed (Ris et al., 2003), we also compare the WAVEx results to results from SWAN. The

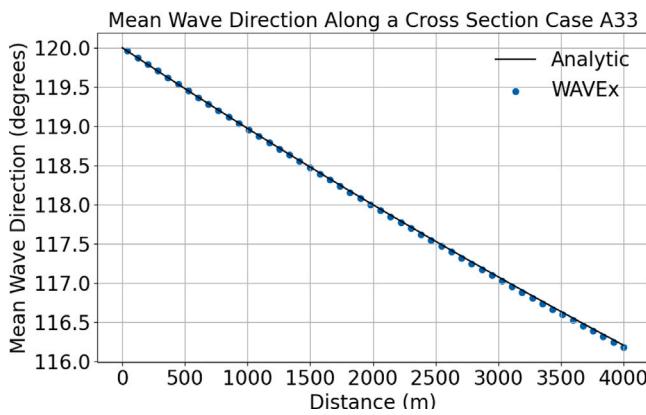


Fig. 13. Mean wave direction from case A33.

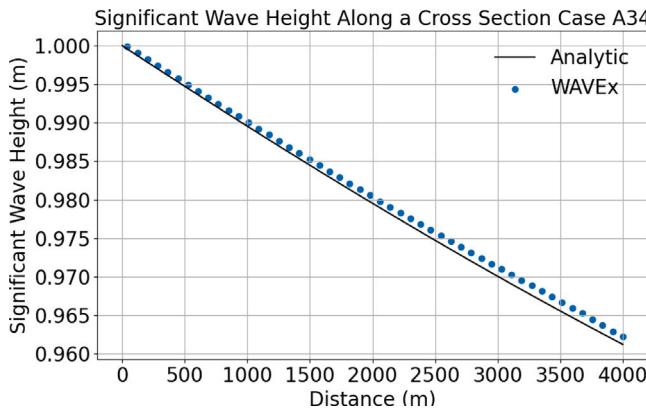


Fig. 14. Significant wave height from case A34.

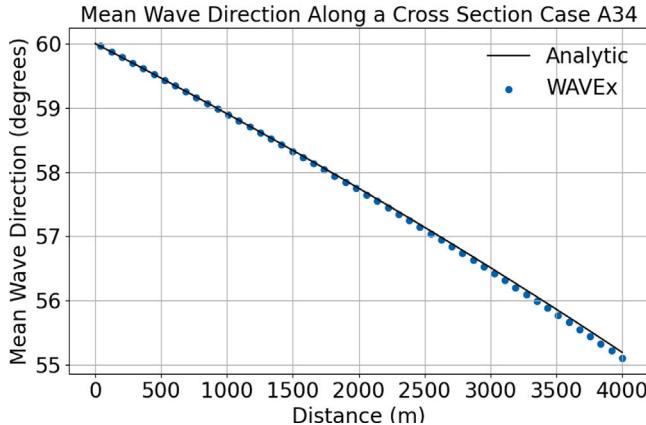


Fig. 15. Mean wave direction from case A34.

SWAN numerical model is set up with the third generation source terms are turned on, however there is no wind in this case. The significant wave heights are recorded both from SWAN and WAVEx and plotted against the observations in Fig. 17. The RMSE in significant wave height with respect to the observations were 0.0064 m for WAVEx and 0.00556 m for SWAN while the  $L^\infty$  errors were 0.0179 m and 0.015 m respectively. We note that SWAN was run in 1-D mode while WAVEx was used in full 2-D mode, we suspect this could have a slight impact in the results.

## 5. Conclusions

The purpose of this work was to develop a finite element spectral wind wave model using the open source FEniCSx library, which we call WAVEx. The motivation behind this was to create a platform so that different numerical schemes and source terms can easily be implemented and tested in the future. First, some stabilized FEM methods were introduced (least squares, SUPG, DG) and convergence rates with respect to mesh refinement were verified in simple 2D test cases. All of the stabilized schemes demonstrated convergence in the  $L^2$  and  $L^\infty$  norms with respect to  $h$  refinement while also reducing oscillations when compared to the standard Bubnov–Galerkin CG approach that produced oscillations when the solution became less smooth.

Next, a spectral wave model with SUPG stabilization in 4 dimensions was implemented. The WAE was split into propagation and source terms via Strang splitting for computational efficiency. The propagation portion was implemented with an implicit time stepping scheme while the (nonlinear) source term portion was implemented with an explicit time stepping scheme in order to keep the discrete problem linear. The new wave model showed accuracy and stability in the analytic test cases and a laboratory test case that included source terms.

In the future, a first step to improving WAVEx would be to introduce a custom FEniCSx assembler in order to speed up run times. One drawback of the way WAVEx is currently implemented is that the assembly routine is relatively slow, but this can be circumvented by developing new tools in the FEniCSx. Other future investigations include how WAVEx scales with more complex test cases in addition to fully implicit time stepping by extending WAVEx to solve nonlinear problems. Additionally, implementing the DG and least squares schemes in the full (4D) operational setting is a natural extension that we have not done at the present time. While we have considered several FEMs, there is a plethora of methods of interest that show promise in handling problems like the WAE such as Lagrange–Galerkin or discontinuous Petrov–Galerkin (DPG) (Demkowicz and Gopalakrishnan, 2010) methods.

## CRediT authorship contribution statement

**Mark Loveland:** Methodology, Software, Validation, Writing – original draft. **Eirik Valseth:** Methodology, Writing – review & editing. **Jessica Meixner:** Methodology, Writing – review & editing. **Clint Dawson:** Resources, Supervision, Project administration.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Data availability

Data will be made available on request.

## Acknowledgments

Author Loveland has been supported by the CSEM Fellowship from the Oden Institute at the University of Texas at Austin, USA. Authors Loveland, Valseth, and Dawson have been supported by the United States National Science Foundation - NSF PREVENTS Track 2 Program, under NSF Grant Number 1855047 and the Department of Homeland Security Coastal Resilience Center, USA research project “Accurate and Fast Wave Modeling and Coupling with ADCIRC”. Author Meixner has been supported by the National Oceanic and Atmospheric Administration, USA. The authors would also like to thank the reviewers of this manuscript for their time and thoughtful suggestions.

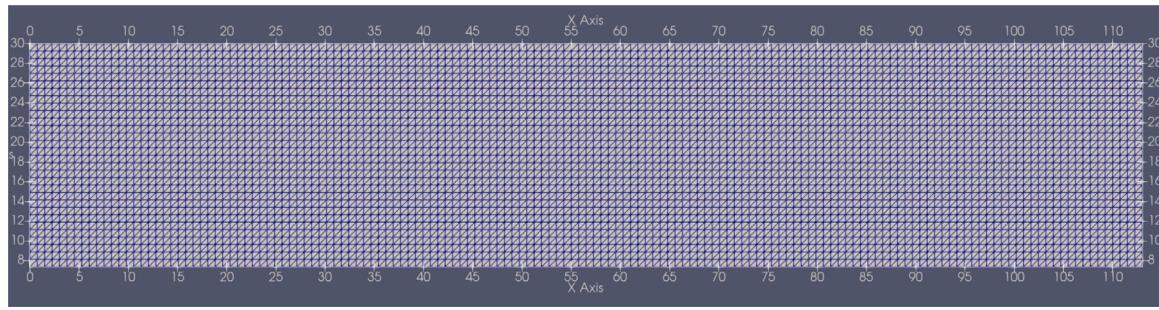


Fig. 16. Geographic mesh for case L11.

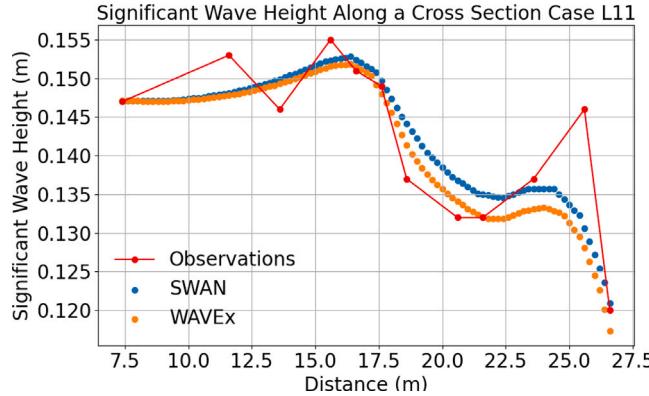


Fig. 17. Significant wave height for case L11.

## Appendix

This appendix illustrates how to rewrite both (20) and (21) into a product of 2-dimensional integrals so that FEniCSx can interpret it through the unified form language (UFL) framework. We evaluate the third term in (21):

$$(\mathbf{c}\phi_i\psi_j \cdot \mathbf{n}, \gamma_k\beta_l)_{\partial\Omega} = (\mathbf{c}\phi_i\psi_j \cdot \mathbf{n}, \gamma_k\beta_l)_{\partial\Omega_1 \times \partial\Omega_2 \times \partial\Omega_1 \times \partial\Omega_2} = \int_{\partial\Omega_2} \psi_j\beta_l \int_{\partial\Omega_1} \phi_i\gamma_k \mathbf{c}_1 \cdot \mathbf{n}_1 dx dy + \int_{\partial\Omega_1} \psi_j\beta_l \mathbf{n}_2 \cdot \int_{\partial\Omega_2} \mathbf{c}_2 \phi_i \beta_k dx dy, \quad (48)$$

and for the continuous Galerkin components of the form in (20) we have:

$$(\frac{\partial N}{\partial t} + \nabla \cdot (\mathbf{c}N) - S, v)_{\Omega}. \quad (49)$$

Replacing the functions with the finite element basis and omitting the transient and source term we get:

$$(\nabla \cdot (\mathbf{c}\phi_i\psi_j), \gamma_k\beta_l)_{\Omega} = (\mathbf{c} \cdot \nabla(\phi_i\psi_j) + (\phi_i\psi_j) \nabla \cdot \mathbf{c}, \gamma_k\beta_l)_{\Omega}, \quad (50)$$

$$(\mathbf{c}_1 \cdot \psi_j \nabla_1 \phi_i + \mathbf{c}_2 \cdot \phi_i \nabla_2 \psi_j + \phi_i \psi_j \nabla_1 \cdot \mathbf{c}_1 + \phi_i \psi_j \nabla_2 \cdot \mathbf{c}_2, \gamma_k\beta_l)_{\Omega},$$

which decomposes into the following terms:

$$(\mathbf{c}_1 \cdot \psi_j \nabla_1 \phi_i, \gamma_k\beta_l)_{\Omega} = \int_{\Omega_2} \psi_j\beta_l \int_{\Omega_1} \mathbf{c}_1 \cdot \nabla_1(\phi_i)\gamma_k dx dy, \quad (51)$$

$$(\mathbf{c}_2 \cdot \phi_i \nabla_2 \psi_j, \gamma_k\beta_l)_{\Omega} = \int_{\Omega_2} \beta_l \nabla_2 \psi_j \cdot \int_{\Omega_1} \mathbf{c}_2 \phi_i \gamma_k dx dy,$$

$$(\phi_i \psi_j \nabla_1 \cdot \mathbf{c}_1, \gamma_k\beta_l)_{\Omega} = \int_{\Omega_2} \psi_j\beta_l \int_{\Omega_1} \nabla_1 \cdot (\mathbf{c}_1) \phi_i \gamma_k dx dy,$$

$$(\phi_i \psi_j \nabla_2 \cdot \mathbf{c}_2, \gamma_k\beta_l)_{\Omega} = \int_{\Omega_1} \phi_i \gamma_k \int_{\Omega_2} \nabla_2 \cdot (\mathbf{c}_2) \psi_j \beta_l dy dx.$$

Now for SUPG stabilization terms in both forms we need to add the upwind component:

$$(\frac{\partial N}{\partial t} + \nabla \cdot (\mathbf{c}N) - S, \tau \mathbf{c} \cdot \nabla(v))_{\Omega}. \quad (52)$$

The source term will be absorbed into the RHS and we will first focus our attention on the streamline diffusion term:

$$(\nabla \cdot (\mathbf{c}N), \tau \mathbf{c} \cdot \nabla v)_{\Omega}. \quad (53)$$

Using our product basis expansion, we consider this term now in the discrete setting and expand using the product rule:

$$\begin{aligned} (\nabla \cdot (\mathbf{c}N), \tau \mathbf{c} \cdot \nabla v)_{\Omega} &\rightarrow (\nabla \cdot (\mathbf{c}\phi_i\psi_j), \tau \mathbf{c} \cdot \nabla(\gamma_k\beta_l))_{\Omega} =, \\ &(\nabla \cdot (\mathbf{c}\phi_i\psi_j), \tau \beta_l \mathbf{c}_1 \cdot \nabla_1 \gamma_k + \tau \gamma_k \mathbf{c}_2 \cdot \nabla_2 \beta_l)_{\Omega} =, \\ &(\mathbf{c} \cdot \nabla(\phi_i\psi_j) + (\phi_i\psi_j) \nabla \cdot \mathbf{c}, \tau \beta_l \mathbf{c}_1 \cdot \nabla_1 \gamma_k + \tau \gamma_k \mathbf{c}_2 \cdot \nabla_2 \beta_l)_{\Omega} =, \\ &(\mathbf{c}_1 \cdot \psi_j \nabla_1(\phi_i) + \mathbf{c}_2 \cdot \phi_i \nabla_2(\psi_j) + (\phi_i\psi_j) \nabla_1 \cdot \mathbf{c}_1 + (\phi_i\psi_j) \nabla_2 \cdot \mathbf{c}_2, \tau \beta_l \mathbf{c}_1 \cdot \nabla_1 \gamma_k \\ &\quad + \tau \gamma_k \mathbf{c}_2 \cdot \nabla_2 \beta_l)_{\Omega}. \end{aligned} \quad (54)$$

We have 4 scalar terms on the left hand part of the inner product and 2 on the right, if we expand all of these out it will lead to a sum of 8 total terms:

$$\begin{aligned} &(\mathbf{c}_1 \cdot \psi_j \nabla_1(\phi_i), \tau \mathbf{c}_1 \cdot \beta_l \nabla_1(\gamma_k))_{\Omega}, \\ &(\mathbf{c}_1 \cdot \psi_j \nabla_1(\phi_i), \tau \mathbf{c}_2 \cdot \gamma_k \nabla_2(\beta_l))_{\Omega}, \\ &(\mathbf{c}_2 \cdot \phi_i \nabla_2(\psi_j), \tau \mathbf{c}_1 \cdot \beta_l \nabla_1(\gamma_k))_{\Omega}, \\ &(\mathbf{c}_2 \cdot \phi_i \nabla_2(\psi_j), \tau \mathbf{c}_2 \cdot \gamma_k \nabla_2(\beta_l))_{\Omega}, \\ &((\phi_i\psi_j) \nabla_1 \cdot \mathbf{c}_1, \tau \mathbf{c}_1 \cdot \beta_l \nabla_1(\gamma_k))_{\Omega}, \\ &((\phi_i\psi_j) \nabla_1 \cdot \mathbf{c}_1, \tau \mathbf{c}_2 \cdot \gamma_k \nabla_2(\beta_l))_{\Omega}, \\ &((\phi_i\psi_j) \nabla_2 \cdot \mathbf{c}_2, \tau \mathbf{c}_1 \cdot \beta_l \nabla_1(\gamma_k))_{\Omega}, \\ &((\phi_i\psi_j) \nabla_2 \cdot \mathbf{c}_2, \tau \mathbf{c}_2 \cdot \gamma_k \nabla_2(\beta_l))_{\Omega}. \end{aligned} \quad (55)$$

Now looking term by term we have first:

$$(\mathbf{c}_1 \psi_j \cdot \nabla_1 \phi_i, \tau \beta_l \mathbf{c}_1 \cdot \nabla_1 \gamma_k)_{\Omega} = \int_{\Omega_2} \psi_j \beta_l \int_{\Omega_1} \tau \mathbf{c}_1 \cdot \nabla_1 \phi_i \mathbf{c}_1 \cdot \nabla_1 \gamma_k dx dy. \quad (56)$$

The second term we get:

$$(\mathbf{c}_1 \psi_j \cdot \nabla_1 \phi_i, \tau \gamma_k \mathbf{c}_2 \cdot \nabla_2 \beta_l)_{\Omega} = \int_{\Omega_2} \psi_j \nabla_2 \beta_l \int_{\Omega_1} \mathbf{c}_2 \tau \gamma_k \mathbf{c}_1 \cdot \nabla_1 \phi_i dx dy. \quad (57)$$

The third term will be:

$$(\mathbf{c}_2 \phi_i \cdot \nabla_2 \psi_j, \tau \beta_l \mathbf{c}_1 \cdot \nabla_1 \gamma_k)_{\Omega} = \int_{\Omega_2} \nabla_2 \psi_j \beta_l \int_{\Omega_1} \tau \mathbf{c}_2 \phi_i \mathbf{c}_1 \cdot \nabla_1 \gamma_k dx dy. \quad (58)$$

Fourth term:

$$(\mathbf{c}_2 \phi_i \cdot \nabla_2 \psi_j, \tau \gamma_k \mathbf{c}_2 \cdot \nabla_2 \beta_l)_{\Omega} = \int_{\Omega_1} \phi_i \gamma_k \int_{\Omega_2} \tau \mathbf{c}_2 \cdot \nabla_2 \psi_j \mathbf{c}_2 \cdot \nabla_2 \beta_l dy dx. \quad (59)$$

Fifth term:

$$((\phi_i\psi_j) \nabla_1 \cdot \mathbf{c}_1, \tau \mathbf{c}_1 \cdot \beta_l \nabla_1(\gamma_k))_{\Omega} = \int_{\Omega_2} \psi_j \beta_l \int_{\Omega_1} \tau \phi_i \nabla_1 \cdot \mathbf{c}_1 \mathbf{c}_1 \cdot \nabla_1(\gamma_k) dx dy. \quad (60)$$

Sixth:

$$((\phi_i\psi_j) \nabla_1 \cdot \mathbf{c}_1, \tau \mathbf{c}_2 \cdot \gamma_k \nabla_2(\beta_l))_{\Omega} = \int_{\Omega_2} \psi_j \nabla_2(\beta_l) \int_{\Omega_1} \mathbf{c}_2 \tau \phi_i \gamma_k \nabla_1 \cdot \mathbf{c}_1 dx dy. \quad (61)$$

Seventh:

$$((\phi_i\psi_j) \nabla_2 \cdot \mathbf{c}_2, \tau \mathbf{c}_1 \cdot \beta_l \nabla_1(\gamma_k))_{\Omega} = \int_{\Omega_1} \phi_i \nabla_1(\gamma_k) \int_{\Omega_2} \mathbf{c}_1 \tau \psi_j \beta_l \nabla_2 \cdot \mathbf{c}_2 dy dx. \quad (62)$$

Eighth:

$$((\phi_i \psi_j) \nabla_2 \cdot \mathbf{c}_2, \tau \mathbf{c}_2 \cdot \gamma_k \nabla_2(\beta_l))_{\Omega} = \int_{\Omega_1} \phi_i \gamma_k \int_{\Omega_2} \tau \mathbf{c}_2 \cdot \nabla_2(\beta_l) \psi_j \nabla_2 \cdot \mathbf{c}_2 dy dx. \quad (63)$$

Now the last part that remains is the upwinding portion based on the choice of time step. Choosing a simple implicit time step we get:

$$(\frac{\partial N}{\partial t}, \tau \mathbf{c} \cdot \nabla v)_{\Omega} \approx (\frac{\phi_i^{n+1} \psi_j^{n+1} - \phi_i^n \psi_j^n}{\Delta t}, \tau \mathbf{c} \cdot \nabla(\gamma_k \beta_l))_{\Omega}. \quad (64)$$

The component with the previous time step  $n$  will go to RHS and the part that is in current time step  $n+1$  will be in stiffness matrix. Both discretizations however will be the same so I will just do the  $n+1$  time step after multiplying by  $\Delta t$ :

$$(\phi_i \psi_j, \tau \mathbf{c} \cdot \nabla(\gamma_k \beta_l))_{\Omega} = (\phi_i \psi_j, \tau(\mathbf{c}_x \cdot \beta_l \nabla_x(\gamma_k) + \mathbf{c}_y \cdot \gamma_k \nabla_y(\beta_l)))_{\Omega}. \quad (65)$$

Focusing on left term we can evaluate using this decomposition:

$$(\phi_i \psi_j, \tau \mathbf{c}_x \cdot \beta_l \nabla_x(\gamma_k))_{\Omega} = \int_{\Omega_2} \psi_j \beta_l \int_{\Omega_1} \tau \phi_i \mathbf{c}_x \cdot \nabla_x \gamma_k dx dy. \quad (66)$$

Then on the right we get:

$$(\phi_i \psi_j, \tau \mathbf{c}_y \cdot \gamma_k \nabla_y(\beta_l))_{\Omega} = \int_{\Omega_2} \psi_j \nabla_y \beta_l \cdot \int_{\Omega_1} \tau \mathbf{c}_y \phi_i \gamma_k dx dy. \quad (67)$$

## References

Abdolali, A., Roland, A., Van Der Westhuyzen, A., Meixner, J., Chawla, A., Hesser, T.J., Smith, J.M., Sikiric, M.D., 2020. Large-scale hurricane modeling using domain decomposition parallelization and implicit scheme implemented in WAVEWATCH III wave model. *Coast. Eng.* 157, 103656.

Abgrall, R., 2006. Residual distribution schemes: current status and future trends. *Comput. & Fluids* 35 (7), 641–669.

Alves, J.-H., Ardhuijn, F., Babanin, A., Banner, M., Belibassakis, K., Benoit, M., Donelan, M., Groeneweg, J., Herbers, T., Hwang, P., Janssen, P., Janssen, T., Lavrenov, I., Magne, R., Monbaliu, J., Onorato, M., Polnikov, V., Resio, D., Rogers, W., Cavalieri, L., 2007. Wave modelling - The state of the art. *Prog. Oceanogr.* 75, 603–674. <http://dx.doi.org/10.1016/j.pocean.2007.05.005>.

Babuska, I., Whiteman, J., Strouboulis, T., 2010. *Finite Elements: An Introduction to the Method and Error Estimation*. Oxford University Press.

Bayal, S., Abhyankar, S., Adams, M., Brown, J., Brune, P., Buschelman, K., Dalcin, L., Dener, A., Eijkhout, V., Gropp, W., et al., 2019. PETSc users manual.

Becker, E.B., Carey, G.F., Oden, J.T., 1981. *Finite Elements: An Introduction*, Vol. 1. Prentice Hall.

Benoit, M., Marcos, F., Becd, F., 1997. TOMAWAC. A prediction model for offshore and nearshore storm waves.

Bermejo, R., Carpio, J., Saavedra, L., 2023. New error estimates of Lagrange-Galerkin methods for the advection equation. <http://dx.doi.org/10.48550/ARXIV.2301.03438>, URL <https://arxiv.org/abs/2301.03438>.

Bochev, P.B., Gunzburger, M.D., 1998. Finite element methods of least-squares type. *SIAM Rev.* 40 (4), 789–837.

Bochev, P., Gunzburger, M., 2006. Least-Squares Finite Element Methods, Vol. 166. <http://dx.doi.org/10.1007/b13382>.

Booij, N., Ris, R.C., Holthuijsen, L.H., 1999. A third-generation wave model for coastal regions: 1. Model description and validation. *J. Geophys. Res.: Oceans* 104 (C4), 7649–7666. <http://dx.doi.org/10.1029/98JC02622>, arXiv: <https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/98JC02622> URL <https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/98JC02622>.

Brenner, S.C., Scott, L.R., 2008. *The Mathematical Theory of Finite Element Methods*, Vol. 3. Springer.

Brezzi, F., Cockburn, B., Marini, L.D., Süli, E., 2006. Stabilization mechanisms in discontinuous Galerkin finite element methods. *Comput. Methods Appl. Mech. Engrg.* 195 (25–28), 3293–3310.

Brezzi, F., Marini, L.D., Süli, E., 2004. Discontinuous Galerkin methods for first-order hyperbolic problems. *Math. Models Methods Appl. Sci.* 14 (12), 1893–1903.

Brooks, A.N., Hughes, T.J., 1982. Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Comput. Methods Appl. Mech. Engrg.* 32 (1), 199–259. [http://dx.doi.org/10.1016/0045-7825\(82\)90071-8](http://dx.doi.org/10.1016/0045-7825(82)90071-8), URL <https://www.sciencedirect.com/science/article/pii/0045782582900718>.

Burman, E., 2010. Consistent SUPG-method for transient transport problems: Stability and convergence. *Comput. Methods Appl. Mech. Engrg.* 199 (17–20), 1114–1123.

Burman, E., Smith, G., 2011. Analysis of the space semi-discretized SUPG method for transient convection-diffusion equations. *Math. Models Methods Appl. Sci.* 21 (10), 2049–2068.

Carey, G.F., Oden, J.T., 1983. *Finite Elements: A Second Course*. Prentice-hall.

Cockburn, B., Karniadakis, G.E., Shu, C.-W., 2000. *The Development of Discontinuous Galerkin Methods*. Springer.

Codina, R., 2000. On stabilized finite element methods for linear systems of convection-diffusion-reaction equations. *Comput. Methods Appl. Mech. Engrg.* 188 (1), 61–82. [http://dx.doi.org/10.1016/S0045-7825\(00\)00177-8](http://dx.doi.org/10.1016/S0045-7825(00)00177-8), URL <https://www.sciencedirect.com/science/article/pii/S0045782500001778>.

Dawson, C., Proft, J., 2001. A priori error estimates for interior penalty versions of the local discontinuous Galerkin method applied to transport equations. *Numer. Methods Partial Differential Equations: Int. J.* 17 (6), 545–564.

Dawson, C., Van Duijn, C., Wheeler, M., 1994. Characteristic-Galerkin methods for contaminant transport with nonequilibrium adsorption kinetics. *SIAM J. Numer. Anal.* 31 (4), 982–999.

Demkowicz, L., Gopalakrishnan, J., 2010. A class of discontinuous Petrov-Galerkin methods. Part I: The transport equation. *Comput. Methods Appl. Mech. Engrg.* 199 (23), 1558–1572. <http://dx.doi.org/10.1016/j.cma.2010.01.003>, URL <https://www.sciencedirect.com/science/article/pii/S0045782510000125>.

Donea, J., Quartapelle, L., 1992. An introduction to finite element methods for transient advection problems. *Comput. Methods Appl. Mech. Engrg.* 95 (2), 169–203. [http://dx.doi.org/10.1016/0045-7825\(92\)90139-B](http://dx.doi.org/10.1016/0045-7825(92)90139-B), URL <https://www.sciencedirect.com/science/article/pii/004578259290139B>.

Douglas, Jr., J., Russell, T.F., 1982. Numerical methods for convection-dominated diffusion problems based on combining the method of characteristics with finite element or finite difference procedures. *SIAM J. Numer. Anal.* 19 (5), 871–885. <http://dx.doi.org/10.1137/0719063>, arXiv:<https://doi.org/10.1137/0719063>.

Egger, H., Schöberl, J., 2010. A hybrid mixed discontinuous Galerkin finite-element method for convection-diffusion problems. *IMA J. Numer. Anal.* 30 (4), 1206–1234.

Ern, A., Guermond, J.-L., 2013. *Theory and Practice of Finite Elements*, Vol. 159. Springer Science & Business Media.

Ern, A., Vohralík, M., Zakerzadeh, M., 2021. Guaranteed and robust L2-norm a posteriori error estimates for 1D linear advection problems. *ESAIM Math. Model. Numer. Anal.* 55, S447–S474.

Glowinski, R., Osher, S., Yin, W., 2016. Splitting Methods in Communication, Imaging, Science and Engineering. <http://dx.doi.org/10.1007/978-3-319-41589-5>.

Gottlieb, S., Grant, Z.J., Isherwood, L., 2017. Strong stability preserving integrating factor runge-kutta methods. <http://dx.doi.org/10.48550/ARXIV.1708.02595> URL <https://arxiv.org/abs/1708.02595>.

Group, T.W., 1988. The WAM model—A third generation ocean wave prediction model. *J. Phys. Oceanogr.* 18 (12), 1775–1810.

Habera, M., Hale, J.S., Richardson, C., Ring, J., Rognes, M., Sime, N., Wells, G.N., 2020. FEniCSX: A sustainable future for the FEniCS project. <http://dx.doi.org/10.4084/m9.figshare.11866101.v1>, URL [https://figshare.com/articles/presentation/FEniCSX\\_A\\_sustainable\\_future\\_for\\_the\\_FEniCS\\_Project/11866101](https://figshare.com/articles/presentation/FEniCSX_A_sustainable_future_for_the_FEniCS_Project/11866101).

Holthuijsen, L.H., 2007. Waves in Oceanic and Coastal Waters. Cambridge University Press, <http://dx.doi.org/10.1017/CBO9780511618536>.

Hsu, T.-W., Ou, S.-H., Liu, J.-M., 2005. Hindcasting nearshore wind waves using a FEM code for SWAN. *Coast. Eng.* 52, 177–195. <http://dx.doi.org/10.1016/j.coastaleng.2004.11.005>.

Janssen, P., 2004. *The Interaction of Ocean Waves and Wind*. Cambridge University Press.

Janssen, P., 2008. Progress in ocean wave forecasting. *J. Comput. Phys.* 227, 3572–3594. <http://dx.doi.org/10.1016/j.jcp.2007.04.029>.

Johnson, C., 2012. *Numerical Solution of Partial Differential Equations by the Finite Element Method*. Courier Corporation.

Johnson, C., Navert, U., Pitkaranta, J., 1984. Finite element methods for linear hyperbolic problems. *Comput. Methods Appl. Mech. Engrg.* 45, 285–312.

Khandekar, M., 1989. Operational Analysis and Prediction of Ocean Wind Waves, Vol. 33. <http://dx.doi.org/10.1007/978-1-4613-8952-1>.

Komen, G.J., Cavalieri, L., Donelan, M., Hasselmann, K., Hasselmann, S., Janssen, P.A.E.M., 1994. *Dynamics and Modelling of Ocean Waves*. Cambridge University Press, <http://dx.doi.org/10.1017/CBO9780511628955>.

Komen, G.J., Hasselmann, S., Hasselmann, K., 1984. On the existence of a fully developed wind-sea spectrum. *J. Phys. Oceanogr.* 14 (8), 1271–1285. [http://dx.doi.org/10.1175/1520-0485\(1984\)014<1271:OTEAOF>2.0.CO;2](http://dx.doi.org/10.1175/1520-0485(1984)014<1271:OTEAOF>2.0.CO;2), URL [https://journals.ametsoc.org/view/journals/phoc/14/8/1520-0485\\_1984\\_014\\_1271\\_oteaoft\\_2\\_0\\_co\\_2.xml](https://journals.ametsoc.org/view/journals/phoc/14/8/1520-0485_1984_014_1271_oteaoft_2_0_co_2.xml).

LeBlond, P., Mysak, L., 1981. Waves in the Ocean. In: ISSN, Elsevier Science, URL [https://books.google.com/books?id=TYG1wzz\\_k8kC](https://books.google.com/books?id=TYG1wzz_k8kC).

LeVeque, R.J., Leveque, R.J., 1992. *Numerical Methods for Conservation Laws*, Vol. 214. Springer.

Loveland, M., 2023. *WAVEx: A New Spectral Wind Wave Model Using Stabilized Finite Elements* (Ph.D. thesis). The University of Texas at Austin.

Loveland, M., Valseth, E., Lukac, M., Dawson, C., 2022. Extending fenics to work in higher dimensions using tensor product finite elements. *J. Comput. Sci.* 64, 101831.

McRae, A.T., Bercea, G.-T., Mitchell, L., Ham, D.A., Cotter, C.J., 2016. Automated generation and symbolic manipulation of tensor product finite elements. *SIAM J. Sci. Comput.* 38 (5), S25–S47.

Meixner, J., 2013. *Discontinuous Galerkin Methods for Spectral Wave/Circulation Modeling* (Ph.D. thesis).

Meixner, J., Dietrich, J.C., Dawson, C., Zijlema, M., Holthuijsen, L.H., 2014. A discontinuous Galerkin coupled wave propagation/circulation model. *J. Sci. Comput.* 59 (2), 334–370. <http://dx.doi.org/10.1007/s10915-013-9761-5>.

Monbaliu, J., 2003. Chapter 5 spectral wave models in coastal areas. In: Lakhan, V. (Ed.), *Advances in Coastal Modeling*. In: Elsevier Oceanography Series, vol. 67, Elsevier, pp. 133–158. [http://dx.doi.org/10.1016/S0422-9894\(03\)80122-8](http://dx.doi.org/10.1016/S0422-9894(03)80122-8), URL <https://www.sciencedirect.com/science/article/pii/S0422989403801228>.

Morton, K., Priestley, A., Suli, E., 1988. Stability of the Lagrange-Galerkin method with non-exact integration. *ESAIM Math. Model. Numer. Anal.* 22 (4), 625–653.

Oden, J.T., Demkowicz, L.F., 2017. *Applied Functional Analysis*. Chapman and Hall/CRC.

Patankar, S.V., 1980. *Numerical Heat Transfer and Fluid Flow*. CRC Press.

Pironneau, O., Pironneau, O., 1989. *Finite Element Methods for Fluids*. Wiley Chichester.

Qi, J., Chen, C., Beardsley, R.C., Perrie, W., Cowles, G.W., Lai, Z., 2009. An unstructured-grid finite-volume surface wave model (FVCOM-SWAVE): Implementation, validations and applications. *Ocean Model.* 28 (1), 153–166. <http://dx.doi.org/10.1016/j.ocemod.2009.01.007>, URL <https://www.sciencedirect.com/science/article/pii/S1463500309000067>, The Sixth International Workshop on Unstructured Mesh Numerical Modelling of Coastal, Shelf and Ocean Flows.

Reddy, J.N., 2019. *Introduction to the Finite Element Method*. McGraw-Hill Education.

Ris, R., Holthuijsen, L., Smith, J., Booij, N., van Dongeren, A., 2003. The ONR TEST bed for coastal and oceanic wave models. ISBN: 9789812382382, pp. 380–391. [http://dx.doi.org/10.1142/9789812791306\\_0033](http://dx.doi.org/10.1142/9789812791306_0033).

Roland, A., 2008. *Development of WWM II: Spectral Wave Modeling on Unstructured Meshes* (Ph.D. thesis).

Roland, A., 2012. Application of residual distribution (RD) schemes to the geographical part of the wave action equation. In: ECMWF Workshop on Ocean Waves. pp. 25–27.

Roland, A., Arduhin, F., 2014. On the developments of spectral wave models: numerics and parameterizations for the coastal ocean. *Ocean Dyn.* 64, 833–846.

Roland, A., Cucco, A., Ferrarin, C., Hsu, T.-W., Liau, J.-M., Ou, S.-H., Umgessner, G., Zanke, U., 2009. On the development and verification of a 2-D coupled wave-current model on unstructured meshes. *J. Mar. Syst.* 78, S244–S254.

Roland, A., Zanke, U., Hsu, T.-W., Ou, S.-H., Liau, J.-M., 2006. Spectral wave modelling on unstructured grids with the WWM (Wind Wave Model) I: The deep water case. In: *Third Chinese–German Joint Symposium on Coastal and Ocean Engineering*, Tainan, Taiwan. p. 17.

Smith, J.M., Sherlock, A.R., Resio, D.T., 2001. *STWAVE: Steady-State Spectral Wave Model User's Manual for STWAVE, Version 3.0*. Tech. Rep., Engineer Research and Development Center Vicksburg MS Coastal and Hydraulics Lab.

Strang, G., 1968. On the construction and comparison of difference schemes. *SIAM J. Numer. Anal.* 5 (3), 506–517. <http://dx.doi.org/10.1137/0705041>, arXiv:<https://doi.org/10.1137/0705041>.

Team, S., et al., 2022. SWAN, Scientific and Technical Documentation, SWAN Cycle III version 41.41. Delft University of Technology, <http://www.swan.tudelft.nl> URL <https://swanmodel.sourceforge.io/download/zip/swantech.pdf>.

Tolman, H.L., 1991. A third-generation model for wind waves on slowly varying, unsteady, and inhomogeneous depths and currents. *J. Phys. Oceanogr.* 21 (6), 782–797. [http://dx.doi.org/10.1175/1520-0485\(1991\)021<0782:ATGMFW>2.0.CO;2](http://dx.doi.org/10.1175/1520-0485(1991)021<0782:ATGMFW>2.0.CO;2), URL [https://journals.ametsoc.org/view/journals/phoc/21/6/1520-0485\\_1991\\_021\\_0782\\_atgmfw\\_2\\_0\\_co\\_2.xml](https://journals.ametsoc.org/view/journals/phoc/21/6/1520-0485_1991_021_0782_atgmfw_2_0_co_2.xml).

Yildirim, B., Karniadakis, G.E., 2012. A hybrid spectral/DG method for solving the phase-averaged ocean wave equation: Algorithm and validation. *J. Comput. Phys.* 231 (14), 4921–4953. <http://dx.doi.org/10.1016/j.jcp.2012.04.013>, URL <https://www.sciencedirect.com/science/article/pii/S0021999112001854>.

Young, I.R., 1999. *Wind Generated Ocean Waves*. Elsevier.

Zienkiewicz, O.C., Taylor, R.L., Nithiarasu, P., 2013. *The Finite Element Method for Fluid Dynamics*. Butterworth-Heinemann.

Zienkiewicz, O.C., Taylor, R.L., Zhu, J.Z., 2005. *The Finite Element Method: Its Basis and Fundamentals*. Elsevier.

Zijlema, M., 2010. Computation of wind-wave spectra in coastal waters with SWAN on unstructured grids. *Coast. Eng.* 57 (3), 267–277.