Downloaded from orbit.dtu.dk on: Aug 16, 2022

DTU Library

A new transform based FourierLegendreGalerkin model for nonlinear water waves

Klahn, Mathias; Madsen, Per A.; Fuhrman, David R.

Published in:
International Journal for Numerical Methods in Fluids

Link to article, DOI:
10.1002/fld.4881

Publication date:
2021

Document Version
Peer reviewed version

Link back to DTU Orbit

Citation (APA):
Klahn, M., Madsen, P. A., & Fuhrman, D. R. (2021). A new transform based FourierLegendreGalerkin model for nonlinear water waves. *International Journal for Numerical Methods in Fluids*, *93*(1), 220-248. https://doi.org/10.1002/fld.4881

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

A new σ-transform based Fourier-Legendre-Galerkin model for nonlinear water waves

Mathias Klahn* | Per A. Madsen | David R. Fuhrman

[1]Department of Mechanical Engineering, Technical University of Denmark, DK-2800 Kgs. Lyngby, Denmark

Correspondence
*Correspondence to: Mathias Klahn, Department of Mechanical Engineering, Technical University of Denmark, Niels Koppels Alle Bygning 403. Email: matkla@mek.dtu.dk

Abstract

This paper presents a new spectral model for solving the fully nonlinear potential flow problem for water waves in a single horizontal dimension. At the heart of the numerical method is the solution to the Laplace equation which is solved using a variant of the σ-transform. The method discretizes the spatial part of the governing equations using the Galerkin method and the temporal part using the classical fourth order Runge-Kutta method. A careful investigation of the numerical method's stability properties is carried out, and it is shown that the method is stable up to a certain threshold steepness when applied to nonlinear monochromatic waves in deep water. Above this threshold artificial damping may be employed to obtain stable solutions. The accuracy of the model is tested for: (1) highly nonlinear progressive wave trains, (2) solitary wave reflection, and (3) deep water wave focusing events. In all cases it is demonstrated that the model is capable of obtaining excellent results, essentially up to very near breaking.

KEYWORDS:
Nonlinear water waves; Spectral methods; Potential flow; σ-transform; Accuracy; Stability

1 | INTRODUCTION

It is well known that a large number of nonlinear water wave phenomena can be simulated accurately and efficiently by the means of fully nonlinear potential flow theory. As a natural consequence, a wide variety of numerical methods to deal with the potential flow problem for water waves have been developed over the years. Despite the fact that they differ both in their mathematical formulations and numerical discretizations, the methods all share the same computational bottleneck, namely, the solution of the Laplace equation inherent to potential flow. While obtaining the solution to the Laplace equation efficiently is crucial to any numerical method, the same is true for the method's ability to compute the solution accurately. In fact, if long term simulations are to be carried out, the results may not be reliable if the kinematics of the flow are not computed to high precision.

Driven by the desire to meet both requirements, much attention has recently been given to methods which solve the Laplace problem in its full dimensionality using variants of the so-called σ-transform. This transform refers to a nonconformal change of coordinates induced by a stretching of the vertical coordinate which maps the fluid domain to a time independent rectangular box. Although it is merely a special case of coordinate changes found in textbooks on numerical methods (see e.g. chapter 6 in the book by Kopriva[1]) it has acquired a special name due to its widespread use. The σ-transform was introduced in the field of meteorology by Phillips[2] as far back as in 1957, and utilized by Freeman et al.[3] for hydrodynamic modelling of lakes, and by Johns[4] in connection with modelling of tidal flow in a channel. Some of the first to utilize the σ-transform for wave simulations were Li & Fleming[5], who employed it together with a second order finite difference based multigrid strategy. Applying their procedure

This article has been accepted for publication and undergone full peer review but has not been through the copyediting, typesetting, pagination and proofreading process which may lead to differences between this version and the Version of Record. Please cite this article as doi: 10.1002/fld.4881

This article is protected by copyright. All rights reserved.

Accepted Article

Accepted Article

in three dimensions they were, among other things, able to accurately reproduce experimental results for wave propagation over an elliptic shoal. Ten years later Bingham & Zhang[6] developed a two dimensional method which extended the method of Li & Fleming by allowing for arbitrary order finite difference schemes. Using an effective preconditioner, Bingham & Zhang demonstrated that the Laplace equation could be solved with a computational effort proportional to the number of degrees of freedom by utilizing the GMRES method. Engsig-Karup et al.[7] extended this method to three dimensions and showed that their implementation based on a multigrid strategy could likewise solve the Laplace equation with optimal scaling. A detailed comparison carried out by Yates & Benoit[8] in two dimensions showed that the number of degrees of freedom used to discretize the vertical dimension could be reduced by using the Chebyshev-Tau method instead of a finite difference approximation. This was very recently utilized by Raoult et al.[9] who developed a three dimensional method based on the Chebyshev-Tau method in the vertical direction and radial basis functions in the two horizontal dimensions.

The above mentioned methods all employ finite difference discretizations to some extent and thus illustrate the fact that this type of discretization remains useful and popular within the field of nonlinear potential flow. Interestingly, Christiansen et al.[10] recently showed that methods based on spectral discretizations in both the horizontal and vertical dimensions have the potential to be orders of magnitude more accurate than the finite difference based methods when the same number of degrees of freedom are used, while using approximately the same amount of time to solve the Laplace equation. Using their method, they found it necessary to introduce artificial damping in the time integration in order to stabilize the propagation of steep waves and argued that the instability was due to the aliasing phenomena as the method is based on a Fourier collocation method in the horizontal dimension. The complete potential of the method was, however, not revealed as its performance was only evaluated on a very limited number of test cases.

Motivated by these facts, this paper will describe and thoroughly test a new method based on spectral discretizations in both the vertical and horizontal dimensions free of aliasing errors. It differs from the method of Christiansen et al. by the fact that it is based on a Legendre-Tau method instead of a Chebyshev-Tau method in the vertical dimension and a Fourier Galerkin method in the horizontal dimension. While the Legendre-Tau and Chebyshev-Tau methods are quite similar both in terms of mathematical formulation and numerical accuracy, the Fourier Galerkin and collocation methods differ significantly in both aspects. Whereas the difference in mathematical formulation is not particularly important for practical purposes, the difference in accuracy due to aliasing errors from nonlinear terms certainly is. We note that this is especially true in the case of water wave simulations, as the governing equations contain nonlinear products up to fourth order. To do justice to the collocation method we mention here that it is in fact algebraically equivalent to the Galerkin method when fully dealiased through the zero padding technique. In this regard, the new method can be seen as an extension to that of Christiansen et al.

The structure of the present paper is as follows: In Section 2 we present the physical model and the governing equations. In Section 3 we derive the numerical method to integrate the state of the system in time and describe how physical quantities are calculated. The accuracy, stability and efficiency of the method are evaluated in Section 4, before conclusions are finally drawn in Section 5.

2 | PHYSICAL SYSTEM AND GOVERNING EQUATIONS

We consider the horizontally periodic motion of an incompressible and irrotational fluid moving in the (x, z)-plane. At any time t the fluid is bounded below by the impermeable flat seabed and above by its free surface, which we will denote by $\eta(x, t)$. We choose the z-axis such that the still water level is located at $z = 0$ and the seabed is located at $z = -h$ and we choose the x-axis such that the horizontal motion has spatial period L, meaning that we only need to consider x-values between 0 and L.

The time evolution of the system is governed by the kinematic and dynamic boundary conditions at the free surface in combination with the fact that the velocity potential $\Phi(x, z, t)$ satisfies the Laplace equation within the region of space occupied by the fluid. Differentiating $\Phi(x, z, t)$ with respect to x and z yields the horizontal and vertical velocities which we will denote $u(x, z, t)$ and $w(x, z, t)$, respectively, in the following. Furthermore, we will denote quantities evaluated at the free surface with a subscript s, i.e. $u_s(x, t) = u(x, \eta(x, t), t)$, $w_s(x, t) = w(x, \eta(x, t), t)$ and $\Phi_s(x, t) = \Phi(x, \eta(x, t), t)$. Following Zakharov[11], the kinematic free surface boundary condition can be written in terms of surface quantities as

$$\frac{\partial \eta}{\partial t} = w_s - u_s \frac{\partial \eta}{\partial x}, \tag{1}$$

This article is protected by copyright. All rights reserved.

Accepted Article

stating that a particle once at the free surface will remain there. Likewise, the dynamic free surface boundary condition can be written as

$$\frac{\partial \Phi_s}{\partial t} = -g\eta - \frac{1}{2}\left(\frac{\partial \Phi_s}{\partial x}\right)^2 + \frac{1}{2}w_s^2\left(1 + \left(\frac{\partial \eta}{\partial x}\right)^2\right), \tag{2}$$

and it accounts for the fact that the pressure is constant along the free surface. Here $g = 9.81\ \mathrm{m/s^2}$ is the gravitational acceleration. We express the complete Laplace problem satisfied by the velocity potential inside the fluid domain as

$$\frac{\partial^2 \Phi}{\partial x^2} + \frac{\partial^2 \Phi}{\partial z^2} = 0, \qquad -h \le z \le \eta(x,t), \tag{3a}$$

$$\frac{\partial \Phi}{\partial z} = 0, \qquad z = -h, \tag{3b}$$

$$\Phi(0,z,t) = \Phi(L,z,t). \tag{3c}$$

Here (3b) is a consequence of the seabed being impermeable, while (3c) accounts for the assumed periodicity in the horizontal direction.

3 | NUMERICAL METHODS

This section describes how $\eta(x,t)$ and $\Phi_s(x,t)$ are integrated in time as well as how physical quantities are computed and how artificial damping is employed, when necessary. The time integration of $\eta(x,t)$ and $\Phi_s(x,t)$ is rather comprehensive and is for that reason divided into three parts: Section 3.1 describes how $\eta(x,t)$ and $\Phi_s(x,t)$ are integrated in time assuming that $u_s(x,t)$ and $w_s(x,t)$ are known, Section 3.2 explains how the Laplace problem (3) is solved and finally Section 3.3 shows how the surface velocities are computed when the solution to the Laplace problem is known.

The method is based on the Galerkin method (see e.g. the book by Kopriva[1]) which is closely connected to the concept of L^2-spaces as it discretizes the weak form of a differential equation. To ease the notation at later stages we here make a general definition of an L^2 space which covers the cases that will be used in the following. For the set $S \subset \mathbb{R}^n$ $(n = 1,2)$ and the inner product

$$\langle\, f(\mathbf{x}) \,|\, g(\mathbf{x}) \,\rangle = \alpha \int\limits_S \overline{f(\mathbf{x})} g(\mathbf{x})\, \mathrm{d}\mathbf{x}, \tag{4}$$

where $\mathbf{x} \in \mathbb{R}^n$, α is a positive real constant and the overline denotes complex conjugation, we define the space $L^2[S]$ as the space of functions with domain S that have a finite norm:

$$L^2[S] = \left\{ f : S \mapsto \mathbb{C} \,\middle|\, \langle\, f \,|\, f \,\rangle < \infty \right\}. \tag{5}$$

In this setting it is natural to represent functions by their expansion coefficients with respect to a certain set of basis functions for $L^2[S]$. In fact, during simulations we keep all functions in the coefficient representation and postpone the evaluation of functions to the post processing step.

3.1 | Time integration of the time dependent boundary conditions

The functions of relevance for the time dependent boundary conditions are $\eta(x,t)$, $\Phi_s(x,t)$, $u_s(x,t)$ and $w_s(x,t)$. As we do not consider overturning or breaking waves it is safe to assume that these functions are elements in $C^\infty([0,L])$ and hence also belong to the vectorspace $L^2[0,L]$ equipped with the inner product

$$\left\langle f(x) \,\middle|\, g(x) \right\rangle_{\mathrm{1D}} = \frac{1}{L}\int\limits_0^L \overline{f(x)} g(x)\, \mathrm{d}x. \tag{6}$$

On top of this, the functions are periodic by assumption. Combining the periodicity with the smoothness, it can be shown that the functions have Fourier series representations whose coefficients decay at a rate faster than any algebraic rate[1]. Accordingly,

This article is protected by copyright. All rights reserved.

Accepted Article

we represent the functions as

$$\eta(x,t) = \sum_{n=-N}^{N} \widehat{\eta}_n \exp(ik_n x), \tag{7a}$$

$$\Phi_s(x,t) = \sum_{n=-N}^{N} \widehat{\Phi}_n \exp(ik_n x), \tag{7b}$$

$$u_s(x,t) = \sum_{n=-N}^{N} \widehat{u}_n \exp(ik_n x), \tag{7c}$$

$$w_s(x,t) = \sum_{n=-N}^{N} \widehat{w}_n \exp(ik_n x), \tag{7d}$$

where the expansion coefficients are functions of time only and $k_n = nk$ with $k = 2\pi/L$. To utilize these representations we rewrite (1) and (2) into their weak forms, i.e.

$$\left\langle \mathcal{T} \,\middle|\, \frac{\partial \eta}{\partial t} \right\rangle_{1\mathrm{D}} = \left\langle \mathcal{T} \,\middle|\, w_s - u_s \frac{\partial \eta}{\partial x} \right\rangle_{1\mathrm{D}}, \tag{8}$$

and

$$\left\langle \mathcal{T} \,\middle|\, \frac{\partial \Phi_s}{\partial t} \right\rangle_{1\mathrm{D}} = \left\langle \mathcal{T} \,\middle|\, -g\eta - \frac{1}{2}\left(\frac{\partial \Phi_s}{\partial x}\right)^2 + \frac{1}{2} w_s^2 \left(1 + \left(\frac{\partial \eta}{\partial x}\right)^2\right) \right\rangle_{1\mathrm{D}}, \tag{9}$$

where $\mathcal{T}(x)$ can be any function belonging to $L^2[0, L]$. In order to discretize these equations we employ the Galerkin method, meaning that we choose the expansion coefficients of $\eta(x,t)$, $\Phi_s(x,t)$, $u_s(x,t)$ and $w_s(x,t)$ such that the equations at all times are satisfied for $\mathcal{T}(x) = \exp(ik_p x)$ with $p \in \{-N, -N+1, ..., N\}$. In this way the equations hold true for any $\mathcal{T}(x)$ in the subspace spanned by the set of functions $\{\exp(ik_p x) \,|\, p = -N, -N+1, ..., N\}$. Setting $\mathcal{T}(x) = \exp(ik_p x)$ and performing the integrals, (8) becomes

$$\frac{\mathrm{d}\widehat{\eta}_p}{\mathrm{d}t} = \widehat{w}_p - \sum_{n_1=-N}^{N} \sum_{n_2=-N}^{N} \widehat{u}_{n_1} (ik_{n_2} \widehat{\eta}_{n_2})\, \delta_{n_1+n_2,p}, \tag{10}$$

where $\delta_{n,p}$ denotes the Kronecker delta. Similarly, the Galerkin method turns (9) into

$$\begin{aligned} \frac{\mathrm{d}\widehat{\Phi}_p}{\mathrm{d}t} = &- g\eta_p - \frac{1}{2} \sum_{n_1=-N}^{N} \sum_{n_2=-N}^{N} \left(ik_{n_1} \widehat{\Phi}_{n_1}\right)\left(ik_{n_2} \widehat{\Phi}_{n_2}\right) \delta_{n_1+n_2,p} + \frac{1}{2} \sum_{n_1=-N}^{N} \sum_{n_2=-N}^{N} \widehat{w}_{n_1} \widehat{w}_{n_2} \delta_{n_1+n_2,p} \\ &+ \frac{1}{2} \sum_{n_1=-N}^{N} \sum_{n_2=-N}^{N} \sum_{n_3=-N}^{N} \sum_{n_4=-N}^{N} \widehat{w}_{n_1} \widehat{w}_{n_2} \left(ik_{n_3} \widehat{\eta}_{n_3}\right)\left(ik_{n_4} \widehat{\eta}_{n_4}\right) \delta_{n_1+n_2+n_3+n_4,p}. \end{aligned} \tag{11}$$

Together, (10) and (11) govern the time evolution of the expansion coefficients of $\eta(x,t)$ and $\Phi_s(x,t)$. In order to integrate these equations in time we have used the classical 4th order Runge Kutta method with a constant time step, which offers a good tradeoff between accuracy and computational work. In fact, using the method outlined in A, the right hand sides of (10) and (11) can be evaluated in $O(N \log(N))$ operations when the fast Fourier transform is employed, and the method thus enables efficient time integration.

3.2 | The Laplace problem

The first step towards obtaining the Fourier coefficients of $u_s(x,t)$ and $w_s(x,t)$ from the Fourier coefficients of $\eta(x,t)$ and $\Phi_s(x,t)$ is to solve the Laplace problem (3). We note that once the coefficients of $\eta(x,t)$ and $\Phi_s(x,t)$ have been specified, this problem does not involve time. For notational convenience we shall therefore suppress the explicit time dependence of the functions in Section 3.2, Section 3.3 and Section 3.4.

This article is protected by copyright. All rights reserved.

Accepted Article

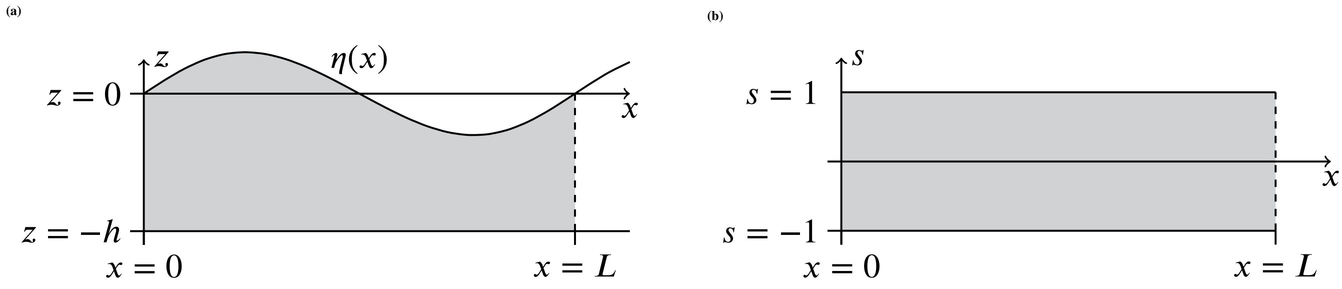

FIGURE 1 The shaded regions illustrate (a) the fluid domain Ω_F and (b) the time independent rectangle Ω_R.

3.2.1 | Coordinate transformation

The main reason why solving the Laplace problem (3) is difficult is that the fluid domain

$$\Omega_F = \left\{(x, z) \in \mathbb{R} \,\middle|\, 0 \leq x < L,\ \ -h \leq z \leq \eta(x)\right\}, \tag{12}$$

is in general not a rectangle, meaning that we cannot seek an approximate solution which is a product of functions depending on a single spatial coordinate each. To overcome this problem we map Ω_F to the time independent rectangle

$$\Omega_R = \left\{(x, s) \in \mathbb{R} \,\middle|\, 0 \leq x < L,\ \ -1 \leq s \leq 1\right\}, \tag{13}$$

through a coordinate transform similar to the σ-transform. More specifically, the mapping of Ω_F to Ω_R is facilitated by the change of coordinates $(x, z) \mapsto (x, s)$ where s is defined by

$$s(x, z) = \frac{2z + h - \eta(x)}{h + \eta(x)}. \tag{14}$$

For all x it is clear that $s(x, z)$ lies in the interval $[-1, 1]$ since z lies in the interval $[-h, \eta(x)]$. The coordinate transform is illustrated in Figure 1. Defining the function $F(x, s(x, z)) = \Phi(x, z)$, it follows directly from the Laplace equation, the definition of $s(x, z)$ and the chain rule, that $F(x, s)$ must satisfy the equation

$$0 = \frac{\partial^2 F}{\partial x^2} + \left(\left(\frac{\partial s}{\partial x}\right)^2 + \left(\frac{\partial s}{\partial z}\right)^2\right)\frac{\partial^2 F}{\partial s^2} + 2\frac{\partial s}{\partial x}\frac{\partial^2 F}{\partial x \partial s} + \frac{\partial^2 s}{\partial x^2}\frac{\partial F}{\partial s}. \tag{15}$$

Inserting the derivatives of $s(x, z)$ into this equation and multiplying both sides by $(h + \eta(x))^2$ yields

$$\begin{aligned} 0 =& (h + \eta)^2 \frac{\partial^2 F}{\partial x^2} + \left(4 + (1 + s)^2 \left(\frac{\partial \eta}{\partial x}\right)^2\right)\frac{\partial^2 F}{\partial s^2} \\ & - 2(h + \eta)\frac{\partial \eta}{\partial x}(1 + s)\frac{\partial^2 F}{\partial x \partial s} + \left(2\left(\frac{\partial \eta}{\partial x}\right)^2 - (h + \eta)\frac{\partial^2 \eta}{\partial x^2}\right)(1 + s)\frac{\partial F}{\partial s}. \end{aligned} \tag{16}$$

The boundary conditions of $F(x, s)$ are derived from the boundary conditions of $\Phi(x, z)$ and read

$$\frac{\partial F}{\partial s}(x, -1) = 0, \tag{17a}$$

$$F(x, 1) = \Phi_s(x), \tag{17b}$$

$$F(0, s) = F(L, s). \tag{17c}$$

Together (16) and (17) constitute the problem that we solve to obtain $F(x, s)$ and in turn $\Phi(x, z)$.

3.2.2 | Discretization of the transformed Laplace equation

At the cost of a more complicated differential equation, we have succeeded in translating the Laplace problem (3) to an equivalent problem on a rectangular domain, namely (16) with the boundary conditions (17). To solve this problem numerically we seek a

This article is protected by copyright. All rights reserved.

Accepted Article

solution which is an element in $L^2\big([0,L]\times[-1,1]\big)$ having the inner product

$$\Big\langle f(x,s) \,\Big|\, g(x,s) \Big\rangle_{2\mathrm{D}} = \frac{1}{L}\int\limits_0^L \int\limits_{-1}^1 \overline{f(x,s)}g(x,s)\,\mathrm{d}x\,\mathrm{d}s. \tag{18}$$

As $F(x,s)$ needs to be periodic in x due to the boundary condition (17c), but is not periodic in the s-coordinate, we approximate $F(x,s)$ by

$$F(x,s) = \sum_{n=-N}^{N}\sum_{m=0}^{M} \widehat{F}_{n,m}\exp(ik_n x)L_m(s), \tag{19}$$

where $L_m(s)$ is the mth Legendre polynomial normalized such that $L_m(1) = 1$. We have chosen to expand the vertical part of $F(x,s)$ in Legendre polynomials as these constitute a basis for $L^2[-1,1]$ and because they are easy to work with as will be explained in the following. The Legendre polynomials can be computed using the recurrence relation

$$L_m(s) = \frac{2m-1}{m}sL_{m-1}(s) - \frac{m-1}{m}L_{m-2}(s), \tag{20}$$

when combined with the fact that the first two Legendre polynomials are $L_0(s) = 1$ and $L_1(s) = 1$. Two important and useful properties of the Legendre polynomials are that they satisfy the orthogonality relation

$$\int\limits_{-1}^{1} L_m(s)L_q(s)\,\mathrm{d}s = \frac{2}{2m+1}\delta_{m,q}, \tag{21}$$

and that they are connected to their derivatives through the relation

$$(2m+1)L_m(s) = \frac{\mathrm{d}L_{m+1}}{\mathrm{d}s}(s) - \frac{\mathrm{d}L_{m-1}}{\mathrm{d}s}(s). \tag{22}$$

Having introduced the approximate form of $F(x,s)$, we now turn our attention to the discretization of (16) and (17). Since the boundary conditions are one-dimensional, their weak form involves the use of the inner product (6):

$$\left\langle \mathcal{T} \,\middle|\, \frac{\partial F}{\partial s}\bigg|_{s=-1} \right\rangle_{1\mathrm{D}} = 0 \qquad \text{and} \qquad \Big\langle \mathcal{T} \,\Big|\, F\Big|_{s=1} \Big\rangle_{1\mathrm{D}} = \Big\langle \mathcal{T} \,\Big|\, \Phi_s \Big\rangle_{1\mathrm{D}}. \tag{23}$$

Setting $\mathcal{T}(x) = \exp(ik_p x)$ with $p \in \{-N, -N+1, ..., N\}$ and inserting the approximation (7b) for $\Phi_s(x)$ yields the linear equations

$$\sum_{m=0}^{M}(-1)^m \widehat{F}^{(0,1)}_{p,m} = 0, \tag{24a}$$

$$\sum_{m=0}^{M}\widehat{F}_{p,m} = \widehat{\Phi}_p. \tag{24b}$$

Here $\widehat{F}^{(0,1)}_{n,m}$ denotes the (n,m)th expansion coefficients of the function $\partial F/\partial s$. More generally, we define $\widehat{F}^{(p,q)}_{n,m}$ as the (n,m)th expansion coefficient of the function $\partial^{p+q}F/\partial^p x\partial^q s$, and it is explained in B how these coefficients are computed from the expansion coefficients of $F(x,s)$. The weak form of (16) stems from the inner product (18) and reads

$$\begin{aligned}0 =& \left\langle \mathcal{R} \,\middle|\, (h+\eta)^2\frac{\partial^2 F}{\partial x^2} + \left(4 + (1+s)^2\left(\frac{\partial \eta}{\partial x}\right)^2\right)\frac{\partial^2 F}{\partial s^2} \right\rangle_{2\mathrm{D}} \\ &+ \left\langle \mathcal{R} \,\middle|\, -2(h+\eta)(1+s)\frac{\partial \eta}{\partial x}\frac{\partial^2 F}{\partial x \partial s} + (1+s)\left(2\left(\frac{\partial \eta}{\partial x}\right)^2 - (h+\eta)\frac{\partial^2 \eta}{\partial x^2}\right)\frac{\partial F}{\partial s} \right\rangle_{2\mathrm{D}},\end{aligned} \tag{25}$$

where $\mathcal{R}(x,s)$ can be any function in $L^2\big([0,L]\times[-1,1]\big)$. To discretize this equation we employ the Galerkin method by setting $\mathcal{R}(x,s) = \exp(ik_p x)L_q(s)$ with $p \in \{-N, -N+1..., N\}$ and $q \in \{0, 1, ..., M-2\}$. Inserting the approximations for $F(x,s)$

This article is protected by copyright. All rights reserved.

Accepted Article

and $\eta(x)$ into the equation and carrying out the integrals yields the linear equation

$$\begin{aligned}
0 = \frac{2}{2q+1} \sum_{n_1=-N}^{N} \sum_{n_2=-N}^{N} \sum_{n_3=-N}^{N} \widehat{F}^{(2,0)}_{n_1,q} \,(h\delta_{n_2,0} + \widehat{\eta}_{n_2})(h\delta_{n_3,0} + \widehat{\eta}_{n_3})\delta_{n_1+n_2+n_3,p} + \frac{8}{2q+1}\widehat{F}^{(0,2)}_{p,q} \\
+ \sum_{m=0}^{M} A_{m,q}\Bigg(\sum_{n_1=-N}^{N} \sum_{n_2=-N}^{N} \sum_{n_3=-N}^{N} \widehat{F}^{(0,2)}_{n_1,m}(ik_{n_2}\widehat{\eta}_{n_2})(ik_{n_3}\widehat{\eta}_{n_3})\delta_{n_1+n_2+n_3,p}\Bigg) \\
+ \sum_{m=0}^{M} B_{m,q}\Bigg(\sum_{n_1=-N}^{N} \sum_{n_2=-N}^{N} \sum_{n_3=-N}^{N} \Big(- 2\widehat{F}^{(1,1)}_{n_1,m}(h\delta_{n_2,0} + \widehat{\eta}_{n_2})(ik_{n_3}\widehat{\eta}_{n_3}) \\
+ 2\widehat{F}^{(0,1)}_{n_1,m}(ik_{n_2}\widehat{\eta}_{n_2})(ik_{n_3}\widehat{\eta}_{n_3}) - \widehat{F}^{(0,1)}_{n_1,m}(h\delta_{n_2,0} + \widehat{\eta}_{n_2})\big((ik_{n_3})^2\widehat{\eta}_{n_3}\big)\Big)\delta_{n_1+n_2+n_3,p}\Bigg),
\end{aligned} \tag{26}$$

where the numbers $A_{m,q}$ and $B_{m,q}$ are given by the integrals

$$A_{m,q} = \int_{-1}^{1} (1+s)^2 L_m(s) L_q(s)\,\mathrm{d}s \qquad \text{and} \qquad B_{m,q} = \int_{-1}^{1} (1+s) L_m(s) L_q(s)\,\mathrm{d}s. \tag{27}$$

From the recurrence relation (20) and the orthogonality relation (21) it follows that $A_{m,q} = 0$ if $|m-q| > 2$ and that $B_{m,q} = 0$ if $|m-q| > 1$. The matrices having $A_{m,q}$ and $B_{m,q}$ as their elements are thus penta- and tridiagonal, respectively. Furthermore, they are clearly independent of time and need therefore only to be calculated once and for all at the beginning of a simulation. Although this calculation could be done exactly using the relations (20) and (21), we compute $A_{m,q}$ and $B_{m,q}$ numerically using a Gauss-Legendre quadrature with $M+1$ points. This quadrature integrates polynomials of degree less than or equal to $2M$ exactly, and since $q \leq M-2$, the numerical computation gives the exact result.

3.2.3 | Efficient solution of the discretized Laplace equation

The expansion coefficients of $F(x,s)$ are obtained by solving the $(2N+1)(M+1)$ linear equations (24) and (26). Solving these equations efficiently is crucial for the performance of the numerical method as it is the computational bottleneck. For that reason this section presents a solution strategy which has a significantly lower computational cost than a direct solution.

Now, (24) and (26) form a system of equations which may be written in the form $\mathcal{M}\widehat{F} = b$ where $\mathcal{M}$ is a matrix, $\widehat{F}$ is a vector containing the expansion coefficients of $F(x,s)$ and b is a vector. Actually constructing the matrix $\mathcal{M}$ is a rather complicated affair, but multiplying a vector by it can be done efficiently because the sums over n_1, n_2 and n_3 in (26) can be computed in $O(N\log(N))$ operations using the fast Fourier transform. A back-of-the-envelope operation count shows that the coefficients $\widehat{F}^{(p,q)}_{n,m}$ with $0 \leq p, q \leq 2$ can be computed from the coefficients $\widehat{F}_{n,m}$ in $O(M^2N)$ operations and once this is done the right hand sides of (24) and (26) can be computed in $O(MN\log(N))$ operations. In practice we will more often than not be in a situation where the number of basis functions used in the horizontal dimension is much larger than the number of basis functions used in the vertical dimension (i.e. $N \gg M$), meaning that we can apply the matrix $\mathcal{M}$ to a vector in $O(N\log(N))$ operations and for that reason we have chosen to solve the system of linear equations using the iterative method GMRES[12]. Assuming exact arithmitic the GMRES method is guaranteed to find the exact solution, but as this might take $(2N+1)(M+1)$ iterations we aim at finding a solution whose relative error, ϵ, is low enough to be accepted. In this context it comes in handy that the computational cost of the first few iterations is dominated by the cost of one matrix-vector product meaning that *if* the number of iterations needed for GMRES to obtain this relative error for some reason is independent of M and N we can solve the Laplace problem in $O(N\log(N))$ operations for large N. At the same time it is clear that the solution cannot be found in less than $O(N\log(N))$ operations implying that the optimal scaling for the method is $O(N\log(N))$.

In an attempt to obtain optimal scaling we have preconditioned the system of linear equations $\mathcal{M}\widehat{F} = b$ from the left using the linearized matrix $P = \mathcal{M}|_{\eta(x)=0}$. This strategy is inspired by the work of Fuhrman & Bingham[13], who considered several ways of preconditioning their finite difference Boussinesq wave model using linearized versions of their full matrix. The matrix P is time independent and has a sparse LU decomposition which can be computed at the beginning of a simulation. In this way P^{-1} may be multiplied onto a vector at a cost which is insignificant to that of the rest of the problem. On the other hand P clearly resembles $\mathcal{M}$ since they become equal in the case where the surface is flat. We therefore expect $P^{-1}\mathcal{M}$ to be well conditioned and in turn that only a few GMRES iterations will be needed to obtain the desired approximate solution.

This article is protected by copyright. All rights reserved.

Accepted Article

3.3 | Computation of $u_s(x,t)$ and $w_s(x,t)$

Having obtained the solution to the Laplace problem we are in a position where we can compute the Fourier coefficients of $u_s(x)$ and $w_s(x)$. By the chain rule and the definitions of $F(x,s)$ and $s(x,z)$ we find that

$$u_s(x) = \sum_{n=-N}^{N}\sum_{m=0}^{M}\left(\widehat{F}^{(1,0)}_{n,m} - \frac{2}{h+\eta(x)}\frac{\partial\eta}{\partial x}\widehat{F}^{(0,1)}_{n,m}\right)\exp(ik_n x), \tag{28}$$

from which it follows that the pth Fourier coefficient of $u_s(x)$ is given by

$$\widehat{u}_p = \sum_{m=0}^{M}\left(\widehat{F}^{(1,0)}_{p,m} - \frac{2}{L}\sum_{n=-N}^{N}\widehat{F}^{(0,1)}_{n,m}\int\limits_0^L \frac{1}{h+\eta(x)}\frac{\partial\eta}{\partial x}\exp(ik_{n-p}x)\,\mathrm{d}x\right). \tag{29}$$

Due to the factor $(h+\eta(x))^{-1}$ we have not been able to compute the integral in this expression exactly, and we have therefore chosen to make the approximation

$$\frac{1}{h+\eta}\frac{\partial\eta}{\partial x} = \sum_{n=-N}^{N} a_n\exp(ik_n x), \tag{30}$$

where

$$a_n = \frac{1}{2N+1}\sum_{r=0}^{2N}\frac{1}{h+\eta(x_r)}\frac{\partial\eta}{\partial x}(x_r)\exp(-ik_n x_r), \tag{31}$$

and $x_r = rL/(2N+1)$. Using this approximation we obtain the relation

$$\widehat{u}_p = \sum_{m=0}^{M}\left(\widehat{F}^{(1,0)}_{p,m} - 2\sum_{n_1=-N}^{N}\sum_{n_2=-N}^{N}\widehat{F}^{(0,1)}_{n_1,m}\,a_{n_2}\,\delta_{n_1+n_2,p}\right), \tag{32}$$

which is the one we use to compute the coefficients of $u_s(x)$. To compute the coefficients of $w_s(x)$ we again start from the definition of $F(x,s)$ and $s(x,z)$ and use the chain rule to arrive at the relation

$$w_s(x) = 2\sum_{n=-N}^{N}\sum_{m=0}^{M}\widehat{F}^{(0,1)}_{n,m}\frac{1}{h+\eta(x)}\exp(ik_n x). \tag{33}$$

Following the same procedure as for $u_s(x)$, this relation implies that the pth coefficient of $w_s(x)$ is given by

$$\widehat{w}_p = 2\sum_{n=-N}^{N}\sum_{m=0}^{M}\widehat{F}^{(0,1)}_{n,m}\frac{1}{L}\int\limits_0^L\frac{1}{h+\eta(x)}\exp(ik_{n-p}x)\,\mathrm{d}x. \tag{34}$$

Due to the presence of the factor $(h+\eta(x))^{-1}$ we have again not been able to calculate the integral in this equation exactly, and we therefore use the approximation

$$\widehat{w}_p = \frac{2}{2N+1}\sum_{n=-N}^{N}\sum_{m=0}^{M}\widehat{F}^{(0,1)}_{n,m}\left(\sum_{r=0}^{2N}\frac{1}{h+\eta(x_r)}\exp(ik_{n-p}x_r)\right). \tag{35}$$

This completes the description of the numerical method since we have now shown how the expansion coefficients of $u_s(x)$ and $w_s(x)$ can be computed when the coefficients of $\eta(x)$ and $\Phi_s(x)$ are known and how this procedure can be used to integrate the coefficients of $\eta(x)$ and $\Phi_s(x)$ in time.

3.4 | Computation of physical quantities

From the definition of the velocities $u(x,z)$ and $w(x,z)$, the definition of $F(x,s)$ and the chain rule it follows immediately that the velocity fields are given by

$$u(x,z) = \sum_{n=-N}^{N}\sum_{m=0}^{M}\widehat{F}^{(1,0)}_{n,m}\exp(ik_n x)L_m(s) - \frac{1}{h+\eta(x)}\frac{\partial\eta}{\partial x}(1+s)\sum_{n=-N}^{N}\sum_{m=0}^{M}\widehat{F}^{(0,1)}_{n,m}\exp(ik_n x)L_m(s), \tag{36}$$

This article is protected by copyright. All rights reserved.

Accepted Article

and

$$w(x,z) = \frac{2}{h+\eta(x)} \sum_{n=-N}^{N} \sum_{m=0}^{M} \widehat{F}^{(0,1)}_{n,m} \exp(ik_n x) L_m(s). \tag{37}$$

We note that the expression for $u(x,z)$ simplifies a great deal at locations where the free surface elevation is flat since we have $\partial\eta/\partial x = 0$ there.

To compute the pressure we start from the Euler equations and invoke the assumption that the motion of the fluid is irrotational. We further require the pressure at the free surface to be zero and find the well known result

$$\frac{p(x,z)}{\rho} = g\big(\eta(x) - z\big) + \frac{1}{2}\Big(u_s(x)^2 - u(x,z)^2\Big) + \frac{1}{2}\Big(w_s(x)^2 - w(x,z)^2\Big) + \left(\frac{\partial\Phi}{\partial t}(x,\eta(x)) - \frac{\partial\Phi}{\partial t}(x,z)\right), \tag{38}$$

where ρ is the density of the water. To evaluate the pressure we obviously need to know the quantity $\partial\Phi/\partial t$. From the Laplace problem (3) it follows that $\partial\Phi/\partial t$ must satisfy the very similar problem given by

$$\frac{\partial^2}{\partial x^2}\frac{\partial\Phi}{\partial t} + \frac{\partial^2}{\partial z^2}\frac{\partial\Phi}{\partial t} = 0, \qquad -h \le z \le \eta(x), \tag{39a}$$

$$\frac{\partial}{\partial z}\frac{\partial\Phi}{\partial t} = 0, \qquad z = -h, \tag{39b}$$

$$\frac{\partial\Phi}{\partial t} = \frac{\partial\Phi_s}{\partial t} - w_s\frac{\partial\eta}{\partial t}, \qquad z = \eta(x), \tag{39c}$$

$$\frac{\partial\Phi}{\partial t}(0,z) = \frac{\partial\Phi}{\partial t}(L,z). \tag{39d}$$

This Laplace problem is identical to (3) except for the boundary condition (39c) meaning that $\partial\Phi/\partial t$ can be computed in the same way as $\Phi(x,z)$ but with a modified boundary condition. Even though this method for computing $\partial\Phi/\partial t$ is considerably more expensive than using e.g. a finite difference approximation in the post processing of a time integration, it has at least two advantages. First, the method allows the computation of the pressure field (38) on the fly and at all times. Second, and more importantly, if $\partial\Phi/\partial t$ was calculated with a finite difference approximation, the spectral accuracy of the whole solution would be destroyed, something which is certainly not desirable.

Using (38) we can compute the dimensionless depth integrated force at the location $x = x_0$ as

$$\frac{\mathcal{F}}{\rho g h^2} = \frac{1}{gh^2}\int_{-h}^{\eta(x_0,t)} \frac{p(x_0,z)}{\rho}\,\mathrm{d}z. \tag{40}$$

We will limit ourselves to the case where $\partial\eta/\partial x(x_0) = 0$ as it simplifies the derivation and it is the only case that will be used. We compute the integral by writing u, w and $\partial\Phi/\partial t$ as the following Legendre series:

$$u(x_0, z(x_0,s)) = \sum_{m=0}^{M} \alpha_m L_m(s), \qquad \text{where} \qquad \alpha_m = \sum_{n=-N}^{N} \widehat{F}^{(1,0)}_{n,m} \exp(ik_n x_0), \tag{41a}$$

$$w(x_0, z(x_0,s)) = \sum_{m=0}^{M} \beta_m L_m(s), \qquad \text{where} \qquad \beta_m = \frac{2}{h+\eta(x_0)} \sum_{n=-N}^{N} \widehat{F}^{(0,1)}_{n,m} \exp(ik_n x_0), \tag{41b}$$

$$\frac{\partial\Phi}{\partial t}(x_0, z(x_0,s)) = \sum_{m=0}^{M} \gamma_m L_m(s), \qquad \text{where} \qquad \gamma_m = \sum_{n=-N}^{N} \widehat{G}_{n,m} \exp(ik_n x_0). \tag{41c}$$

Here $\widehat{G}_{n,m}$ denotes the (n,m)th expansion coefficient of the function $G = \partial\Phi/\partial t$ which is obtained by solving the Laplace problem (39). Using these expansions, combined with the relations (21) and (22) satisfied by the Legendre polynomials, we find

This article is protected by copyright. All rights reserved.

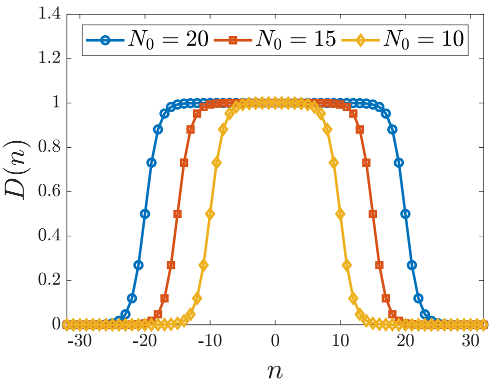

FIGURE 2 The function $D(n)$ for $N = 32$ and different values of N_0.

that the dimensionless force (40) is given by the expression

$$\begin{aligned}\frac{\mathcal{F}}{\rho g h^2} = {} & \frac{1}{2}\left(1+\frac{\eta(x_0)}{h}\right)^2 + \frac{1}{2}\left(1+\frac{\eta(x_0)}{h}\right)\left(\left(\frac{u_s(x_0)}{\sqrt{gh}}\right)^2+\left(\frac{w_s(x_0)}{\sqrt{gh}}\right)^2\right)\\ & -\frac{1}{2}\left(1+\frac{\eta(x_0)}{h}\right)\sum_{m=0}^{M}\frac{1}{2m+1}\left(\left(\frac{\alpha_m}{\sqrt{gh}}\right)^2+\left(\frac{\beta_m}{\sqrt{gh}}\right)^2\right)\\ & +\left(1+\frac{\eta(x_0)}{h}\right)\sum_{m=1}^{M}\frac{\gamma_m}{gh}.\end{aligned} \tag{42}$$

3.5 | Artificial Damping

We have found that the time integration of the above method is unstable when applied to a wave field which is locally very steep. The instability is, however, sufficiently weak that it can be remedied by using artificial damping. In this work we have employed artificial damping by multiplying the nth expansion coefficient of $\eta(x,t)$ and $\Phi_s(x,t)$ by the function $D(n)$ defined by

$$D(n) = \frac{1}{\exp(|n| - N_0) + 1}, \tag{43}$$

every time step. Here N_0 is a parameter to be chosen. The damping function is illustrated in Figure 2 for the case $N = 32$ and different values of N_0. The figure shows that $D(n)$ is approximately 1 for small $|n|$ and that $D(n)$ starts to decay exponentially with $|n|$ for $|n| \geq N_0$. In order to see if we could obtain more accurate results we also tested the damping functions

$$D(n) = \begin{cases} 1 & \text{for } |n| \leq N_0 \\ 0 & \text{otherwise,} \end{cases}, \tag{44a}$$

$$D(n) = \frac{1}{8}\left(5 + 4\cos\left(\frac{\pi|n|}{N}\right) - \cos\left(\frac{2\pi|n|}{N}\right)\right), \tag{44b}$$

$$D(n) = \begin{cases} 1 & \text{for } |n| \leq N_0 \\ \exp\left(-\alpha\left(\frac{n-N_0}{N-N_0}\right)^p\right) & \text{otherwise,} \end{cases} \tag{44c}$$

with $\alpha = 36$ and $p = 12$ in (44c). We found that these damping functions did not provide results which were significantly different from the results obtained when using the damping function (43).

Accepted Article

This article is protected by copyright. All rights reserved.

Accepted Article

	Percentage of limiting steepness				
kh	10%	50%	90%	99%	100%
0.3	0.0036	0.0182	0.0327	0.0360	0.0364
1	0.0099	0.0470	0.0894	0.0984	0.0994
2π	0.0140	0.0700	0.1260	0.1387	0.1401

TABLE 1 The steepness H/L used for the convergence curves of $u_s(x, t_0)$ and $w_s(x, t_0)$ as a function of kh and the percentage of the limiting steepness given by (45).

4 | NUMERICAL RESULTS

This section presents the results that we have obtained using the above method to solve a number of nonlinear wave problems. A thorough evaluation of the method's accuracy, stability and efficiency properties is provided, covering shallow, intermediate and deep water cases for which the wave steepness is varied from small values up to those near the maximum physically allowed. In an attempt to establish the limits of the method we not only report the cases for which the method is successful, but also the cases for which the method starts to fail.

Before we begin, a few words about initialization of the nonlinear waves are appropriate. Several of the test problems that we present are based on monochromatic nonlinear waves and we initialize these using the method presented by Clamond & Dutykh[14]. To our knowledge this is the most accurate method to compute monochromatic nonlinear waves. In fact, when executed in double precision, the method is capable of computing $\eta(x, t)$, $\Phi_s(x, t)$, $u_s(x, t)$ and $w_s(x, t)$ to double precision. The method computes the surface quantities on a grid which is not equidistant, meaning that the discrete Fourier transform cannot be directly employed to compute the Fourier coefficients. To compute the coefficients we have therefore used the method of Clamond & Dutykh to calculate the surface quantities at a very high resolution and then interpolated these to an equidistant grid before using the discrete Fourier transform.

4.1 | Accuracy of wave kinematics for given surface quantities

The first problem we consider is that of determining the kinematics of a nonlinear wave when the surface quantities $\eta(x, t)$ and $\Phi_s(x, t)$ are specified at an instant in time, t_0. More specifically, we consider monochromatic nonlinear waves and test how accurate our method can compute the surface velocities $u_s(x, t_0)$ and $w_s(x, t_0)$ as well as the velocity profiles $u(x_0, z, t_0)$ and $w(x_0, z, t_0)$ where x_0 is some horizontal location. The former test is of importance to the time integration of the governing equations, as the surface velocities enter these directly, while the latter test is important for applications.

4.1.1 | Computation of the surface velocities

To demonstrate how accurately our method can compute the surface velocities, we have constructed the convergence curves of $u_s(x, t_0)$ and $w_s(x, t_0)$ to the exact results as N and M are increased. We have done so for a shallow water case with $kh = 0.3$, an intermediately deep water case with $kh = 1$ and a deep water case with $kh = 2\pi$. For each of these cases we have set H/L to be 10%, 50%, 90% and 99% of the maximally allowed steepness

$$\frac{H_{\max}}{L} = 0.1401 \tanh\left(0.8863kh\right), \tag{45}$$

as found by Battjes[15]. At this point we want to highligt that the cases where H/L is 99% of the limiting steepness are indeed extreme cases. In fact, as far as we know, no numerical method for the initial value problem for water waves has ever been put to this test for a higher steepness than 90% of the steepness limit. The actual values used for H/L are given in Table 1 as a function of kh and the percentage of the limiting steepness. For all calculations we have used $\epsilon = 10^{-14}$ to ensure that no error comes from the iterative solution of the linear system of equations. We have measured the error of $u_s(x, t_0)$ and $w_s(x, t_0)$ as

$$\text{Error}\left[u_s\right] = \frac{\left\| u_s - u_s^{(T)} \right\|}{\left\| u_s^{(T)} \right\|}, \qquad \text{Error}\left[w_s\right] = \frac{\left\| w_s - w_s^{(T)} \right\|}{\left\| w_s^{(T)} \right\|}, \tag{46}$$

This article is protected by copyright. All rights reserved.

Accepted Article

where $\|\cdot\|$ is the norm induced by the inner product (6) and $u_s^{(T)}(x,t_0)$ and $w_s^{(T)}(x,t_0)$ are the target velocities initialized from the method of[14]. In the case of $u_s(x,t_0)$ this means that

$$\left\| u_s - u_s^{(T)} \right\|^2 = \sum_{n=-N}^{N} \left| \widehat{u}_n - \widehat{u}_n^{(T)} \right|^2 + \sum_{N<|n|\leq N_T} \left| \widehat{u}_n^{(T)} \right|^2, \tag{47a}$$

$$\left\| u_s^{(T)} \right\| = \sum_{n=-N_T}^{N_T} \left| \widehat{u}_n^{(T)} \right|^2, \tag{47b}$$

where $\widehat{u}_n^{(T)}$ denotes the nth Fourier coefficient of the target velocity $u_s^{(T)}(x,t_0)$ and N_T is the number of coefficients needed such that the target velocities are correct to double precision. The error of $w_s(x,t_0)$ is computed in a similar way. To get the target velocities correct to double precision we have used $N_T = 2000$ for the steepest shallow water case while we have taken $N_T = 500$ for all other cases.

The convergence of $u_s(x,t)$ and $w_s(x,t)$ to the target result for the different values of kh and H/L is shown in Figure 3 and Figure 4. The figures illustrate that our method is capable of achieving highly accurate results for all values of H/L and kh, with a low number of degrees of freedom, as the error decreases exponentially with N and M in all cases. It is clear that the convergence rate becomes smaller as H/L increases but this is no surpise since the decay of the expansion coefficients of $\eta(x,t)$ goes from being geometric to algebraic as the limiting steepness is approached. Furthermore, the figures show that once the smallest possible error has been reached, the error stays constant with N and M. We emphasize this point, because it illustrates the consistency of the method and because it has been demonstrated that other spectral methods (see e.g. Dommermuth & Yue[17] and Bateman et al.[18]) give less accurate results as the vertical resolution is increased when H/L is sufficiently large. The figures also clearly illustrate the fact that the accuracy tends to be limited by N in shallow water and by M in deep water. Even though this is quite intuitive it is worth keeping in mind when doing physical simulations.

To give more substance to the claim that our method is capable of obtaining high accuracy with a low number of degrees of freedom, we end this section by showing the convergence of $w_s(x,t)$ to the target result for the same case as Yates & Benoit[8] used for their Figure 2. In this case $kh = 2\pi$ and $H/L = 0.1$ and for the comparison it should be kept in mind that our parameter for the horizontal resolution, N, is related to their parameter N_x through the relation $N_x = 2N+1$, and that our parameter for the vertical resolution, M, equals their parameter N. The result is shown in Figure 5 which has been produced using $\epsilon = 10^{-14}$ and $N_T = 500$. When comparing the results shown in the figure to those of Yates & Benoit it is clear that our method is in general several orders of magnitude more accurate than their method when using the same number of degrees of freedom. Comparing our results for $N = 47$ (meaning that our total number of degrees of freedom in the horizontal dimension is $2N + 1 = 95$) to their results for $N_x = 96$ shows that our method is roughly four orders of magnitude more accurate for $M = 20$ and roughly eight orders of magnitude more accurate for $M = 30$.

4.1.2 | Computation of velocity profiles

In order to show that our method can produce accurate velocity profiles we compare our results to those obtainable by the streamfunction method of Rienecker & Fenton[16]. We do, however, use the method of Clamond & Dutykh for initializing $\eta(x,t)$ and $\Phi_s(x,t)$. Here we only report the results for the vertical profile of the horizontal velocity below the wave crest, but we have obtained equally good results for both the horizontal and vertical velocity profiles at other locations. To cover a wide range of values for both kh and H/L we have used the same cases as in the previous section. The velocity profiles are shown in Figure 6, and for all cases our results are in good agreement with the results of the streamfunction method. To produce these results we have chosen the computational parameters N, M and ϵ as well as the number of points for the streamfunction method such that the results are converged on the scale of the figure.

4.2 | Stability and accuracy of time integration of nonlinear monochromatic waves in deep water

For monochromatic nonlinear waves in deep water the temporal stability of our method is highly dependent on the wave steepness. We have found that for fine enough spatial and temporal resolution there exists a threshold steepness such that for all practical purposes the time integration is stable when the steepness is below the threshold and unstable when the steepness is above the threshold. To two significant digits this threshold value is $H/L = 0.059$ and the stability on both sides of this value is illustrated in Figure 7 and Figure 8 which have been made using the parameters $N = 32$, $M = 24$, $\epsilon = 10^{-14}$, $kh = 2\pi$ and $\Delta t = T/100$ where T is the wave period. The former figure clearly shows that the method is capable of stable and accurate

This article is protected by copyright. All rights reserved.

Accepted Article

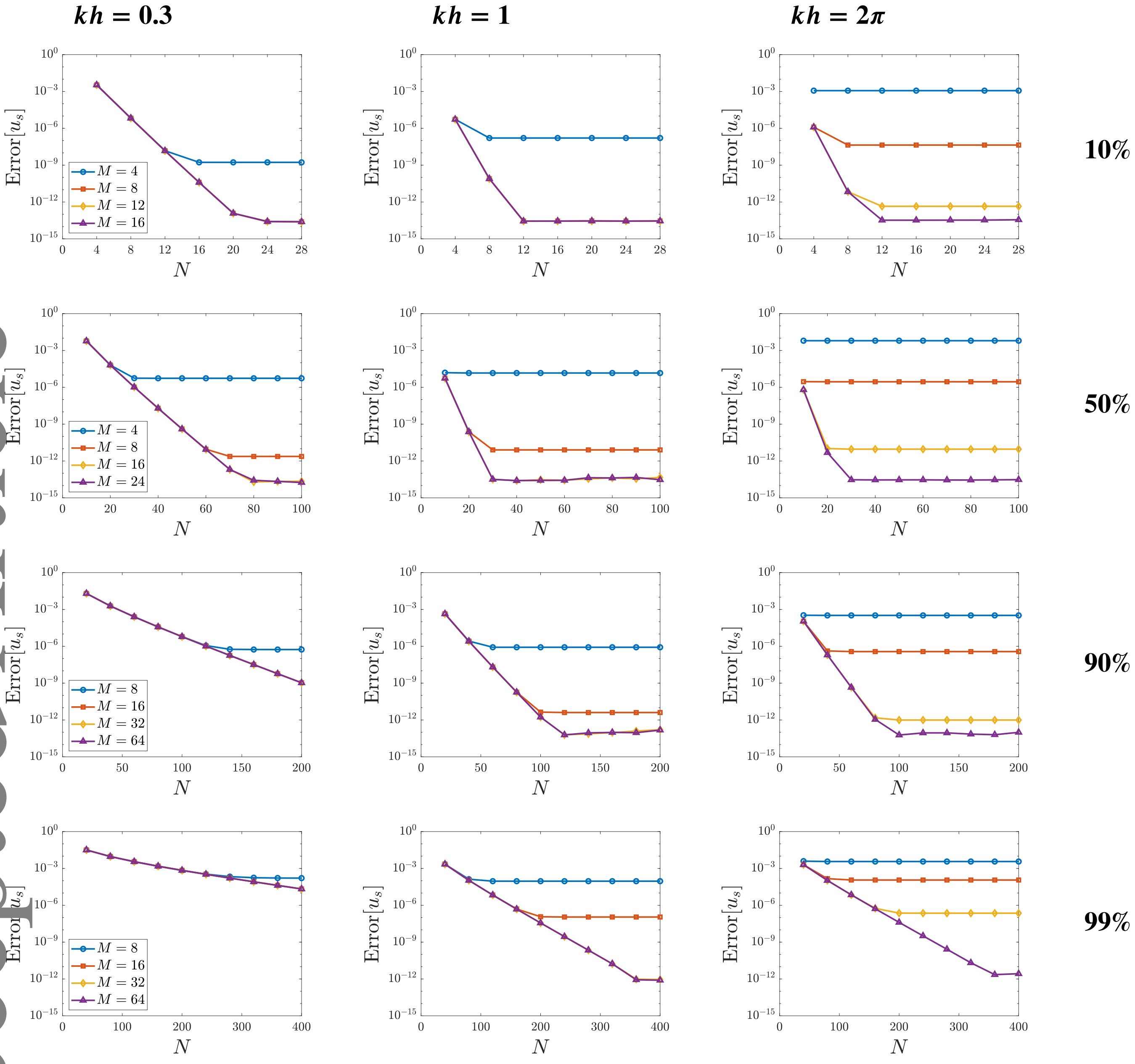

FIGURE 3 The convergence of $u_s(x,t)$ to the exact result as a function of N for different values of M. Starting from the top, the rows correspond to 10%, 50%, 90% and 99% of the maximal steepness, respectively, while the columns from left to right correspond to the shallow, intermediate and deep water cases, respectively. The legends apply row by row.

time integration of a wave with $H/L = 0.058$, while the latter figure clearly shows that the time integration of a wave with $H/L = 0.059$ blows up after roughly 32.5 periods. Moreover, we have found that the time at which the time integration blows up decreases with the steepness. Defining t_B as the first time the maximal free surface elevation is twice as large as the maximal elevation of the initial wave, this point is illustrated in Figure 9. The figure shows t_B/T as a function of H/L for the parameters $N = 32$, $M = 24$, $\epsilon = 10^{-14}$, $kh = 2\pi$ and $T/\Delta t = 100$ and it is clear that t_B/T decreases with H/L.

The presence of the instability in the method is consistent with the seemingly ubiquitous need to apply damping in one form or another with fully nonlinear potential flow wave models almost regardless of the chosen discretization. To underline this point, we mention here that both finite difference methods[7], boundary element methods[19], spectral methods[18 20] and spectral element methods[21] have all been reported unstable if no damping was applied. At this point we want to emphasize that we (nor seemingly anybody else) have not been able to identify the exact cause of the instability. What we do know, however, is that it

This article is protected by copyright. All rights reserved.

Accepted Article

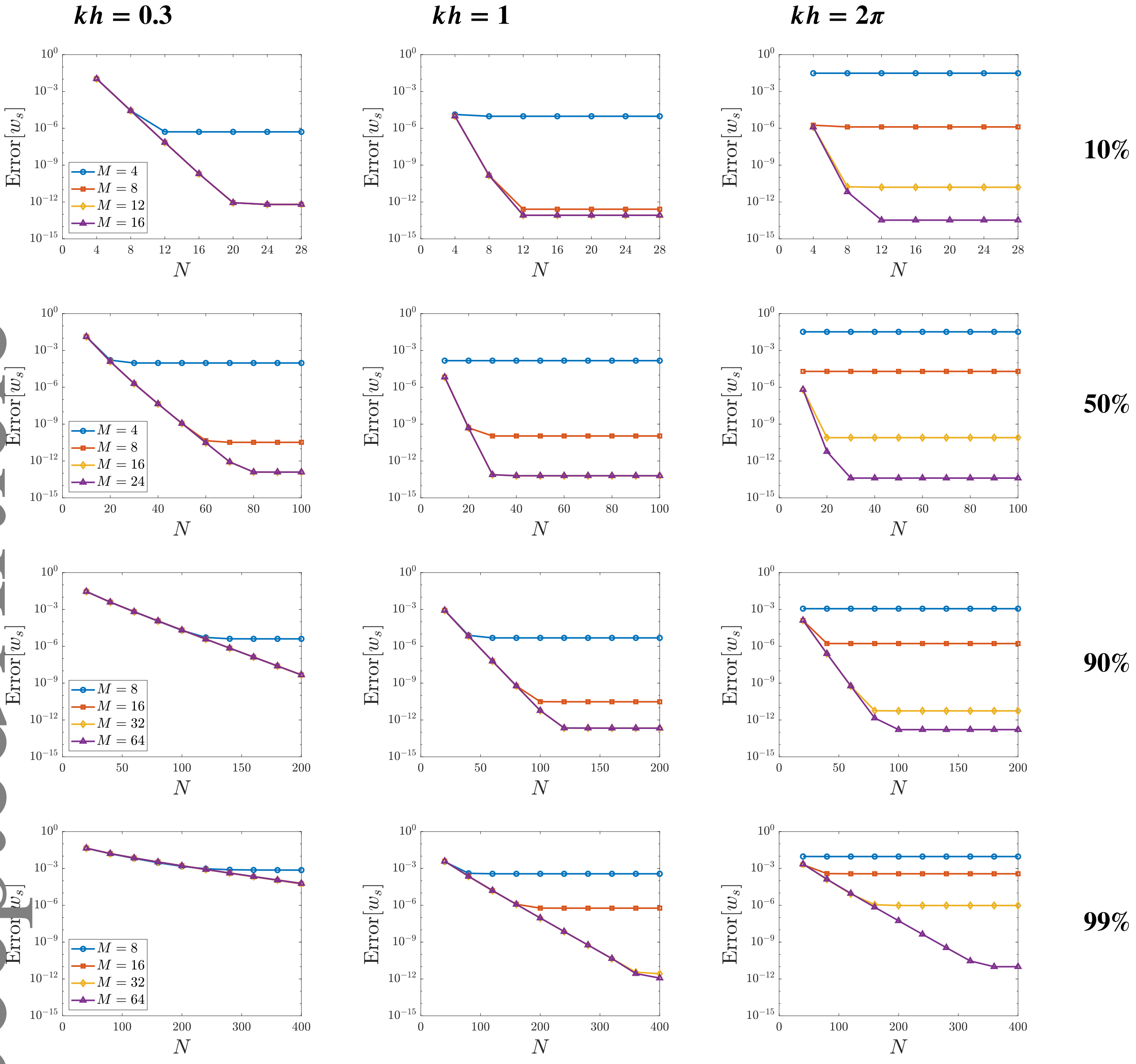

FIGURE 4 The convergence of $w_s(x, t)$ to the exact result as a function of N for different values of M. Starting from the top, the rows correspond to 10%, 50%, 90% and 99% of the maximal steepness, respectively, while the columns from left to right correspond to the shallow, intermediate and deep water cases, respectively. The legends apply row by row.

is not due to aliasing errors as the Galerkin method used to discretize the nonlinear free surface equations is inherently free of the aliasing phenomena.

4.2.1 | Local nonlinear stability analysis

In an attempt to get a better understanding of the instability of monochromatic nonlinear waves in deep water we have performed the following analysis. The analysis starts by noting that the discretized time dependent surface boundary conditions (10) and (11) together form an autonomous system of coupled first order differential equations on the form

$$\frac{\mathrm{d}y}{\mathrm{d}t} = f(y), \tag{48}$$

This article is protected by copyright. All rights reserved.

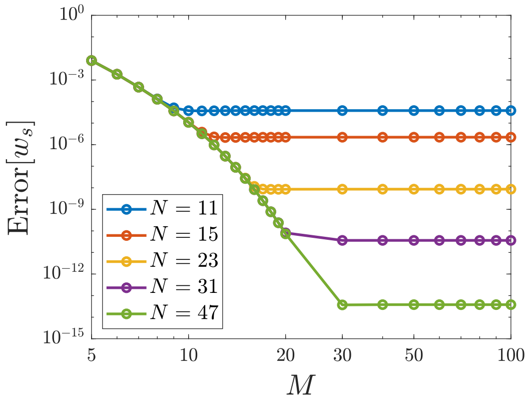

FIGURE 5 The convergence of $w_s(x,t)$ to the exact result as a function of M for different values of N. Here the relative water depth is $kh = 2\pi$, the wave steepness is $H/L = 0.1$ and $\epsilon = 10^{-14}$ has been used. The values of N correspond to $N_x = 23$, 31, 47, 63 and 95 where N_x is the horizontal resolution of Yates & Benoit[8].

Accepted Article

where $y = [\hat{\eta}_{-N}, ..., \hat{\eta}_N, \hat{\Phi}_{-N}, ..., \hat{\Phi}_N]^T$. Linearizing this system of equations around the initial nonlinear wave yields the equation

$$\frac{\mathrm{d}y}{\mathrm{d}t} \approx f(\tilde{y}) + J(\tilde{y})(y - \tilde{y}), \tag{49}$$

where $\tilde{y}$ is the vector containing the expansion coefficients of the initial nonlinear wave and $J(\tilde{y})$ is the Jacobian matrix defined by

$$J(\tilde{y}) = \left[\frac{\partial f}{\partial \hat{\eta}_{-N}}(\tilde{y}) \ \cdots \ \frac{\partial f}{\partial \hat{\eta}_{N}}(\tilde{y}) \ \frac{\partial f}{\partial \hat{\Phi}_{-N}}(\tilde{y}) \ \cdots \ \frac{\partial f}{\partial \hat{\Phi}_{N}}(\tilde{y}) \right]. \tag{50}$$

For a locally linear system of equations like (49) the stability can be analyzed by considering the eigenvalues of $J(\tilde{y})$. If just a single eigenvalue has a positive real part then for all practical purposes the exact system (48) is very likely to be unstable no matter how small we choose the time step. To support this claim we have computed the eigenvalues of $J(\tilde{y})$ as a function of H/L in steps of 0.001 for the parameters $N = 32$, $M = 24$, $\epsilon = 10^{-14}$ and $kh = 2\pi$ and found that the analysis predicts waves with $H/L < 0.059$ to be stable and waves with $H/L \geq 0.059$ to be unstable which is in line with the results of the previous section. The computation of the Jacobian matrix was done using an 8th order centered finite difference scheme, as lower order methods did not lead to converged results. An excerpt of the computation is illustrated in Figure 10 which shows the eigenvalues of $J(\tilde{y})$ for $H/L = 0.058$, 0.059 and 0.13. It is clear from the figure that the analysis predicts the case with $H/L = 0.058$ to be stable and the other cases to be unstable. Moreover, the figure predicts the wave with $H/L = 0.13$ to blow up earlier than the wave with $H/L = 0.059$ due to the fact that the real parts of the eigenvalues are larger for $H/L = 0.13$ than for $H/L = 0.059$. This prediction is again in agreement with the earlier results for the blow up time.

When performing this analysis it is important to keep in mind that each set of parameters has to be checked individually. To this end, a reasonable question is whether the instability of the time integration could be removed by using a different set of parameters. To answer this question we have computed the eigenvalues of the Jacobian matrix for the case $H/L = 0.13$ using $N = 16$, 32 and 64 and $M = 8$, 16 and 24 and found that the instability could *not* be overcome by using another spatial resolution. For this reason we conclude that artificial damping is the only remedy against the instability at our disposal.

Before proceeding, we here duly note that the analysis is based on the same basic idea as presented by Fuhrman et al.[22]. Our method, however, improves that of Fuhrman et al. in the sense that the Jacobian matrix used in our method is the (numerically) exact Jacobian matrix in contrast to the Jacobian matrix based on approximate analytical derivatives used by Fuhrman et al.

4.2.2 | Using artificial damping to stabilize the time integration

In order to stabilize the time integration we have employed the damping strategy outlined in Section 3.5. Using this strategy enables accurate long time propagation of even very steep waves, as it is only very high wavenumbers which are damped. To illustrate this, we have integrated monochromatic nonlinear waves of varying steepness in time for exactly 50 periods with $kh = 2\pi$ using the parameters $N = 32$, $M = 24$, $\epsilon = 10^{-14}$, $\Delta t = T/150$ and $N_0 = 20$ and compared the end state with the

This article is protected by copyright. All rights reserved.

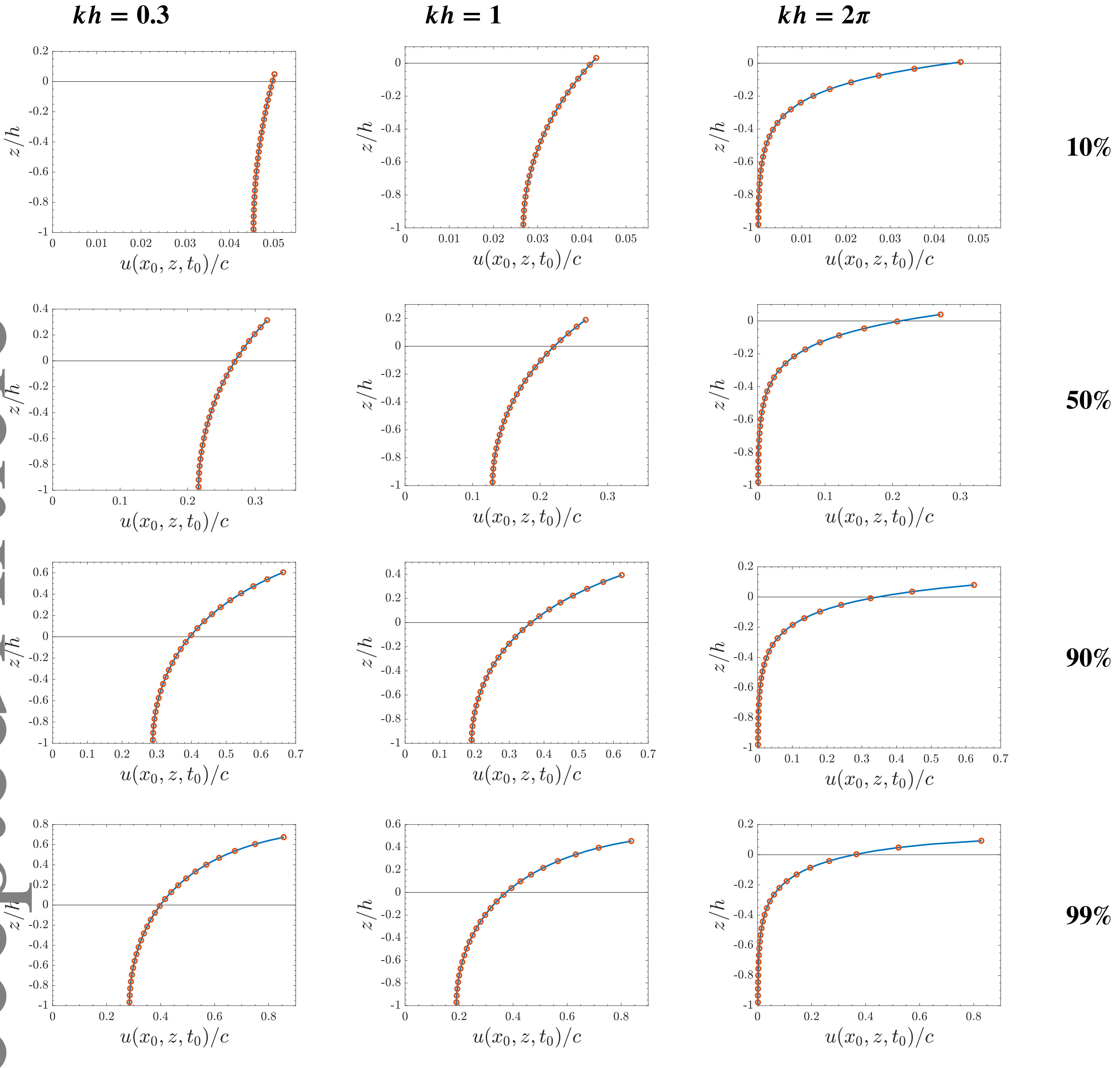

FIGURE 6 The horizontal velocity profile below the wave crest. The blue line is obtained using the streamfunction method while the red circles are obtained using our method. Starting from the top, the rows correspond to 10%, 50%, 90% and 99% of the maximal steepness, respectively, while the columns from left to right correspond to the shallow, intermediate and deep water cases, respectively.

Accepted Article

initial condition. As the accuracy of the time integration decreases with H/L we here only show the result of this calculation for $H/L \geq 0.125$. Figure 11 shows that the propagated wave is in very good agreement with the initial wave for $H/L = 0.125$. For $H/L = 0.13$ a closer look is needed to reveal that the propagated wave is not identical to the initial wave while for $H/L = 0.135$ a clear difference between the propagated wave and the initial wave is evident. For $H/L = 0.14$ the difference between the propagated wave and the initial wave is, not surprisingly, even larger.

To quantify the accuracy of the time integration, we have computed the relative error of the wave celerity after 50 periods of propagation as this quantity does not depend on the time duration of the simulation. In this work we define the relative error of the celerity such that it is positive when the exact wave moves faster than the computed wave. We have computed the relative error

This article is protected by copyright. All rights reserved.

Accepted Article

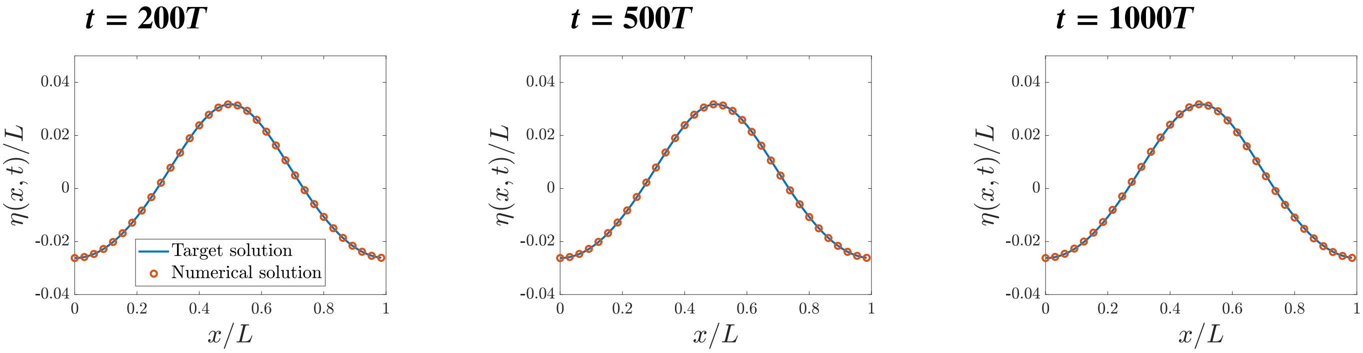

FIGURE 7 The free surface elevation of a monochromatic nonlinear wave with $H/L = 0.058$ at different instances in time for the parameters $N = 32$, $M = 24$, $\epsilon = 10^{-14}$, $kh = 2\pi$ and $\Delta t = T/100$. The legend applies to the entire row.

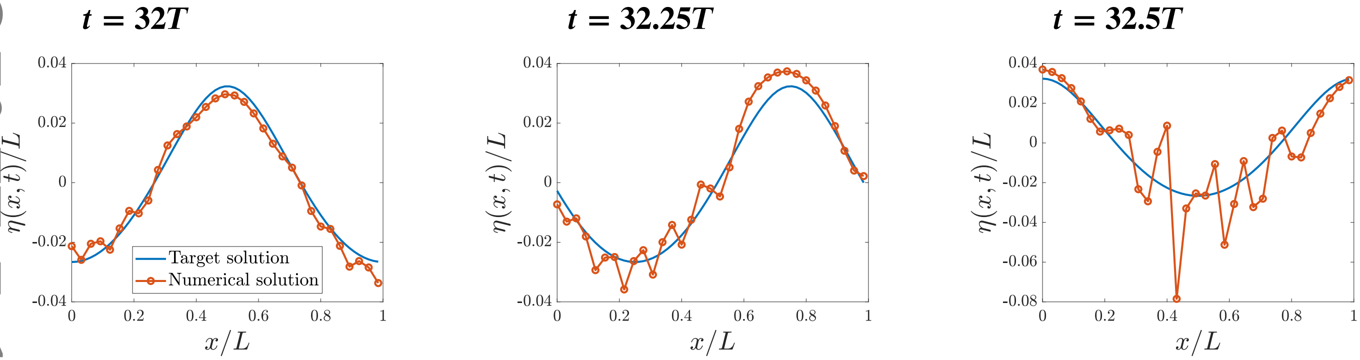

FIGURE 8 The free surface elevation of a monochromatic nonlinear wave with $H/L = 0.059$ at different instances in time just before the numerical blowup for the parameters $N = 32$, $M = 24$, $\epsilon = 10^{-14}$, $kh = 2\pi$ and $\Delta t = T/100$. The legend applies to the entire row.

of the celerity as a function of H/L and $T/\Delta t$ again using the parameters $N = 32$, $M = 24$, $\epsilon = 10^{-14}$, $N_0 = 20$ and $kh = 2\pi$. The results are given in Tables 2 and 3. The results indicate that using 100 time steps per wave period is enough to obtain highly accurate results for $H/L \leq 0.13$. For $H/L = 0.13$ it would, for example, take roughly 2300 periods of propagation before the position of the computed wave crest would differ by half a wavelength when compared to the location of the exact crest. It must be noted, however, that the results also show that the accuracy does not become much better for more than 100 time steps per period indicating that the largest source of error is no longer the time integration but rather the spatial resolution of the problem. For this reason we have tried using a finer spatial resolution and different values of the damping parameter N_0 in an attempt to make the error of the celerity smaller. Unfortunately, this did not lead to more accurate results because the simulations quickly became unstable if N_0 was much larger than 20. For all practical purposes this effectively implies that we can accurately time integrate waves which are well described by $N \approx 20$.

4.3 | Reflection of a solitary wave from a wall

To assess the method's ability to handle nonlinear wave phenomena in relatively shallow water, we have employed it to simulate the reflection of solitary waves by a vertical wall. Denoting the dimensionless initial amplitude of the solitary wave by a/h, the wave will be reflected elastically off the wall in the limit of small a/h. Furthermore, the maximum dimensionless surface elevation during the reflection is approximately $2a/h$ at that limit. For large values of a/h the event is more complicated, as

This article is protected by copyright. All rights reserved.

Accepted Article

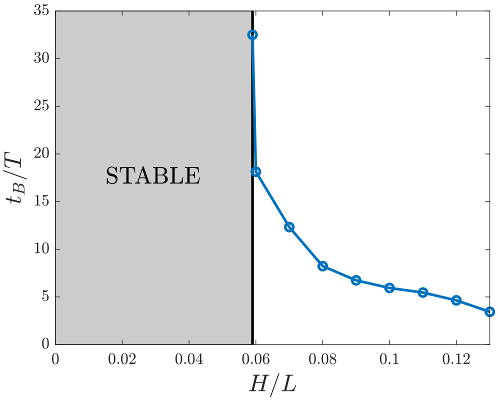

FIGURE 9 t_B/T as a function of H/L for the parameters $N = 32$, $M = 24$, $\epsilon = 10^{-14}$, $kh = 2\pi$ and $\Delta t = T/100$. The vertical black line is located at $H/L = 0.059$.

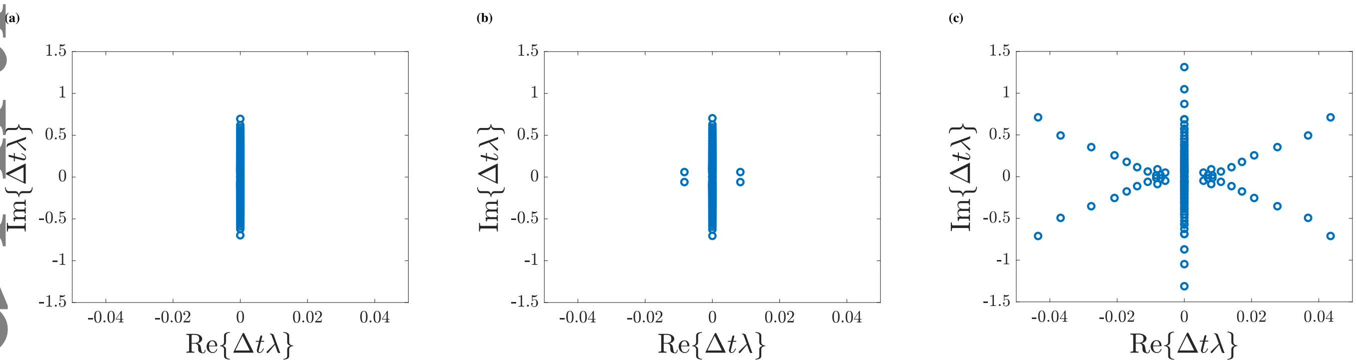

FIGURE 10 The eigenvalues of the Jacobian matrix for (a) $H/L = 0.058$, (b) $H/L = 0.059$ and (c) $H/L = 0.13$ scaled with the time step $\Delta t = T/100$ for the parameters $N = 32$, $M = 24$, $\epsilon = 10^{-14}$ and $kh = 2\pi$.

	H/L						
$T/\Delta t$	0.03	0.04	0.05	0.06	0.07	0.08	0.09
50	27.5×10^{-7}	34.9×10^{-7}	46.1×10^{-7}	66.2×10^{-7}	10.2×10^{-6}	16.9×10^{-6}	30.3×10^{-6}
60	13.5×10^{-7}	17.1×10^{-7}	22.3×10^{-7}	31.6×10^{-7}	4.77×10^{-6}	7.68×10^{-6}	13.4×10^{-6}
70	7.63×10^{-7}	9.84×10^{-7}	12.9×10^{-7}	18.2×10^{-7}	2.68×10^{-6}	4.20×10^{-6}	7.11×10^{-6}
80	4.92×10^{-7}	6.55×10^{-7}	8.78×10^{-7}	12.3×10^{-7}	1.79×10^{-6}	2.71×10^{-6}	4.40×10^{-6}
90	3.55×10^{-7}	4.95×10^{-7}	6.82×10^{-7}	9.61×10^{-7}	1.37×10^{-6}	2.02×10^{-6}	3.14×10^{-6}
100	2.84×10^{-7}	4.15×10^{-7}	5.89×10^{-7}	8.36×10^{-7}	1.18×10^{-6}	1.70×10^{-6}	2.53×10^{-6}
110	2.46×10^{-7}	3.76×10^{-7}	5.49×10^{-7}	7.85×10^{-7}	1.11×10^{-6}	1.56×10^{-6}	2.24×10^{-6}
120	2.27×10^{-7}	3.61×10^{-7}	5.38×10^{-7}	7.74×10^{-7}	1.07×10^{-6}	1.51×10^{-6}	2.12×10^{-6}
130	2.18×10^{-7}	3.59×10^{-7}	5.43×10^{-7}	7.86×10^{-7}	1.10×10^{-6}	1.52×10^{-6}	2.10×10^{-6}
140	2.17×10^{-7}	3.64×10^{-7}	5.58×10^{-7}	8.12×10^{-7}	1.14×10^{-6}	1.56×10^{-6}	2.13×10^{-6}
150	2.22×10^{-7}	3.75×10^{-7}	5.80×10^{-7}	8.56×10^{-7}	1.18×10^{-6}	1.61×10^{-6}	2.19×10^{-6}

TABLE 2 The relative error of the celerity as a function of steepness and the number of time steps per period when using the parameters $N = 32$, $M = 24$, $\epsilon = 10^{-14}$, $N_0 = 20$ and $kh = 2\pi$.

the solitary wave is reflected inellastically, because it loses energy to a dispersive tail. Moreover, the maximum dimensionless surface elevation can become significantly larger than $2a/h$ during the reflection, as a thin jet is formed close to the wall, and the

This article is protected by copyright. All rights reserved.

Accepted Article

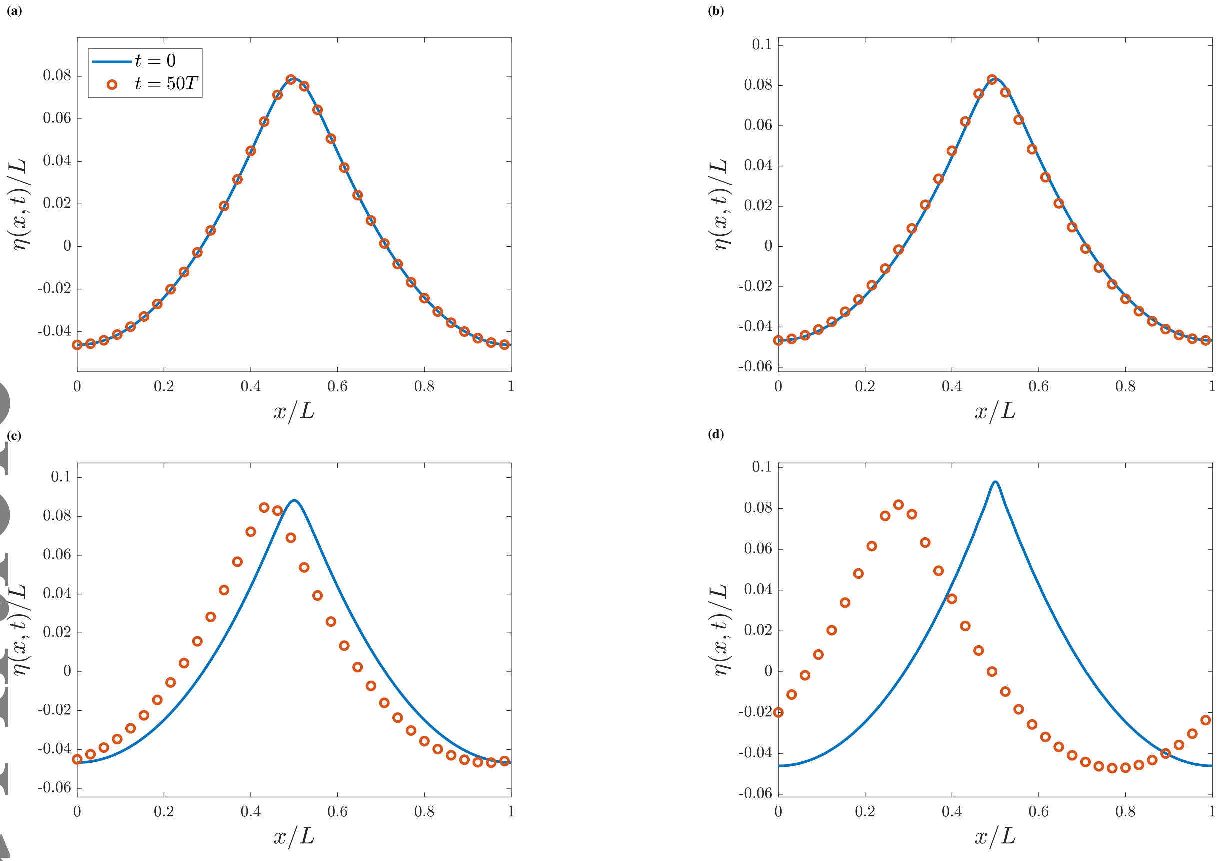

FIGURE 11 A comparison of $\eta(x, 0)$ and $\eta(x, 50T)$ for (a) $H/L = 0.125$, (b) $H/L = 0.13$, (c) $H/L = 0.135$ and (d) $H/L = 0.14$ using the parameters $N = 32$, $M = 24$, $\epsilon = 10^{-14}$, $N_0 = 20$, $kh = 2\pi$ and $\Delta t = T/150$ for all four cases.

	H/L						
$T/\Delta t$	0.10	0.11	0.12	0.125	0.13	0.135	0.14
50	62.1×10^{-6}	15.0×10^{-5}	47.7×10^{-5}	10.0×10^{-4}	23.2×10^{-4}	5.03×10^{-3}	7.79×10^{-3}
60	26.9×10^{-6}	6.46×10^{-5}	21.0×10^{-5}	4.65×10^{-4}	12.3×10^{-4}	3.39×10^{-3}	6.36×10^{-3}
70	13.7×10^{-6}	3.21×10^{-5}	10.4×10^{-5}	2.36×10^{-4}	6.98×10^{-4}	2.41×10^{-3}	5.53×10^{-3}
80	8.09×10^{-6}	1.81×10^{-5}	5.74×10^{-5}	1.32×10^{-4}	4.30×10^{-4}	1.84×10^{-3}	5.01×10^{-3}
90	5.44×10^{-6}	1.14×10^{-5}	3.48×10^{-5}	0.81×10^{-4}	2.91×10^{-4}	1.51×10^{-3}	4.69×10^{-3}
100	4.13×10^{-6}	0.81×10^{-5}	2.31×10^{-5}	0.54×10^{-4}	2.17×10^{-4}	1.32×10^{-3}	4.56×10^{-3}
110	3.46×10^{-6}	0.62×10^{-5}	1.67×10^{-5}	0.40×10^{-4}	1.77×10^{-4}	1.22×10^{-3}	4.50×10^{-3}
120	3.14×10^{-6}	0.53×10^{-5}	1.32×10^{-5}	0.32×10^{-4}	1.54×10^{-4}	1.17×10^{-3}	4.47×10^{-3}
130	3.01×10^{-6}	0.48×10^{-5}	1.12×10^{-5}	0.27×10^{-4}	1.42×10^{-4}	1.15×10^{-3}	4.48×10^{-3}
140	2.98×10^{-6}	0.46×10^{-5}	1.00×10^{-5}	0.24×10^{-4}	1.36×10^{-4}	1.14×10^{-3}	4.50×10^{-3}
150	3.02×10^{-6}	0.45×10^{-5}	0.94×10^{-5}	0.23×10^{-4}	1.34×10^{-4}	1.13×10^{-3}	4.46×10^{-3}

TABLE 3 The relative error of the celerity as a function of steepness and the number of time steps per period when using the parameters $N = 32$, $M = 24$, $\epsilon = 10^{-14}$, $N_0 = 20$ and $kh = 2\pi$.

reflection event thus constitutes a demanding test case including several nonlinear features. Evaluating the quality of the results is, in contrast, a relatively simple task due to the many careful numerical studies of such events over the years. Here we mention Cooker et al.[23] and Chambarel et al.[24], who both solved the fully nonlinear potential flow problem using boundary integral

This article is protected by copyright. All rights reserved.

Accepted Article

methods, Papoutsellis et al.[25] who solved the fully nonlinear potential flow problem using a finite difference discretization of a Hamiltonian coupled-mode theory and Madsen et al.[26] and Engsig-Karup et al.[27] who discretized the same highly accurate Boussinesq formulation using a finite difference method and a discontinuous Galerkin finite element method, respectively. As the results of the former three references agree for all values of a/h despite the fact that they use different methods, we believe that their results are all highly accurate.

Now, to simulate the reflection event we exploit the well-known fact that the event is indistinguishable from the collision of two identical solitary waves moving in opposite directions, provided that their initial positions are such that any overlap of the waves is negligible. In order to initialize $\eta(x,t)$ and $\Phi_s(x,t)$ for the two solitary waves, we have used the method of Dutykh & Clamond[28] which is capable of computing solitary waves to double precision when executed in double precision. As the method takes as input the Froude number, $\mathscr{F} = c/\sqrt{gh}$, where c is the celerity of the solitary wave, we have employed the method in combination with a simple bisection algorithm in order to compute waves of a given amplitude. Knowing the Froude number of the wave, we base the choice of the time step on the CFL-like criteria

$$\Delta t = C_r \frac{L/(2N+1)}{\mathscr{F}\sqrt{gh}}, \tag{51}$$

where the constant C_r is a dimensionless number. For $C_r \leq 1$ we have found that the size of C_r is mainly important for the accuracy of the time integration and that it has little influence on the stability of the calculation. In fact, we could simulate the reflection event without any artificial damping for $a/h \leq 0.6$, but for steeper waves the simulations would blow up if no damping was added. We have not been able to identify the cause of the instability, but in the light of the results reported by Chambarel et al. it is interesting to note that the instability sets in for $a/h > 0.6$. For these amplitudes they found that a residual jet is formed during the downrush of the wave from the wall, and according to their Figure 9, this residual jet is in general not a single-valued function. As our method can only deal with surface elevations which are single-valued it is probably not coincidental that it starts to break down at exactly this point. We note that this is in line with the results reported by Engsig-Karup et al., who had to increase the amount of damping necessary to stabilize the simulation significantly when reaching $a/h = 0.6$.

During the reflection event we monitor the instantaneous depth integrated force on the wall as function of time as well as the crest elevation at the times of attachment, detachment and maximum runup which we denote by η_a, η_d and η_0, respectively. Here the attachment time is defined as the first time at which the crest is located at the wall while the detachment time is defined as the last time at which the crest is located at the wall. For the parameters $N = 300$, $M = 20$, $\epsilon = 10^{-14}$, $C_r = 0.1$, and $L/h = 50$ our results for these quantities without using any artificial damping are shown in Figure 12 for different values of a/h together with the results of Cooker et al. As mentioned above, the simulations are unstable for $a/h > 0.6$ and we have stopped them just prior to the blowup. In that regard it is interesting that Cooker et al. were able to compute η_0 for $a/h = 0.65, 0.7$ and continue the force curve for $a/h = 0.7$ up to $(t-t_0)/\sqrt{h/g} > 0$. We have, however, left out those results in order to make the figures appear more clear.

It is evident from Figure 12 that our results agree well with those of Cooker et al. There is a slight discrepancy between the results for the dimensionless force for $a/h = 0.6$ but this is, however, not unexpected as that case is right on the edge of being unstable. In that connection an interesting question is whether more accurate results can be obtained using artificial damping. Not surprisingly we have found the answer to this question to be "no", as the effect of the damping in this case seems to be to prevent an actual physical phenomena, namely that the free surface becomes multivalued.

4.4 | Nonlinear wave focusing

As a final validation of our method we use it to simulate nonlinear focusing events in deep water. Such events have been studied experimentally by Johannesen & Swan[29] and in this section we aim at simulating their experiments. Since our method at the moment is only implemented in one horizontal dimension we will restrict ourselves to their unidirectional cases.

Starting our simulation at time $t = -t_0$, we take the initial condition for the simulation to be such that linear theory predicts the focusing event to take place at $t = 0$ at the location $x = L/2$. More specifically, we take

$$\eta(x,-t_0) = \sum_{n=1}^{N_{\mathrm{freq}}} a_n \cos\left(\kappa_n\left(x - \frac{L}{2}\right) + \omega_n t_0\right), \tag{52a}$$

$$\Phi_s(x,-t_0) = \sum_{n=1}^{N_{\mathrm{freq}}} \frac{g a_n}{\omega_n} \sin\left(\kappa_n\left(x - \frac{L}{2}\right) + \omega_n t_0\right), \tag{52b}$$

This article is protected by copyright. All rights reserved.

Accepted Article

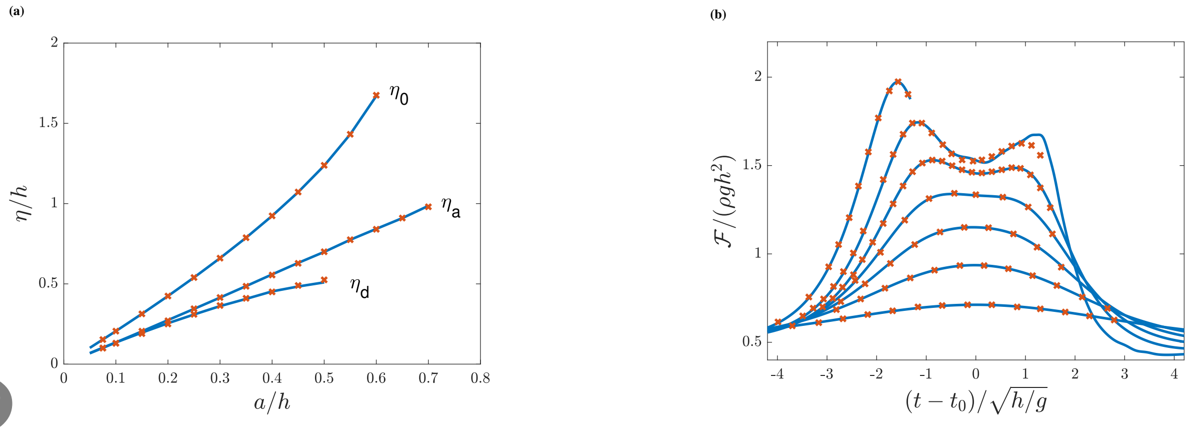

FIGURE 12 (a) The free surface amplitude at the instant of attachment, η_a, detachment, η_d, and maximum runup, η_0, as a function of the initial amplitude a/h. (b) The dimensionless force excerted on the wall close to the runup as a function of time for $a/h = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7$ where t_0 denotes the instant of maximal runup. In both figures the blue line shows our results while the red crosses show the results of Cooker et. al.[23].

and due to nonlinear effects the waves will in fact focus at the time $t_f > 0$. In (52) the angular frequencies ω_n and the wavenumbers κ_n are tied together by the linear dispersion relation $\omega_n^2 = g\kappa_n \tanh(\kappa_n h)$ and the expansion coefficients a_n are determined from the two relations

$$a_n \propto \omega_n^{-2}, \qquad A = \sum_{n=1}^{N_{\text{freq}}} a_n, \tag{53}$$

where A is the linear amplitude sum at the focus. For this to make sense, one of course has to specify ω_n and A. In this work we consider two cases which we will refer to as B52ud and D61ud, respectively, corresponding to the broad and narrow banded unidirectional cases of Johannessen & Swan. The parameters defining these cases are given in Table 4 where the numbers ω_{min} and ω_{max} determine ω_n as we take

$$\omega_n = \omega_{\text{min}} + (\omega_{\text{max}} - \omega_{\text{min}})\frac{n-1}{N_{\text{freq}}-1}. \tag{54}$$

Regarding the values in the table, we want to draw attention to the fact that we have taken $A = 57.5$ mm for the B52ud case which differs from the value $A = 52$ mm reported by Johannessen & Swan by roughly 11%. Using their value we were not able to make the simulations agree with the experimental results and therefore changed A until a good agreement was found. While this way of making a simulation agree with experimental results is questionable at best, we note that Bateman et al.[18] and Fuhrman & Madsen[30] did similarly for the B52ud case. In fact Fuhrman & Madsen note that they used $A = 59.0$ mm and that Bateman et al. used $A = 57.2$ mm for that particular case. It is clear that if A had not been changed, the comparison of velocity profiles would not make sense as different crest heights give rise to different velocity profiles. To some extend the change of A can thus be justified.

Now, using the parameters $N = 250$, $M = 20$, $\epsilon = 10^{-6}$, $\Delta t = 0.02$ s, $t_0 = 8$ s, $h = 1.2$ m and $L = 25.6$ m we have obtained the results shown in Figure 13 without using any artificial damping. For both cases the figure shows our results together with the results of Johannessen & Swan for the surface elevation at the focus point as a function of time as well as the vertical profile of the horizontal velocity at the focus point at the instant in time where the maximal surface elevation occurs. Although a good match is found for both cases, it is particularly uplifting that the results for the D61ud case agree well, since we used the same value for A as reported by Johannessen & Swan for that case. It is clear that some time after the focus event our results for the free surface elevation for the case D61ud start to deviate from the experimental results. We are, however, confident that the discrepancy is not due to the accuracy of our method as it is much more likely to be caused by the fact that our numerical method is spatially periodic in the horizontal direction and therefore neither includes a wave maker nor the reflection and absorption effects present in the experimental wave tank. In order to include the effect of a wave maker one could use the finite element model developed by Gidel et al.[31] (see their Section entitled "Dealing with moving boundaries"), and we note that the wave

This article is protected by copyright. All rights reserved.

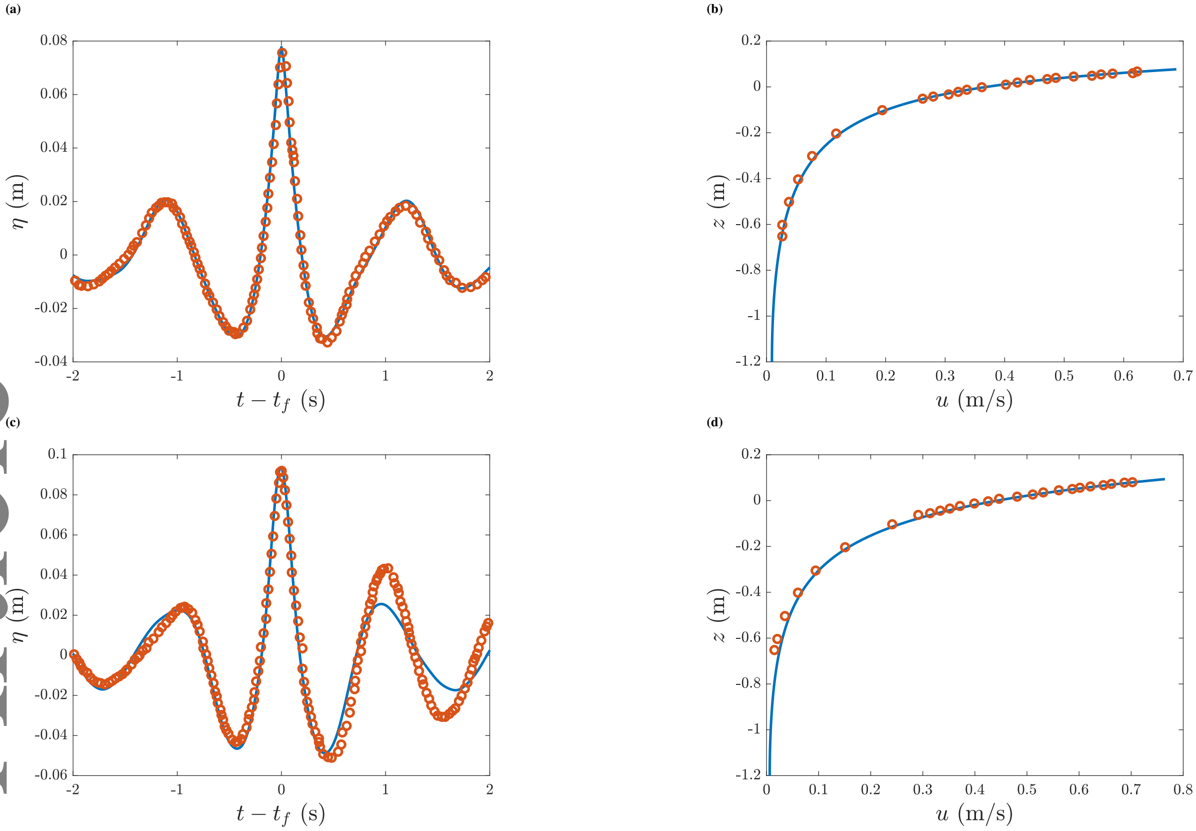

FIGURE 13 Left column: The free surface elevation at the focus point as a function of time relative to the focusing time for (a) case B52ud and (c) case D61ud. Right column: The horizontal velocity profile below the crest at the focus for (b) case B52ud and (d) case D61ud. The full line shows the result of our calculation while the circles are the measured values of Johannessen & Swan[29].

maker approach was successfully employed by Gidel[32] to accurately simulate focusing wave experiments carried out at the Maritime Research Institute Netherlands.

Case	A [mm]	$\omega_{\min}$ [rad s^{-1}]	$\omega_{\max}$ [rad s^{-1}]	N_{freq}
B52ud	57.5	$92\pi/64$	$212\pi/64$	61
D61ud	61.0	$106\pi/64$	$160\pi/64$	28

TABLE 4 The parameters used for the two cases B52ud and D61ud.

4.5 | Computational cost to solve the Laplace problem

The above test cases illustrate the fact that our method is capable of obtaining very accurate results for a wide variety of nonlinear water wave phenomena. As we have only implemented the method in one horizontal dimension, the computational effort has so far not been prohibitive, which might, however, not be the case when the method is implemented in two horizontal dimensions. For that reason we have investigated the performance of our method when applied to a large problem in one horizontal dimension. More specifically, we have tested how many GMRES iterations and how much CPU time is needed to solve the Laplace problem for different values of H/L and ϵ as a function of N. We have done so for a wave field consisting of many repetitions of the

Accepted Article

This article is protected by copyright. All rights reserved.

same monochromatic nonlinear wave for $kh = 2\pi$, where each wavelength is described using 27 basis functions. For all cases we have used $M = 10$.

In Section 3.2.3 we argued that the Laplace equation could optimally be solved in $O(N \log(N))$ operations and that this situation could only be achieved if the number of iterations needed to solve the equation would not grow with N. The left column of Figure 14 shows that the employed preconditioning strategy does not achieve this, as the number of iterations clearly grows with N for all values of H/L and ϵ. Needless to say, the CPU time needed to solve the equation is not proportional to $N \log(N)$ which is evident from the right column of Figure 14. At this point we note that although a lower CPU time might be obtained using a more efficient implementation of the method, such an implementation will also not exhibit a computational effort of $O(N \log(N))$. For this reason, our future work will include the development of a more efficient preconditioning strategy.

It is interesting to note that Christiansen et al.[10] and Engsig-Karup et al.[7] performed a different test which made them claim that their methods scaled in an optimal way. Instead of considering an increasing number of wavelengths with fixed resolution per wavelength, they considered a fixed number of wavelengths with increasing resolution per wavelength. Inspired by this we have measured the number of GMRES iterations and the CPU time needed to solve the Laplace equation for a single wavelength of a nonlinear monochromatic wave with $kh = 2\pi$ and $M = 10$ as a function of N. The results are shown in Figure 15 and it is clearly seen that the number of iterations stays constant with N meaning that our preconditioning strategy has the potential for optimal scaling when measured in this way. It is, however, also clear, that our current implementation does in fact not scale optimally as the CPU time needed to solve the Laplace equation grows faster than $N \log(N)$.

5 | CONCLUSIONS

A new method to solve the fully nonlinear potential flow problem for water waves in one horizontal dimension based on the σ-transform has been presented. The method discretizes the transformed Laplace equation by a Fourier Galerkin method in the horizontal dimension and a Legendre-Tau method in the vertical dimension and solves the discretized equations using the iterative method GMRES preconditioned with the matrix arising from the linearized problem. Moreover, the method discretizes the spatial part of the time dependent nonlinear boundary conditions at the free surface using a Fourier Galerkin method and integrates them in time using the classical fourth order Runge Kutta method.

It has been shown that the method can achieve highly accurate results for a number of different test cases covering very nonlinear phenomena in both shallow and deep water. The convergence of the surface velocities to the (numerically) exact velocities has been established and it has been found that the convergence rate is exponential when tested on a very wide range of water depths and wave steepnesses. Furthermore, the method has proven capable of accurately computing the vertical profile of the horizontal velocity below the wave crest for the same range of water depths and wave steepnesses. Following the analysis of the time independent part of the problem, a careful evaluation of the method's stability properties for nonlinear monochromatic waves in deep water has been given. For such waves the method is capable of stable time integration only up to a certain threshold steepness above which artificial damping must be employed in order to secure stable and accurate time integration. A method for analyzing the stability of the discretized system of equations has been presented and used to predict the threshold steepness at which the time integration becomes unstable. As the method is inherently free of the aliasing phenomena it has implicitly been shown that aliasing is not the only cause of instability related to nonlinear waves.

The method has been employed to compute the reflection of solitary waves from a wall, the corresponding dimensionless force exerted on the wall, as well as the attachment, detachment and maximum runup heights. The results match very well with the results of other highly accurate numerical methods. Using no artificial damping, the stability of this simulation is limited by a threshold steepness, but it has been argued that this instability very well might be physical.

It has been demonstrated that the method can be used to accurately simulate nonlinear wave focusing events without using any damping. The results for the free surface elevation at the focus point as well as the horizontal velocity profile below the crest at the instant of focusing have been simulated and agree well with results from experiments.

Finally, the effectiveness of the employed preconditioning strategy has been clarified and it has been found that the scaling properties of the strategy depend on the problem at hand. When tested on nonlinear monochromatic waves in deep water, the method does not admit optimal scaling if the number of wavelengths is increased with fixed resolution per wavelength. On the other hand the method does admit optimal scaling if the number of wavelengths is kept constant and the resolution per wavelength is increased.

This article is protected by copyright. All rights reserved.

Accepted Article

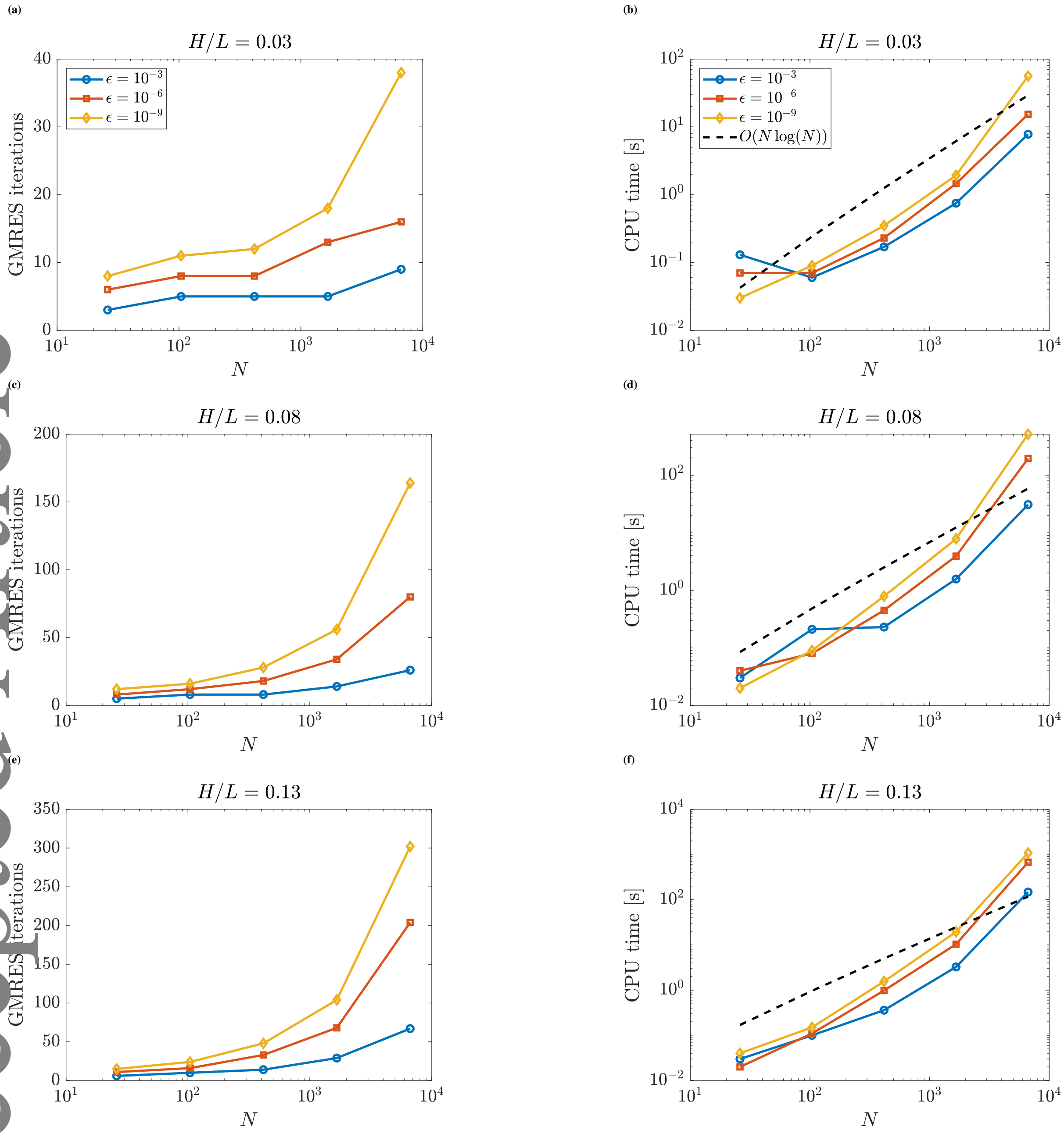

FIGURE 14 (a), (c) and (e): The number of GMRES iterations needed to solve the Laplace problem with an increasing number of wavelengths for the wave steepnesses for $H/L = 0.03$, $H/L = 0.08$ and $H/L = 0.13$, respectively, and different values of ϵ. (b), (d) and (f): The CPU time needed to to solve the Laplace problem for the same values of H/L and ϵ. The legends apply columnwise.

ACKNOWLEDGEMENT

The authors gratefully acknowledge the funding received from Centre for Oil and Gas - DTU/Danish Hydrocarbon Research and Technology Centre (DHRTC). Project name: DEWIOS. Project ID: FL_110.

Accepted Article

This article is protected by copyright. All rights reserved.

Accepted Article

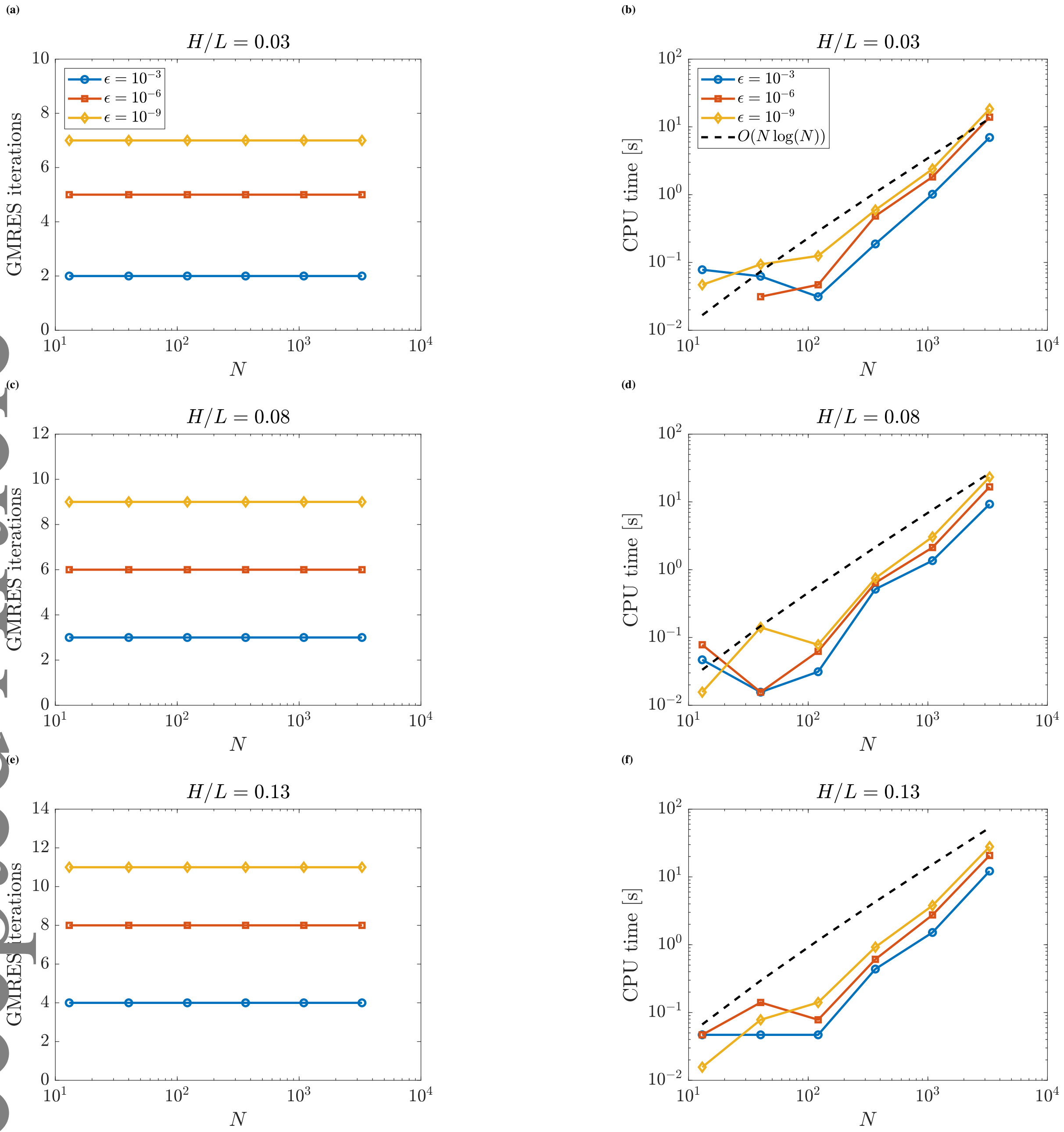

FIGURE 15 (a), (c) and (e): The number of GMRES iterations needed to solve the Laplace problem with increasing resolution per wavelength for the wave steepnesses for $H/L = 0.03$, $H/L = 0.08$ and $H/L = 0.13$, respectively, and different values of ϵ. (b), (d) and (f): The CPU time needed to to solve the Laplace problem for the same values of H/L and ϵ. The legends apply columnwise.

APPENDIX

A EFFICIENT COMPUTATION OF CONVOLUTION SUMS

An essential part of our numerical method is the efficient evaluation of the convolution sums

$$S_p^{(r)} = \sum_{n_1=-N}^{N} \cdots \sum_{n_r=-N}^{N} a_1(n_1) \cdots a_r(n_r)\, \delta_{n_1+\ldots+n_r,p}, \tag{A1}$$

This article is protected by copyright. All rights reserved.

Accepted Article

where $a_t(n_t)$ denotes the n_tth component of the vector a_t and p ranges from $-N$ to N. For this reason we here show how the sums can be computed in $O(N\log(N))$ operations using the fast Fourier transform. First, we rewrite the Kronecker delta as an integral over a complex exponential function:

$$S_p^{(r)} = \sum_{n_1=-N}^{N} \cdots \sum_{n_r=-N}^{N} a_1(n_1)\cdots a_r(n_r)\frac{1}{2\pi}\int_0^{2\pi} \exp(i(n_1+...+n_r-p)x)\,\mathrm{d}x. \tag{A2}$$

Defining the function $f_t(x)$ for $t \in \{1,2,...,r\}$ as

$$f_t(x) = \sum_{n_t=-N}^{N} a_t(n_t)\exp(in_t x), \tag{A3}$$

we can rearrange (A2) to obtain the relation

$$S_p^{(r)} = \frac{1}{2\pi}\int_0^{2\pi} f_1(x)\cdots f_r(x)\exp(-ipx)\,\mathrm{d}x. \tag{A4}$$

The integral can be computed exactly by the trapezoidal rule if $2rN+1$ points are used, meaning that we can write

$$S_p^{(r)} = \frac{1}{2rN+1}\sum_{j=0}^{2rN}\Big(f_1(x_j)\cdots f_r(x_j)\exp(-ipx_j)\Big), \tag{A5}$$

where $x_j = 2\pi j/(2rN+1)$. Now, the functions $f_1(x)$, ..., $f_r(x)$ can be evaluated at the points x_j in $O(N\log(N))$ operations by zero padding each of the vectors $[a_t(-N),...,a_t(N)]$ and using the inverse fast Fourier transform. When this is done the $2N+1$ convolution sums can be computed in $O(N\log(N))$ operations using the fast Fourier transform.

Regarding the efficiency of this method, one might point out the obvious fact that $2rN+1$ is not a power of 2 for any choice of N and that the fast Fourier transform is therefore not exploited optimally. The computational cost of the method still scales with $N\log(N)$ but with a larger constant of proportionality than in the optimal case, and for large problems this difference is crucial. For this reason our future extensions of the method will be based on Fourier series of the form

$$\sum_{n=-N}^{N-1} a_n\exp(ik_n x), \tag{A6}$$

as these allow optimal use of the fast Fourier transform since the number of degrees of freedom can be taken as a power of 2.

B OBTAINING THE COEFFICIENTS $\widehat{F}_{N,M}^{(P,Q)}$ FROM THE COEFFICIENTS $\widehat{F}_{N,M}$

It is the aim of this appendix to give a brief explanation of how the set of coefficients $\{\widehat{F}_{n,m}^{(p,q)}\}$ can be obtained from the set of coefficients $\{\widehat{F}_{n,m}\}$ where $-N \le n \le N$ and $0 \le m \le M$. Instead of working out the most general case we will restrict ourselves to $0 \le p, q \le 2$ since we do not need higher derivatives to solve the transformed Laplace equation (16).

For any function $f(x)$ on the form

$$f(x) = \sum_{n=-N}^{N} \widehat{f}_n\exp(ik_n x), \tag{B7}$$

it holds that if $f(x)$ is differentiated p times we obtain the function $f^{(p)}(x)$ given by

$$f^{(p)}(x) = \sum_{n=-N}^{N}\Big((ik_n)^p\widehat{f}_n\Big)\exp(ik_n x). \tag{B8}$$

If we let $\widehat{f}_n^{(p)}$ denote the nth expansion coefficient of $f^{(p)}(x)$ this calculation shows that $\widehat{f}_n^{(p)} = (ik_n)^p\widehat{f}_n$ for $n \in \{-N,...,N\}$. We can write this relation more compactly as

$$\begin{bmatrix}\widehat{f}_{-N}^{(p)}\\ \vdots\\ \widehat{f}_{N}^{(p)}\end{bmatrix} = \mathcal{D}_x^{(p)}\begin{bmatrix}\widehat{f}_{-N}\\ \vdots\\ \widehat{f}_{N}\end{bmatrix} \qquad \text{where} \qquad \mathcal{D}_x^{(p)} = \begin{bmatrix}(ik_{-N})^p & \dots & 0\\ \vdots & \ddots & \vdots\\ 0 & \dots & (ik_N)^p\end{bmatrix}. \tag{B9}$$

This article is protected by copyright. All rights reserved.

Accepted Article

For the function $g(s)$ given in terms of the Legendre series

$$g(s) = \sum_{m=0}^{M} \widehat{g}_n L_m(s), \tag{B10}$$

it analogously holds that the qth derivative, $g^{(q)}(s)$, can be expressed as

$$g^{(q)}(s) = \sum_{m=0}^{M} \widehat{g}_n^{(q)} L_m(s), \tag{B11}$$

due to the fact that differentiating a polynomial yields a polynomial of lower degree and because the Legendre polynomials $L_0(s), ..., L_M(s)$ span the subspace consisting of polynomials of degree less than or equal to M. At this point we are content with stating that if the matrices $D_s^{(1)}$ and $D_s^{(2)}$ with dimension $(M+1)\times(M+1)$ are defined by

$$\left(D_s^{(1)}\right)_{m,m'} = \begin{cases} 2m+1 & \text{for } m' = m+1, m+3, ... \leq M \\ 0 & \text{otherwise,} \end{cases} \tag{B12a}$$

$$\left(D_s^{(2)}\right)_{m,m'} = \begin{cases} \left(m+\frac{1}{2}\right)\left(m'(m'+1) - m(m+1)\right) & \text{for } m' = m+2, m+4, ... \leq M \\ 0 & \text{otherwise,} \end{cases} \tag{B12b}$$

it is stated in the book of Hesthaven, S. Gottlieb & D. Gottlieb[33] (see their Appendix B.1.4) that the relation

$$\begin{bmatrix} \widehat{g}_0^{(q)} \\ \vdots \\ \widehat{g}_M^{(q)} \end{bmatrix} = \mathcal{D}_s^{(q)} \begin{bmatrix} \widehat{g}_0 \\ \vdots \\ \widehat{g}_M \end{bmatrix}, \tag{B13}$$

holds for $q = 1, 2$. If we now consider the function $F(x, s)$ given by

$$F(x,s) = \sum_{n=-N}^{N} \sum_{m=0}^{M} \widehat{F}_{n,m} \exp(ik_n x) L_m(s), \tag{B14}$$

we can compute the expansion coefficients of its derivatives using the same idea of matrix-based differentiation. In fact, defining the Kronecker product of $D_s^{(q)}$ and $D_x^{(p)}$ by

$$\mathcal{D}_s^{(q)} \otimes \mathcal{D}_x^{(p)} = \begin{bmatrix} \left(\mathcal{D}_s^{(q)}\right)_{0,0} \mathcal{D}_x^{(p)} & \dots & \left(\mathcal{D}_s^{(q)}\right)_{0,M} \mathcal{D}_x^{(p)} \\ \vdots & \ddots & \vdots \\ \left(\mathcal{D}_s^{(q)}\right)_{M,0} \mathcal{D}_x^{(p)} & \dots & \left(\mathcal{D}_s^{(q)}\right)_{M,M} \mathcal{D}_x^{(p)} \end{bmatrix}, \tag{B15}$$

and the vectors containing the coefficients of $\widehat{F}$ and $\widehat{F}^{(p,q)}$ as

$$\widehat{F} = \begin{bmatrix} \begin{pmatrix} \widehat{F}_{0,-N} \\ \vdots \\ \widehat{F}_{0,N} \end{pmatrix} \\ \vdots \\ \begin{pmatrix} \widehat{F}_{M,-N} \\ \vdots \\ \widehat{F}_{M,N} \end{pmatrix} \end{bmatrix} \quad \text{and} \quad \widehat{F}^{(p,q)} = \begin{bmatrix} \begin{pmatrix} \widehat{F}_{0,-N}^{(p,q)} \\ \vdots \\ \widehat{F}_{0,N}^{(p,q)} \end{pmatrix} \\ \vdots \\ \begin{pmatrix} \widehat{F}_{M,-N}^{(p,q)} \\ \vdots \\ \widehat{F}_{M,N}^{(p,q)} \end{pmatrix} \end{bmatrix}, \tag{B16}$$

then it holds that

$$\widehat{F}^{(p,q)} = \mathcal{D}_s^{(q)} \otimes \mathcal{D}_x^{(p)} \widehat{F}. \tag{B17}$$

References

1. D. A. Kopriva, Implementing Spectral Methods for Partial Differential Equations. Springer, 2009.

2. N. A. Phillips, A coordinate system having some special advantages for numerical forecasting, J. Meteor. 14 (1957) 184-185.

This article is protected by copyright. All rights reserved.

Accepted Article

3. N. G. Freeman, A. M. Hale, M. B. Danard, A modified sigma equations approach to the numerical modelling of Great Lakes hydrodynamics, J. Geophys. Res. 77 (1972) 1050-1060.

4. B. Johns, The modelling of tidal flow in a channel using a turbulence energy closure scheme, J. Phys. Oceanogr. 8 (1978) 1042-1049.

5. B. Li, C. A. Fleming, A three dimensional multigrid model for fully nonlinear water waves, Coast. Eng. 30 (1997) 235-258.

6. H. B. Bingham, H. Zhang, On the accuracy of finite-difference solutions for nonlinear water waves, J. Eng. Math. 58 (2007) 211-228.

7. A. P. Engsig-Karup, H. B. Bingham, O. Lindberg, An efficient flexible-order model for 3D nonlinear water waves, J. Comput. Phys. 228 (2009) 2100-2118.

8. M. L. Yates, M. Benoit, Accuracy and efficiency of two numerical methods of solving the potential flow problem for highly nonlinear and dispersive water waves, Int. J. Numer. Methods Fluids 77 (2015) 616-640.

9. C. Raoult, M. Benoit, M. L. Yates, Development and validation of a 3D RBF-spectral model for coastal wave simulation, J. Comput. Phys. 378 (2019) 278-302.

10. T. B. Christiansen, H. B. Bingham, A. P. Engsig-Karup, G. Ducrozet, P. Ferrant, Efficient Hybrid-Spectral Model for Fully Nonlinear Numerical Wave Tank, Proceedings of the AMSE 2013 32nd International Conference on Ocean, Offshore and Arctic Engineering OMAE 2013.

11. V. E. Zakharov, Stability of periodic waves of finite amplitude on the surface of a deep fluid, J. Appl. Mech. Tech. Phys. 9:190-194 (1968).

12. Y. Saad, M. H. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Stat. Comput. 7 (1986) 856-869.

13. D. R. Fuhrman, H. B. Bingham, Numerical solutions of fully non-linear and highly dispersive Boussinesq equations in two horizontal dimensions, Int. J. Numer. Meth. Fluids 44 (2004) 231-255.

14. D. Clamond, D. Dutykh, Accurate fast computation of steady two-dimensional surface gravity waves in arbitrary depth, J. Fluid Mech. 844 (2018) 491-518.

15. J. A. Battjes, Surf similarity, Proceedings of the 14th Coastal Engineering Conference, Copenhagen, ASCE, (1974) 466-480.

16. M. M. Rienecker, J. D. Fenton, A Fourier approximation method for steady water waves, J. Fluid Mech. 104 (1981) 119-137.

17. D. G. Dommermuth, D. K. P. Yue, A high-order spectral method for the study of nonlinear gravity waves, J. Fluid Mech. 184 (1987) 267-288.

18. W. J. D. Bateman, C. Swan, P. H. Taylor, On the Efficient Numerical Simulation of Directionally Spread Surface Water Waves, J. Comput. Phys. 174 (2001) 277-305.

19. M. S. Longuet-Higgins, E. D. Cokelet, The deformation of steep surface waves on water I. A numerical method of computation, Proc. R. Soc. Lond. A. 350 (1976) 1-26.

20. L. Xu, P. Guyenne, Numerical simulation of three-dimensional nonlinear water waves, J. Comput. Phys. 228 (2009) 8446-8466.

21. A. P. Engsig-Karup, C. Eskilsson, D. Bigoni, Stabilised nodal spectral element method for fully nonlinear water waves, J. Comput. Phys. 318 (2016) 1-21.

22. D. R. Fuhrman, H. B. Bingham, P. A. Madsen, P. G. Thomsen, Linear and non-linear stability analysis for finite difference discretizations of high-order Boussinesq equations, Int. J. Numer. Meth. Fluids 45 (2004) 751-773.

23. M. J. Cooker, P. D. Weidman, D. S. Bale, Reflection of a high-amplitude solitary wave at a vertical wall, J. Fluid Mech. 342 (1997) 141-158.

This article is protected by copyright. All rights reserved.

24. J. Chambarel, C. Kharif, J. Touboul, Head-on collision of two solitary waves and residual falling jet formation, Nonlin. Processes Geophys. 16 (2009) 111-122.

25. Ch. E. Papoutsellis, A. G. Charalampopoulos, G. A. Athanassoulis, Implementation of a fully nonlinear Hamiltonian Coupled-Mode Theory, and application to solitary wave problems over bathymetry, Eur. J. Mech. / B Fluids. 72 (2018) 199-224.

26. P. A. Madsen, H. B. Bingham, H. Liu, A new Boussinesq method for fully nonlinear waves from shallow to deep water, J. Fluid Mech 462 (2002) 1-30.

27. A. P. Engsig-Karup, J. S. Hesthaven, H. B. Bingham, P. A. Madsen, Nodal DG-FEM solution of high-order Boussinesq-type equations, J. Eng. Math. 56 (2006) 351-370.

28. D. Dutykh, D. Clamond, Efficient computation of steady solitary gravity waves, Wave Motion 51 (2014) 86-99.

29. T. B. Johannessen, C. Swan, A laboratory study of the focusing of transient and directionally spread surface water waves, Proc. R. Soc. Lond. A 457 (2001) 971-1006.

30. D. R. Fuhrman, P. A. Madsen, Numerical simulation of extreme events from focused directionally spread wavefields, Proceedings of the 30th International Conference on Coastal Engineering Vol. 1 pp. 772-781 (2006).

31. F. Gidel, O. Bokhove, M. Kelmanson, Driven nonlinear potential flow with wave breaking at shallow-water beaches, Proceedings of the AMSE 2017 36th International Conference on Ocean, Offshore and Arctic Engineering OMAE 2017.

32. F. Gidel, Variational water-wave models and pyramidal freak waves, PhD Thesis, University of Leeds, 2018.

33. J. S. Hesthaven, S. Gottlieb, D. Gottlieb, Spectral Methods for Time-Dependent Problems, Cambridge University Press, 2007.

Accepted Article

This article is protected by copyright. All rights reserved.