

Nodal DG-FEM solution of high-order Boussinesq-type equations

Allan P. Engsig-Karup¹, Jan S. Hesthaven², Harry B. Bingham¹ and Per A. Madsen¹

¹ *Technical University of Denmark, Lyngby, Denmark.*

² *Brown University, Providence, USA.*

January 3, 2006

Abstract. We present a discontinuous Galerkin finite element method (DG-FEM) solution to a set of high-order Boussinesq-type equations for modelling highly nonlinear and dispersive water waves in one and two horizontal dimensions. The continuous equations are discretized using nodal polynomial basis functions of arbitrary order in space on each element of an unstructured computational domain. A fourth order explicit Runge-Kutta scheme is used to advance the solution in time. Methods for introducing artificial damping to control mild nonlinear instabilities are also discussed. The accuracy and convergence of the model with both h (grid size) and p (order) refinement are verified for the linearized equations, and calculations are provided for two nonlinear test cases in one horizontal dimension: harmonic generation over a submerged bar; and reflection of a steep solitary wave from a vertical wall. Test cases for two horizontal dimensions will be considered in a future paper.

Keywords: Discontinuous Galerkin finite element method, high-order Boussinesq-type equations, gravity waves, unstructured grids.

1. Introduction

Boussinesq-type equations are in widespread use by coastal engineers for the numerical simulation of nonlinear wave motion in near-shore regions. The main idea behind these equations is to reduce the three dimensional problem to a two dimensional problem by assuming a polynomial variation in the vertical direction. Such equations represent an important engineering tool for the prediction of wave-wave, wave-bottom, and coastal wave-structure interaction; phenomena which are critical to the design of *e.g.* offshore windmill foundations, harbours, and exposed loading facilities.

There are a large number of Boussinesq-type models in use today, and a comprehensive review of the history of Boussinesq theory can be found in Madsen and Schäffer (1999). Basically, there are two classes of Boussinesq models: *two-variable* formulations which are in terms of an elevation and a horizontal velocity or flux variable; and *three-variable* formulations which also retain the vertical component of velocity as an unknown. Two-variable formulations, for example Peregrine (1967),



© 2006 Kluwer Academic Publishers. Printed in the Netherlands.

Madsen and Sørensen (1992) or Nwogu (1993) are generally more efficient, but have a limited radius of convergence and are thus limited to a distinct value of kd (k the wave number and d the water depth) regardless of the order retained in the expansion in terms of dispersion. Three-variable formulations, on the other hand, can have an infinite radius of convergence, as shown by Madsen and Agnon (2003). We focus here on the most accurate of these yet developed, that of Madsen, Bingham and Liu (2002). The Padé (4,4) version of this method is capable of accurately modelling nonlinear waves up to the point of breaking and out to relative water depths of approximately $kd = 25$, while accurate kinematics (the vertical variation of the flow) are obtained out to approximately $kd = 10$. The Padé (2,2) version obtained by using the optimized coefficients determined in Jamois, Fuhrman and Bingham (2005) is accurate in dispersion out to approximately $kd = 10$ and with accurate kinematics to approximately $kd = 4$. These equations are thus appropriate for modelling a broad range of water wave phenomena in near-coastal environments. This additional accuracy and range of dispersion is however obtained at the cost of additional variables (degrees of freedom) and more complicated systems of linear equations to solve relative to the two-equation models; and this provides new challenges for obtaining efficient numerical solutions.

Many of these challenges have been successfully overcome using a structured (uniformly-spaced, rectangular) grid finite difference solution as reported by Fuhrman, Bingham and Madsen (2004) and previous publications cited therein. This finite difference scheme is however at a fixed order of accuracy; and it lacks geometric flexibility both in terms of local grid refinement and the inclusion of other than piecewise rectangular structures. The goal of the present work is to provide a solution with both geometric and order flexibility by means of an unstructured DG-FEM. We note that the unstructured grid is particularly attractive since it allows the grid to be adapted to the physical scales of the problem which can potentially lead to a significant reduction in the computational effort.

The unstructured finite element method (FEM) has been applied to solve various two-variable Boussinesq-type formulations. The classical Boussinesq equations of Peregrine (1967) were solved by Antunes Do Carmo and Seabra Santos (1993), Ambrosi and Quartapelle (1998) and again by Eskilsson and Sherwin (2004). A set of weakly dispersive Boussinesq-type equations was solved and studied by Langtangen and Pedersen (1998). Li, Liu, Yu and Lai (1999) solved the improved Boussinesq equations by Beji and Nadaoka (1996). The extended Boussinesq equations by Madsen and Sørensen (1992) were solved using a FEM model by Sørensen, Schäffer and Sørensen (2004), while the equations

of Nwogu (1993) were solved by Walkley and Berzins (1999,2002), and Eskilsson and Sherwin (2003,2005). Recently, Eskilsson and Sherwin (2002,2005) solved the same set of extended equations using a DG-FEM method.

At first glance it may seem strange to use a discontinuous method to solve continuous problems where we expect smooth solutions (*i.e.* up to wave breaking); especially when a continuous form of the method is available. The reasons for this choice are: Ease of implementation; and improved computational efficiency due to the local nature of the DG-FEM, which is especially important for implementing higher-order operators (in this case up to fifth order derivatives).

The remainder of this paper is organized as follows. Section 2 reviews the governing equations of the high-order Boussinesq-type formulation which we seek to solve. Section 3 presents a method of lines stability analysis of the continuous linearized equations in both one and two horizontal dimensions. Section 4 describes the numerical methods chosen for discretizing the governing equations in one horizontal dimension. A discrete stability analysis is given along with descriptions of the techniques used for wave generation and absorption as well mild filtering techniques which is necessary to stabilize the nonlinear calculations. Section 5 provides some numerical results obtained using the proposed methodology. The last section provides a summary along with concluding remarks.

2. The high-order Boussinesq-type equations

If viscous effects are small enough to be neglected then potential flow theory provides a reasonably good mathematical description of the flow of water waves. Boussinesq-type equations are derived from an exact infinite series solution to the fully nonlinear potential flow problem.

The fluid motion is described by the horizontal velocities $\hat{\mathbf{u}}$, the vertical velocity \hat{w} evaluated at the expansion level $z = \hat{z}$ and the surface elevation η . The variables defined at the expansion level can be related to the surface variables defined at $z = \eta(\mathbf{x})$, which are denoted by a ' \sim '. The still water depth $d(\mathbf{x})$ is measured from the still water level (SWL), which is positioned at the still water surface where $z = 0$. See sketch in Figure 1. The gravitational acceleration constant g is set to $9.81m^2/s$.

The set of high-order Boussinesq-type equations are given in terms of variables defined directly on the free surface as

$$\partial_t \eta = \tilde{w} - \nabla \eta \cdot \tilde{\mathbf{u}}, \quad (1)$$

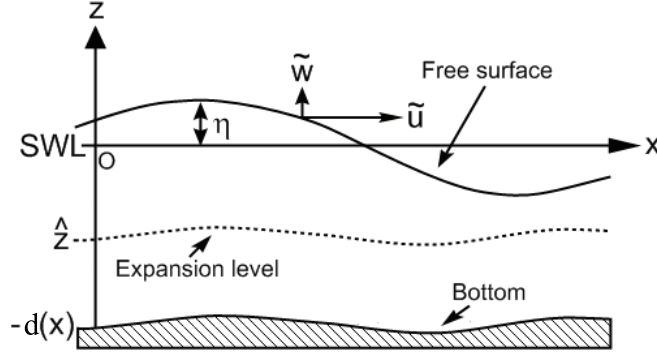


Figure 1. Definition sketch. One horizontal dimension.

$$\partial_t \tilde{\mathbf{U}} = -\nabla \left(g\eta + \frac{\tilde{\mathbf{U}} \cdot \tilde{\mathbf{U}}}{2} - \frac{\tilde{w}^2}{2} (1 + \nabla \eta \cdot \nabla \eta) \right), \quad (2)$$

where $\tilde{\mathbf{U}} \equiv \tilde{\mathbf{u}} + \tilde{w} \nabla \eta = (\tilde{U}, \tilde{V})^T$ and $\nabla = (\partial_x, \partial_y)^T$ is the two dimensional gradient operator in a Cartesian coordinate system and ∂_x is the partial derivative with respect to the \mathbf{x} -coordinate. From appropriate initial conditions the free surface variable $\eta(\mathbf{x}, t)$ is evolved for $t > t_0$ by Eq. (1) and the tangential velocity $\tilde{\mathbf{U}}(\mathbf{x}, t)$ by the momentum equation Eq. (2). Note, that $\mathbf{x} \in \Omega$. However, in order to evolve these variables the free surface vertical velocity $\tilde{w}(\mathbf{x}, t)$ has to be determined. This is done by first solving an algebraic system and then using the solution to compute the vertical free surface velocity. In one horizontal dimension (1D) solve

$$\begin{bmatrix} \tilde{U} \\ 0 \end{bmatrix} = \begin{bmatrix} A_2 - \partial_x \eta \cdot B_2 & B_2 + \partial_x \eta \cdot A_2 \\ A_3 + \partial_x d \cdot S_1 & B_3 - \partial_x d \cdot S_2 \end{bmatrix} \begin{bmatrix} \hat{u}^* \\ \hat{w}^* \end{bmatrix}, \quad (3)$$

and then the vertical free surface velocities can be obtained from

$$\tilde{w} = -B_2 \hat{u}^* + A_2 \hat{w}^*. \quad (4)$$

In two horizontal dimensions (2D) solve

$$\begin{bmatrix} \tilde{U} \\ \tilde{V} \\ 0 \end{bmatrix} = \begin{bmatrix} A_{11} - \partial_x \eta \cdot B_{11} & A_2 - \partial_x \eta \cdot B_{12} & B_{11} + \partial_x \eta \cdot A_1 \\ A_2 - \partial_y \eta \cdot B_{11} & A_{22} - \partial_y \eta \cdot B_{12} & B_{12} + \partial_y \eta \cdot A_1 \\ A_{01} + S_{01} & A_{02} + S_{02} & B_0 + S_{03} \end{bmatrix} \begin{bmatrix} \hat{u}^* \\ \hat{v}^* \\ \hat{w}^* \end{bmatrix}, \quad (5)$$

and similarly to the 1D case the vertical free surface velocities can be computed from the solution to this algebraic system as

$$\tilde{w} = -B_{11} \hat{u}^* - B_{12} \hat{v}^* + A_1 \hat{w}^*. \quad (6)$$

The terms allowing for a mildly varying bottom are given as

$$S_{01} = \partial_x d \cdot C_{11} + \partial_y d \cdot C_2, \quad (7)$$

$$S_{02} = \partial_x d \cdot C_2 + \partial_y d \cdot C_{22}, \quad (8)$$

$$S_{03} = -\partial_x d \cdot C_{13} - \partial_y d \cdot C_{23}. \quad (9)$$

The different continuous operators used to define the complete equation systems Eqs. (3)-(9) are given in Appendix A. Further, the structure of the global operators on the right hand sides in Eq. (3) or Eq. (5) is *unsymmetric*.

A similar solution procedure to the one suggested by Madsen, Bingham and Liu (2002) is used for solving the set of governing equations. Madsen *et al.* also concludes that a constant expansion level of approx. $\hat{z} = -0.5d$ leads to optimal accuracy with respect to the vertical velocity distribution and therefore this expansion level is chosen throughout this paper. This choice simplifies the linear system in Eq. (5) and results in a block symmetric system, and therefore for this specific case reduces the number of different operators.

At the physical domain boundaries we generally specify impermeable wall conditions. If needed, we generate waves internally using relaxation zones and approximate radiation boundaries using sponge layers. See section 4.9.

3. Stability considerations

Fuhrman, Bingham, Madsen and Thomsen (2000) considered numerical stability for their finite difference model and found it to be linearly stable and under sufficient artificial damping to be also nonlinearly stable.

In this section we consider the linear stability of the analytic problem by a Fourier analysis. It is shown that the eigenspectra of the continuous operators are purely imaginary and upper bounds for the maximum eigenvalues are found.

3.1. FOURIER ANALYSIS

The starting point for the Fourier analysis is to consider the linearized equation system in a periodic domain containing a flat bottom, which can be given in the semi-discrete form as

$$\partial_t \mathbf{q} = L(\mathbf{q}) = J\mathbf{q}, \quad (10)$$

where $\mathbf{q} = (\eta, \mathbf{u})^T$ is a vector of primitive variables. In 1D this gives the form as

$$\partial_t \begin{bmatrix} \eta \\ u \end{bmatrix} = \begin{bmatrix} 0 & J_{12} \\ -g\partial_x & 0 \end{bmatrix} \begin{bmatrix} \eta \\ u \end{bmatrix}, \quad (11)$$

where we solve Eq. (3) and insert the solution into Eq. (4) to obtain the continuous operator

$$J_{12} = \frac{A_2 A_3 + B_2 B_3}{A_3 B_2 - A_2 B_3}. \quad (12)$$

We perform a Fourier linear stability analysis of the Padè (4,4) formulation by Madsen, Bingham and Liu (2002) and assume a harmonic variation in space

$$\eta(x, t) = \hat{\eta}(t)e^{ikx}, \quad u(x, t) = \hat{u}(t)e^{ikx} \quad (13)$$

in Eq. (11). We find that in the Fourier space the operator is given as

$$J_{12} = -i \frac{60kd(1008 + 112(kd)^2 + (kd)^4)(15120 + 420(kd)^2 + (kd)^4)}{914457600 + 431827200(kd)^2 + 25804800(kd)^4 + 408240(kd)^6 + 1740(kd)^8 + (kd)^{10}}, \quad (14)$$

and the eigenvalues of the Jacobian J in Eq. (11) are

$$\lambda(J) = \pm i \sqrt{\frac{(g/d)(kd)^2 60(15240960 + 2116800(kd)^2 + 63168(kd)^4 + 532(kd)^6 + (kd)^8)}{914457600 + 431827200(kd)^2 + 25804800(kd)^4 + 408240(kd)^6 + 1740(kd)^8 + (kd)^{10}}}, \quad (15)$$

which are both complex and imaginary. Further, in the shallow water limit $kd < 0.31$, the eigenvalues reduce to $\lambda(J_{SWE}) = \pm k\sqrt{gd}$ - i.e. they reduce to the eigenvalues from the linearized shallow water equations.

We conclude that; i) the eigenvalues of the linearized Boussinesq formulation reduce to the shallow water eigenvalues for relatively low depths and ii) the exact eigenspectra of the linearized formulation are imaginary and iii) the maximum absolute eigenvalue of the continuous problem is bounded for a constant still water depth d since from Eq. (15) we find that

$$|\lambda_{max}^{1D}| = \lim_{kd \rightarrow \infty} |\lambda(J)| = 2\sqrt{15}\sqrt{\frac{g}{d}}. \quad (16)$$

In other words, for a given still water depth d , the maximum absolute eigenvalue of the Jacobian increases with kd . Increasing kd on a fixed depth d corresponds to allowing the wave length $L \rightarrow 0$, i.e. the smallest (and fastest) wave harmonic determines the size of the absolute maximum eigenvalue on a fixed depth.

Similar results are obtained for the Padé (2,2) version, and for both versions of the formulation in 2D. See Table I. The upper bound for the eigenvalues remains the same for the Padé (2,2) formulation in both 1D and 2D. For the Padé (4,4) formulation, the upper bound is lower than the one found for the one dimensional case. Thus, the above analysis

Table I. Analytical bounds for $|\lambda_{max}|\sqrt{\frac{d}{g}}$ for the linearized formulations in different horizontal dimensions. $\sigma = -0.5$.

	1D	2D
Padé (2,2)	$2\sqrt{6}$	$2\sqrt{6}$
Padé (4,4)	$2\sqrt{15}$	$3\sqrt{5}$

leads to the expectation that the less accurate Padé (2,2) formulation retaining third order rather than fifth order differential operators as in the Padé (4,4) formulation, has better stability properties if the discrete properties of the Jacobian mimic this behavior.

4. Numerical methods

In this section we outline the numerical methods used to solve the governing Eqs. (3)-(9). A method of lines approach has been used, where the spatial and temporal discretizations are considered separately. The discontinuous Galerkin finite element method is used for the spatial discretization of the governing equations and the resulting semi-discrete equation system is solved using an explicit Runge-Kutta method to evolve the equations in time from appropriate initial conditions.

4.1. THE DISCONTINUOUS GALERKIN FINITE ELEMENT METHOD

The Discontinuous Galerkin Finite Element Method (DG-FEM) is used for the spatial discretization. Originally, the method was developed for solving conservation laws, e.g. see review on DG-FEM in Cockburn and Shu (2001). In this paper, the DG-FEM is used to discretize and solve the high-order Boussinesq-type formulation, which is not given in conservation form.

The DG-FEM is a high-order *spectral* finite element method. On each element, it is based on the Galerkin form of the more general Mean Weighted Residual methods, which determines the way we satisfy (or minimize the residual of) our governing PDE when we seek an

approximate solution. Before constructing any DG-FEM scheme, the domain is subdivided into local elements. The main idea of the DG-FEM is to allow the solution to be discontinuous over the local element boundaries and only couple adjacent elements using suitable numerical fluxes. The choice of numerical fluxes can be borrowed from the finite volume framework, which is theoretically well-established. The DG-FEM combined with nodes defined directly at the boundaries offers the opportunity to determine the numerical fluxes directly. Further, the method can by construction easily handle complex geometries, irregular meshes, and polynomials of different degrees in different elements. Thus the method supports hp -adaptive strategies for obtaining convergent solutions. A major advantage of the DG-FEM is that it conceptually generalizes to higher dimensions, and therefore once familiar with the one dimensional method, it is straightforward to apply the concepts in higher dimensions.

4.2. SUBDIVISION OF THE COMPUTATIONAL DOMAIN

Before applying the discontinuous Galerkin procedure to discretize these equations in space, we partition the computational domain into K non-overlapping elements, such that $\Omega = \bigcup_{k=1}^K \Omega_k$. The size of the k 'th element is denoted as h_k .

On the k 'th element the solution is approximated locally by a finite sum as

$$u_N^k(\mathbf{x}) = \sum_{n=0}^N \hat{u}_n^k \phi_n(\mathbf{x}) = \sum_{n=0}^N u_n^k l_n(\mathbf{x}), \quad (17)$$

where we have $N + 1$ degrees of freedom inside each element. The $l_n(\mathbf{x})$ are the Lagrange polynomials defined on the same set of nodes as used in combination with the chosen orthogonal basis. The basis functions $\phi_n, i = 0, \dots, N$ are chosen such that $u_N^k \in V_k^N$, where the approximation space V_k^N is defined as

$$V_k^N = \{v : v_k \in P^N(\Omega_k), \forall \Omega_k \in \Omega\}, \quad (18)$$

i.e. polynomials of at most order N in each element. The classical Legendre polynomials $P_n(x)$ are used as our orthogonal basis in 1D and are normalized such that

$$\phi_n(x) = \frac{P_n(x)}{\sqrt{2n+1}}. \quad (19)$$

As shown in Hesthaven and Warburton (2002) we can choose to represent our polynomial basis functions in a Lagrangian (i.e. a nodal) basis and we use the Legendre-Gauss-Lobatto nodes in one dimension. By

uniqueness of the polynomial interpolations, the following (constant) transformation matrix can be defined for transforming the variables between the modal and nodal spaces

$$V_{ij} = \phi_j(\mathbf{x}_i), \quad i, j = 0, \dots, N. \quad (20)$$

Here \mathbf{x}_i is the i 'th nodal point inside the k 'th element. This leaves us with the following relationship between the modal and nodal spaces and the basis functions

$$u_N = V\hat{u}_N, \quad \phi_i(\mathbf{x}) = V^T l_i(\mathbf{x}). \quad (21)$$

These relationships allow us to determine both the mass matrix and the stiffness matrix exactly for a standard element without using high-order Gaussian quadrature rules for the elemental integrals.

4.3. SPATIAL DISCRETIZATION

The high-order Boussinesq formulation constitutes a complex system of equations and numerically we are faced with a number of challenges. In particular, the time-dependent equations Eq.'s (1)-(2) are nonlinear and not given in a conservative form, and the involved algebraic system Eq. (3) or Eq. (5) is nonlinear, time-dependent and contains high-order differential operators.

To apply the DG-FEM we first rewrite Eqs. (1)-(2) slightly by introducing some auxiliary variables as

$$\partial_t \eta = g(\mathbf{x}), \quad (22)$$

$$g(\mathbf{x}) = \tilde{w} - \nabla \eta \cdot (\tilde{\mathbf{U}} - \tilde{w} \nabla \eta), \quad (23)$$

where $g(x)$ is assumed to be a source function, and

$$\partial_t \tilde{\mathbf{U}} = -\nabla F(\mathbf{x}), \quad (24)$$

$$F(\mathbf{x}) = g\eta + \frac{\tilde{\mathbf{U}} \cdot \tilde{\mathbf{U}}}{2} - \frac{\tilde{w}^2}{2}(1 + \nabla \eta \cdot \nabla \eta). \quad (25)$$

Following the DG-FEM procedure, for each element in the domain we multiply Eq.'s (22) and (24) with a smooth weight function $v(x) = \phi(x)$. Two integration by parts are done on the terms containing spatial derivatives, i.e. only the second term in Eq. (24). In the intermediate step of these integrations, the analytic flux function $F(\mathbf{x})$ is interchanged with a continuous numerical flux function $F^*(\mathbf{x}^{k,-}, \mathbf{x}^{k,+})$ to be chosen, which allow us to connect adjacent elements. By this approach, the starting point for the *strong* DG-FEM formulation of Eq.'s (22) and (24) for the k 'th element, $k = 1, \dots, K$, becomes

$$\int_{\Omega_k} \phi \partial_t \eta d\mathbf{x}^k = \int_{\Omega_k} \phi g(\mathbf{x}) d\mathbf{x}^k, \quad (26)$$

$$\int_{\Omega_k} \phi \partial_t \tilde{U} d\mathbf{x}^k = - \int_{\Omega_k} \phi \nabla F(\mathbf{x}) d\mathbf{x}^k + \oint_{\partial\Omega_k} \phi \mathbf{n} \cdot (F(\mathbf{x}) - F^*(\mathbf{x})) d\mathbf{x}^k, \quad (27)$$

where $\mathbf{n} = (n_x, n_y)^T$ is an outward pointing normal vector to the k 'th element.

Before arriving at the final numerical scheme, we need to replace the continuous variables with finite polynomial approximations of the form given in Eq. (17), and choose suitable consistent numerical fluxes. We choose the simplest numerical fluxes available, namely central fluxes or Lax-Friedrichs fluxes given by the trace operators

$$F_{CF}^*(a, b) = 0.5(F(a) + F(b)), \quad (28)$$

$$F_{LF}^*(a, b) = 0.5(F(a) + F(b) + s(a - b)), \quad s = \max\{|F'(u)|\}. \quad (29)$$

The central fluxes are used for computing spatial derivatives and the (global) Lax-Friedrichs fluxes are used for the time-dependent momentum Eq. (27) to introduce a small amount of artificial viscosity into the numerical scheme.

In 1D, the final scheme for the time-dependent equations can be written in a compact form (discrete operator notation borrowed from Sherwin (1999)) using the following discrete elemental operators defined and computed for a standard element as

$$M_{ij} = \int_{-1}^1 \phi_i(\xi) \phi_j(\xi) d\xi, \quad M = (VV^T)^{-1}, \quad (30)$$

$$S_{ij} = \int_{-1}^1 \phi_i(\xi) \phi_j'(\xi) d\xi, \quad S = MD, \quad (31)$$

$$E_{ij} = \phi_i(-1) \phi_j(-1), \quad F_{ij} = \phi_i(-1) \phi_j(1), \quad (32)$$

$$G_{ij} = \phi_i(1) \phi_j(-1), \quad H_{ij} = \phi_i(1) \phi_j(1), \quad (33)$$

where M is the *Mass Matrix* and S is the *Stiffness matrix* and D is the nodal differentiation matrix.

Using the given discrete standard operators with a nodal representation, i.e. set $\phi_i(x) = l_i(x)$, we can write the numerical scheme in 1D for the k 'th element in compact notation as

$$\frac{h_k}{2} M \frac{d}{dt} \eta_N^k = \frac{h_k}{2} M g_N^k, \quad (34)$$

$$g_N^k = \tilde{w}_N^k - f_N^k \tilde{u}_N^k, \quad (35)$$

$$\frac{h_k}{2} M f_N^k = S \eta_N^k + 0.5(G \eta_N^{k+1} - H \eta_N^k) - 0.5(F \eta_N^{k-1} - E \eta_N^k), \quad (36)$$

and using Lax-Friedrichs numerical fluxes

$$\frac{h_k}{2} M \frac{d}{dt} \tilde{U}_N^k = S F_N^k + 0.5(G F_N^{k+1} - H F_N^k) - 0.5(F F_N^{k-1} - E F_N^k)$$

$$- s(0.5(G\tilde{U}_N^k - H\tilde{U}_N^{k-1}) - 0.5(F\tilde{U}_N^k - E\tilde{U}_N^{k-1})), \quad (37)$$

$$F_N^k = g\eta_N^k + \frac{(\tilde{U}_N^k)^2}{2} - \frac{(\tilde{w}_N^k)^2}{2}(1 + (f_N^k)^2). \quad (38)$$

To arrive at this numerical scheme, it should be clear that in Eq. (36) we obtained a DG-FEM approximation to $f = \partial_x \eta$ by using a strong form DG-FEM and central fluxes. Also, the nonlinear terms in the functions given in Eq. (23) and Eq. (25) are determined using direct products as shown in Eq. (35) and Eq. (38) at the grid nodes in a collocation manner instead of using aliasing-free orthogonal projections. This approach is used for efficiency reasons and is both convenient and consistent, but may lead to aliasing errors in the computations requiring the use of a de-aliasing technique.

To set up a numerical scheme for the algebraic system Eq. (3) or Eq. (5) requires a global assembly procedure for the discrete operators to be assembled into global operators. For the finite domain case, this procedure involves the construction of the first order differential operators having suitable boundary conditions imposed. However, for convenience we present the assembly procedure for a periodic domain, since the modifications for the finite domain case are straightforward.

The discrete global operators are constructed using a Local Discontinuous Galerkin (LDG) method. Central fluxes are used to connect the elements as in Bassi and Rebay (1997). Each of the high-order differential operators in the system are decomposed into a set of first order differential operators. E.g. the third derivative operator in x as

$$z = \partial_{xxx}u \rightarrow z = \partial_x q, \quad q = \partial_x r, \quad r = \partial_x u. \quad (39)$$

The high-order differential operators are determined by matrix-matrix products of the discrete global first order differential operators with suitable boundary conditions imposed.

The global assembly of the discrete first order differential operators defined on a periodic domain needs to be determined from the local representation on each element. Consider the PDE

$$\partial_x u = f, \quad x \in [0, L], \quad (40)$$

which by the strong form of the DG-FEM procedure leads to the local scheme

$$\int_{\Omega^k} \phi \partial_x u dx^k + \oint_{\partial\Omega^k} \phi \mathbf{n} \cdot (u^* - u) dx^k = \int_{\Omega^k} \phi f dx^k. \quad (41)$$

The computational domain is subdivided in the usual way into K elements. Then for each element, we can write the numerical nodal

scheme as

$$Su_N^k + 0.5(Gu_N^{k+1} - Hu_N^k) - 0.5(Fu_N^{k-1} - Eu_N^k) = \frac{h_k}{2} M f_N^k. \quad (42)$$

Now, if we for simplicity only consider two elements (set $K = 2$) and use central fluxes to connect our elements, then we can write the globally assembled numerical scheme as

$$\begin{bmatrix} S + 0.5(E - H) & 0.5(G - F) \\ 0.5(G - F) & S_x + 0.5(E - H) \end{bmatrix} \begin{bmatrix} u_N^1 \\ u_N^2 \end{bmatrix} = \begin{bmatrix} \frac{h_1}{2} M & 0 \\ 0 & \frac{h_2}{2} M \end{bmatrix} \begin{bmatrix} f_N^1 \\ f_N^2 \end{bmatrix}, \quad (43)$$

or in a more compact form as

$$D_x^G u = f, \quad (44)$$

where both u and f are $DoF \times 1$ sized vectors, where $DoF = K \cdot (N+1)$, and the global differentiation operator with respect to x with boundary terms included is defined as

$$D_x^G = \begin{bmatrix} \frac{\Delta x_1}{2} M & 0 \\ 0 & \frac{\Delta x_2}{2} M \end{bmatrix}^{-1} \begin{bmatrix} S + 0.5(E - H) & 0.5(G - F) \\ 0.5(G - F) & S + 0.5(E - H) \end{bmatrix}. \quad (45)$$

To construct the global third order differential operator given in Eq. (39) we can easily construct it from this global operator as

$$\partial_{xxx} \rightarrow D_x^G \cdot D_x^G \cdot D_x^G. \quad (46)$$

By this approach, the high-order differential operators can be built. For example, the 1D operator A_2 in Eq. (3), can be discretized as

$$A_2^G = I - \alpha_2 \cdot (D_x^G)^2 + \alpha_4 \cdot (D_x^G)^4, \quad (47)$$

where I is the unitary matrix of size $DoF \times DoF$ and $\alpha_i, i = 2, 4$ are diagonal matrices.

When each of the discrete block-differential operators are constructed in this way, we can easily assemble the discrete global operator in Eq. (3).

The assembly of the nonlinear operators have been done in a collocation manner. For example,

$$\partial_x \eta \cdot B_2 \rightarrow \eta_x \cdot B_2^G, \quad (48)$$

where η_x is a diagonal matrix holding the discrete nodal values of $\partial_x \eta$. This approach may lead to aliasing errors, however it allows for a fast and efficient way of reconstructing the operators, which is necessary for the fully nonlinear simulations to keep the computational costs low.

Due to the locality of the DG-FEM, the global assembly result in a relatively sparse global matrix. This matrix can be reordered in a preprocessing step such that the bandwidth of the matrix in the 1D finite domain case is given as $2 \times (N + 1) \times (2q_{max} + 1)$, where q_{max} is the order of the largest derivative in the formulation. Thus, the bandwidth is dependent on both the local polynomial order and the highest derivative in the block operators. The global matrix system is usually sparse and hence it can be solved efficiently by an appropriate sparse solution method. Since this global matrix system has to be both built (for the fully nonlinear problem) and solved at every time step, this becomes the computational bottleneck in solving the numerical scheme. For the present computations we have used a direct sparse solution technique.

4.4. BOUNDARY CONDITIONS

In the current models, we have imposed impermeable wall boundary conditions using a *symmetry technique* (or mirror principle) and whenever needed radiation boundaries are setup using relaxation zones.

The impermeable wall boundary conditions are imposed weakly at the outer boundaries of the computational domain through the numerical flux function, which is denoted by q^* here. Hence, by the symmetry technique dirichlet boundary-type conditions are imposed as

$$q^* = -q^-, \quad (49)$$

such that the averages of the two distinct boundary values at each boundary face correspond to the values we seek to impose. Neumann boundary-type conditions are imposed as

$$q^* = q^-, \quad (50)$$

such that the averages are left unmodified.

By this technique the boundary conditions are imposed approximately to the accuracy of the scheme.

4.5. TEMPORAL DISCRETIZATION

A fourth order $2N$ -low storage explicit Runge-Kutta scheme with 5 stages (labelled LSERK45 here) by Carpenter and Kennedy (1994) is used for solving the semi-discrete governing equations in time. The absolute stability region encloses a large part $\lambda\Delta t \leq 3.345$ of the imaginary (inviscid stability) axis in the complex plane and is easily implemented.

For hyperbolic-type problems which are solved using explicit time-integration schemes, where $\lambda\Delta t$ should reside inside the absolute stability region of the chosen time integration scheme for the numerical scheme to be *absolutely stable*, the maximum time step Δt_{max} is restricted by

$$\Delta t_{max} \leq \frac{|s_A|}{|\lambda_{max}|}, \quad (51)$$

where s_A is the size of the absolute stability region measured on the imaginary axis at the point where the absolute stability region crosses the imaginary axis in the complex plane.

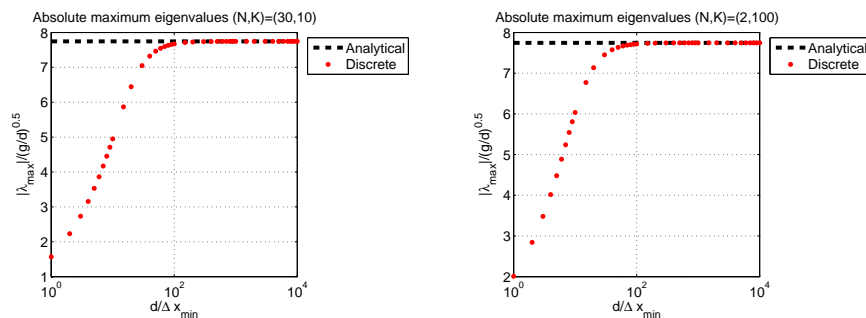
4.6. LINEAR STABILITY

A numerical study of linear stability in 1D was carried out to investigate the linear stability properties of the DG-FEM scheme to determine if the analytical bounds given in Table I holds for the discrete cases. The study was performed using structured domains at various constant depths, where both the number of elements and the fixed local polynomial orders were varied. From the resulting discrete eigenspectra we numerically determined the magnitude of the largest eigenvalue for the given discretization and depth.

It was found that the discrete eigenspectra of the numerical schemes based on the linearized formulation in 1D result in an upper bound closely matching the analytical results presented in Table I. This has been illustrated in Figure 2 by two tests; the first test using few elements and a high local polynomial order, and in the second test many elements and low local polynomial order. Remarkably, linear stability in 1D was found to be *independent* of the size of the elements, the number of elements and also the local polynomial order used within the elements (as long as the chosen order lead to well-conditioned local operators, which is the case for all moderate polynomial orders used in applications, usually $N < 10$).

Therefore, the time step size in the linearized model in 1D can in fact be chosen based on accuracy considerations alone as long as the analytical criterion is satisfied. However, we conclude from Figure 2 that for shallow depths the upper bound for linear stability is too restrictive if the smallest scale waves cannot be resolved on the grid.

Finally, we note that the time step is usually chosen based on accuracy considerations alone, and therefore the above results are mainly important for testing an application.



a) High local polynomial order and few elements. b) Low local polynomial order and many elements.

Figure 2. Computed normalized absolute maximum eigenvalues for the linearized Padé (4,4) formulation for constant depths in 1D using a structured domain.

4.7. GENERAL STABILITY

For the more general cases a CFL-like criterion is used. The general cases includes both varying depth and fully nonlinear problems, in which cases the linear stability criteria Eq. (51) or the analytical bounds in Table I do not hold. As is usually the case with most numerical methods, the upper limit for the time step is restricted by the numerical growth of the eigenvalues of the discrete operators. Instead, we use the following CFL-like condition

$$\Delta t \geq CFL \frac{\Delta x_{min}}{c_{max}}, \quad (52)$$

where the maximum wave propagation speed is estimated using the general linear dispersion relation. Here Δx_{min} is set to be the smallest distance between any two points on an element in the domain. The magnitude of the CFL constant is $O(1)$.

4.8. DE-ALIASING TECHNIQUES

For nonlinear problems which lack (any) numerical diffusivity, we may need to maintain numerical stability by applying a general filtering or de-aliasing technique. In the presented models, we have used the DG-FEM in a collocation manner which may give rise to aliasing errors, since we have chosen to evaluate the nonlinear terms by direct products for computational efficiency. To reduce such aliasing errors we have used the filtering techniques mentioned here.

For mildly nonlinear cases in 1D it was found to be sufficient to use a mild nodal filter constructed in a similar way as the one presented in Fischer and Mullen (2001). Instead of constructing the filter on the

Gauss-Legendre (GL) nodes, we instead base the filter on the Lobatto-Gauss-Legendre (LGL) nodes. Hence, following their procedure, the filter is constructed as follows. Let I_N^M be an interpolation operator for a polynomial of degree N defined on $N + 1$ GLL nodes onto the $M + 1$ GLL nodes. Define $\prod_N := I_N^{N+1} I_{N+1}^N$ to be a projection from P_{N+1}^1 to P_N^1 on $[-1,1]$. The filter is constructed as

$$F_\alpha := \alpha \prod_N + (1 - \alpha) I_{N+1}^{N+1}. \quad (53)$$

For mildly nonlinear problems in 1D we have found that this filter works well with approx. $\alpha = 0.1 - 0.2$.

For more extreme nonlinear wave problems the model has been successfully stabilized with a Savitzky-Golay-type (SG) polynomial averaging filter, see Savitzky and Golay (1964), which has also been used by Fuhrman and coworkers (e.g. Jamois, Fuhrman and Bingham (2005)). The SG filter was then applied locally by using an iso-parametric approximation procedure after having averaged the interface values to an equi-distant/structured grid where the polynomial filtering could be carried out before returning (by interpolation back) to the original grid. It is noted that this approach is semi-local in the sense that the averaging is done *across* element interfaces locally. This approach was found to be practical in order to avoid unnecessary large jump discontinuities and to reduce any aliasing errors in the solution which might otherwise have been introduced by alternative filtering procedures. The computational cost of this approach is negligible in comparison with the cost of solving the algebraic system every time step when it is set up in the pre-processing stage before the computations.

However, in 2D models where complex geometries may be used, the SG filtering procedure is impractical and hence alternative de-aliasing means are to be found. Possibly, we need to reduce the aliasing errors by a more consistent or exact evaluation of the integrands for handling the nonlinear terms in the DG-FEM, and subsequently project the solution to the original grid.

4.9. RELAXATION ZONES

We need to be able to both absorb and generate waves which are generally uni-directional inside the computational domain. A convenient method by Larsen and Dancy (1983) for both generating and absorbing waves while preserving the outer domain boundary conditions is to modify the discrete function values in a smooth way in space using a relaxation function $\Gamma(\mathbf{x})$ such that the relaxed solution $u^*(\mathbf{x})$ is given as

$$u^*(\mathbf{x}_i) = \Gamma(\mathbf{x}_i)u(\mathbf{x}_i) + (1 - \Gamma(\mathbf{x}_i))u_e(\mathbf{x}_i), \quad (54)$$

where $\Gamma(\boldsymbol{x}) \in [0, 1]$ has to be a single-valued function for $\boldsymbol{x} \in R^d$, where d is the number of horizontal dimensions, and u_e is an analytical solution. The first term acts as a "sponge layer" which will effectively damp any wave motion in the zone. The last terms containing u_e act as "source terms" within the relaxation zone, and if they are included in the relaxation, allow us to generate waves.

This relaxation method is somehow ad-hoc and the following parameters should be balanced before arriving at satisfactory results: a) shape of relaxation functions, b) position of relaxation zones and c) the length of the relaxation zones.

The following relaxation functions have been used in the simulations for respectively sponge layers and wave generation

$$\Gamma(x; p) = 1 - x^p, \quad (55)$$

$$\Gamma(x) = -2x^3 + 3x^2. \quad (56)$$

The relaxation zones are positioned appropriately where waves are to be both/either generated and/or absorbed and the rule of thumb is that the relaxation should be invoked in a region covering approx. 1-2 wave lengths of the primary wave in the region. A disadvantage of the relaxation method is that it takes up a part of the computational domain, but the compensation lies in the fact that it is a straightforward procedure, which is both easy and flexible to use. For this reason it is our method of choice.

5. Computational examples

We conclude with some numerical examples confirming that the proposed methodology provide a basis for solving the high-order Boussinesq-type equations accurately.

5.1. SPATIAL ACCURACY

Consistency tests of the numerical model based on the Padé (4,4) formulation have been carried out to numerically determine convergence rates for h - and p -type refinement strategies. The linearized system of equations in 1D is used as the basis for modelling a linear standing wave in a finite domain $x \in [-1, 1]^2$. The tests are carried out in deep waters with a relative depth corresponding to $kd = 2\pi$, which should be solved accurately by the high-order system, since at this depth all the spatial derivatives are important for obtaining accurate solutions.

It is shown numerically that in the approximate limits, which is reached by either h - or p -type refinement or a combination hereof,

where sufficient spatial resolution is used to capture the solution, the numerical model has the expected spatial order of accuracy, which is obtained by using central numerical fluxes.

Consistency tests have been carried out for a linear standing wave given by the initial surface elevation $\eta(x, t) = H \cos(\omega t) \cos(kx)$ and no kinetic energy, where the angular frequency ω has been determined using the linear Boussinesq dispersion relation derived in Madsen, Bingham and Liu (2002). The numerical model was run for 100 time steps with a time step small enough for the spatial errors to dominate in each test. The normalized max-norm errors have been determined using the analytical solution for a linearly standing wave.

As demonstrated in Figure 3 convergence is obtained for both h -type and p -type strategies until the levels of truncation errors from the solution of the algebraic system dominates. Clearly, the p -type convergence strategy is superior giving rise to exponentially fast convergence as opposed to the alternative h -type convergence strategy leading to algebraic convergence. The spatial order of accuracy is found from the data presented in Figure 3 a) to be optimal $O(h^{N+1})$ for even polynomial orders and sub-optimal $O(h^N)$ for odd polynomial orders.

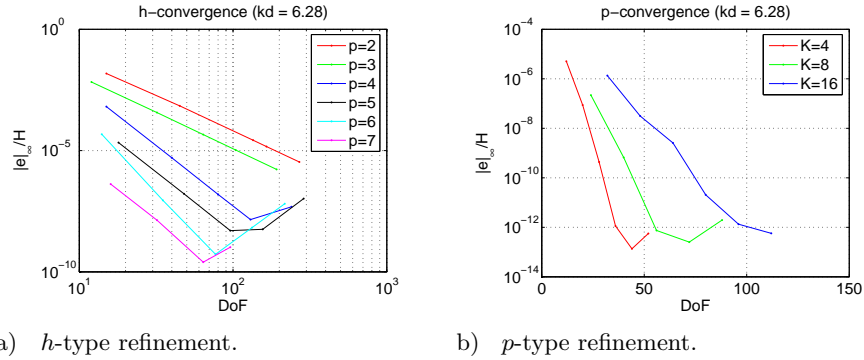


Figure 3. Consistency test for different convergence strategies.

5.2. HARMONIC GENERATION OVER A SUBMERGED BAR

Although potential benefits in using DG-FEM for solving the Boussinesq-type equations are expected to be more significant in solving 2D problems, we demonstrate the geometric flexibility and the potential benefits in using the proposed methodology by simulating harmonic generation over a submerged bar.

The experiment where an input wave is propagating over a submerged bar during which the wave will undergo transformation was originally proposed by Beji and Battjes (1994). Later an equivalent ex-

periment, which was scaled by a factor of two, was carried out by Luth, Klopman and Kitou (1994). The bathymetry was defined as shown in Figure 4 for the original unscaled experiment.

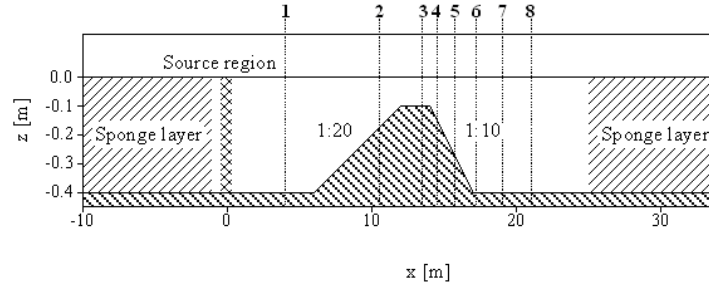


Figure 4. Submerged bar test setup in original experiment.

In this experiment the input wave propagates toward the bar on a constant bottom with no change in form until it reaches the bar. On the up-slope the wave will then steepen due to nonlinear shoaling. At the top of the bar bound harmonics will be released as free harmonics decomposing the wave into shorter waves which will result in a rapidly varying wave profile behind the bar. The shorter wave components or free higher harmonics require a highly accurate dispersion relation to be propagated correctly. Therefore this experiment has been used as a benchmark test for dispersive wave models. See for example Madsen and Schäffer (1999), Madsen, Bingham and Liu (2002) and Walkley (1999).

We have considered a setup corresponding to Case A in the original experiment by Beji and Battjes. The model based on the high-order formulation by Madsen, Bingham and Liu (2002) was used. For this case, the input wave on a undisturbed depth of $d = 0.40$ m is given as; Case A: $T = 2.02$ s and $H = 2$ cm. The input wave was generated using 2nd order Stokes Theory. Computed results are compared with the experimental data obtained by Luth *et al.* in Figure 5. Locally on each element a polynomial order of $N = 8$ is used and $K = 110$ equi-sized elements. At every stage of the explicit Runge-Kutta scheme the spatial solutions for $\eta(x, t)$ and $U(x, t)$ are filtered using a mild nodal filter with $\alpha = 0.1$. As shown in Figure 5 we achieve excellent agreement with the data with minimal differences in surface elevations (due to the accurate dispersion relation) at all gauge locations.

A numerical simulation has been done using an unstructured grid adapted to the problem as depicted in Figure 6 a). Inside the domain, the grid varies linearly from a element size of $h_0 = 0.67$ m to an element size of $h_k = 0.35$ m positioned at the top of the bar. From the top of the bar and forward the element size remain constant to be slightly

increased in the relaxation region where it is instead $h_k = 0.4\text{m}$. Using a local polynomial order of 8 and a total of 80 elements the degrees of freedom are reduced by 26% compared to the former experiment. This is possible because the spatial resolution is coarsened in the region to the left of the bar and fine in the region where the spurious (or faster) wave components are released after the bar. As depicted in Figure 6 the results are indistinguishable from the ones presented in Figure 5. This test serve to illustrate the potential gain that can be obtained by using an adapted grid compared to a structured grid.

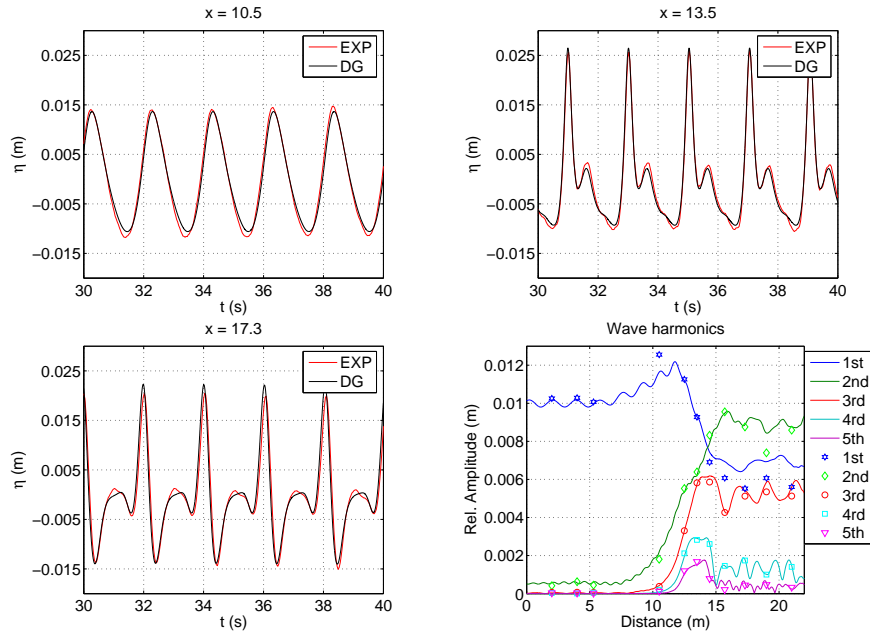


Figure 5. Computed and measured time series of free surface elevations for case A at three gauge positions; *before*, *at*, and *after* the top of the submerged bar. A *structured* grid was used. Last figure shows a comparison by a time harmonics analysis between computed and experimental data.

5.3. HIGH-AMPLITUDE SOLITARY WAVES

A demanding test is to model the propagation of high-amplitude solitary waves which is reflected on a vertical solid wall. This experiment demands a model capable of capturing both the nonlinear and dispersive characteristics. A highly accurate initial condition for the solitary waves can be computed using the method by Tanaka (1986). From the initial starting point, which is assumed to be sufficiently far from the walls for the boundary conditions to be satisfied, the wave will propagate inside the computational domain toward the left wall. Within

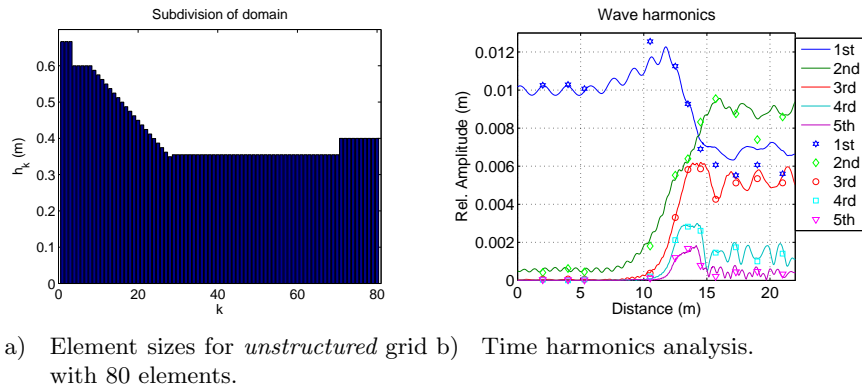


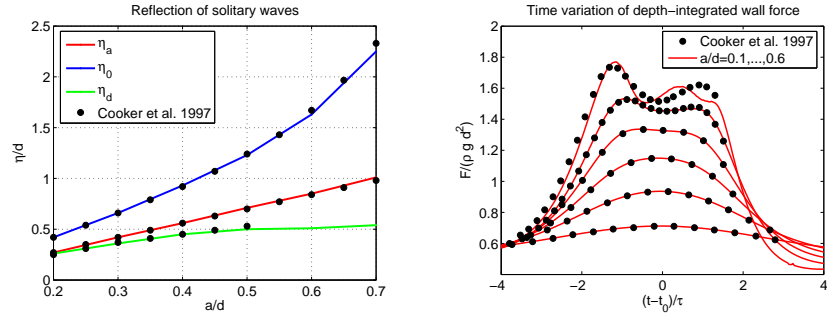
Figure 6. Submerged bar test using an *unstructured* grid adapted to the physical scales.

a distance of approx. $2d$ from the wall measured from the wave crest, the wave crest will start to accelerate leading to a tall and thin yet shooting up in close proximity to the wall. This effect is a pure nonlinear effect. As concluded by Cooker, Weidman and Bale (1997) on the basis of experimental data and in agreement with their numerical results, the phase-change during the reflection is spatially dependent and hence these waves are not solitons, which result in a (usually minor) loss of energy during the reflection of the solitary waves. Therefore the wave cannot maintain its original shape after the reflection. However, the wave will after the reflection again take the form of a solitary wave with slightly smaller amplitude and this adjustment will result in a dispersive trail.

Similar to the results obtained by Madsen, Bingham and Liu (2002), we present results for the attachment, detachment and max wave run up amplitudes. Numerical simulations using *structured* grids have been carried out for solitary waves with varying wave steepness. For solitary waves with amplitudes $a/d \leq 0.5$ a mild nodal filter with $\alpha = 0.1 - 0.2$ have been applied every time step. For the more extreme and steep solitary waves the SG-polynomial averaging (nodal) filter has been applied every time step with stencil half size $\alpha = 6$ and filter order $p = 8$. In all simulations a local polynomial order of $N = 8$ has been used. The spatial resolution needed, increase with the magnitude of the solitary wave amplitude. The number of elements ranged from $K = 40 - 100$ for amplitudes $0.1 - 0.6$ for the waves to be considered as well-resolved in a computational domain $x \in [0, 45]m$. For the high-amplitude solitary waves to be accurately resolved a model capable of adapting the grid "on the fly" to allow for changes in the spatial resolution where it is

needed would be more appropriate in terms of computational efficiency. However, this is not possible in the current application.

As experienced by both Cooker, Weidman and Bale (1997) and Madsen, Bingham and Liu (2002) it is difficult to model the thin jet shooting up numerically due to the rapid and subsequent collapse of the jet afterward, which demands sufficient spatial resolution in this region. However, we introduced a small amount of artificial viscosity in the vicinity of the wall in order to suppress spurious oscillations and therefore we are able to continue the detachment curve where previous results have not been obtained in Figure 7 a) even for the high-amplitude cases. Further, as can be seen in Figure 7 b) the numerical data for the non-dimensional wall forces at the left wall are in excellent agreement up to nearly $a/d = 0.6$. Around this limit it is found necessary to introduce the beforehand mentioned artificial viscosity.



a) Attachment, detachment and max wave run up curves. b) Computed wall forces at the left wall during the reflection.

Figure 7. Reflection of high-amplitude solitary waves.

6. Summary & concluding remarks

We have presented a methodology based on the DG-FEM for solving high-order Boussinesq-type equations, which supports the use of unstructured meshes. This has potential benefits in terms of possible reductions of the degrees of freedom and local adaptivity in complex geometries.

It has been shown that the methodology can be used to produce accurate results for demanding test cases describing relevant dispersive and nonlinear wave phenomena.

Reducing the computational effort remains a future task. The computational bottleneck in solving the system of equations is to both reconstruct and solve the involved algebraic system efficiently. Hence,

it is crucial for large-scale applications that an efficient sparse solver is used to enable the model to be cost-efficient. Currently, the methodology is based on simple approximations in a collocation manner possibly leading to aliasing errors. These aliasing errors can be removed by introducing frequency-dependent artificial viscosity by e.g. nodal filtering or Savitzky-Golay polynomial averaging filtering.

The current application is only capable of adapting the grid to the initial geometry of the domain. For some problems, e.g. high-amplitude solitary waves problems, it is more convenient to be able to adapt the grid during the simulation based on for example steep gradients in the solution.

We seek to test the methodology for fully nonlinear problems in two horizontal dimensions and a thorough description along with more results will appear in a future paper.

Appendix

A. Differential operators

A number of differential operators are derived for the high-order Boussinesq type formulation presented in section 2. The definitions for use with the one and two dimensional formulations are given.

A.1. TWO DIFFERENT FORMULATIONS

Two different formulations have been used in setting up the numerical models.

The Padé (4,4) formulation retaining up to fifth order derivatives given in Madsen, Bingham and Liu (2002) result in the following coefficients

$$\alpha_2 \equiv \frac{(z - \hat{z})^2}{2} - \frac{\hat{z}^2}{18}, \quad (57)$$

$$\alpha_4 \equiv \frac{(z - \hat{z})^4}{24} - \frac{\hat{z}^2(z - \hat{z})^2}{36} + \frac{\hat{z}^4}{504}, \quad (58)$$

$$\beta_1 \equiv z - \hat{z}, \quad (59)$$

$$\beta_3 \equiv \frac{(z - \hat{z})^3}{6} - \frac{\hat{z}^2(z - \hat{z})}{18}, \quad (60)$$

$$\beta_5 \equiv \frac{(z - \hat{z})^5}{120} - \frac{\hat{z}^2(z - \hat{z})^3}{108} + \frac{\hat{z}^4(z - \hat{z})}{504}, \quad (61)$$

$$\gamma_2 \equiv \frac{4}{9}, \quad \gamma_3 \equiv \frac{1}{9}, \quad \gamma_4 \equiv \frac{1}{63}, \quad \gamma_5 \equiv \frac{1}{945}. \quad (62)$$

The optimized parameters leading to a minimization of the linear shoaling errors over $kd \in [0, 30]$ ($\sigma = -0.5$) are given as

$$\begin{aligned} c_2 &= 0.357739, & c_4 &= 0.00663819, \\ s_3 &= 0.0753019, & s_5 &= -6.31532x10^{-5}. \end{aligned} \quad (63)$$

The Padé (2,2) formulation retaining up to third order derivatives is given as

$$\alpha_2 \equiv \frac{(z - \hat{z})^2}{2} - \frac{\hat{z}^2}{10}, \quad (64)$$

$$\alpha_4 \equiv 0, \quad (65)$$

$$\beta_1 \equiv z - \hat{z}, \quad (66)$$

$$\beta_3 \equiv \frac{(z - \hat{z})^3}{6} - \frac{\hat{z}^2(z - \hat{z})}{10}, \quad (67)$$

$$\beta_5 \equiv 0, \quad (68)$$

$$\gamma_2 \equiv \frac{2}{5}, \quad \gamma_3 \equiv \frac{1}{15}, \quad \gamma_4 \equiv 0, \quad \gamma_5 \equiv 0. \quad (69)$$

The optimized parameters leading to a minimization of the linear shoaling errors over $kd \in [0, 6]$ ($\sigma = -0.5$) presented in Jamois, Fuhrman and Bingham (2005) are given as

$$c_2 = -0.0593982, \quad s_3 = -0.00113222. \quad (70)$$

Further, the height of the water-column from the bottom to the chosen expansion level is defined as

$$\lambda = d + \hat{z} \quad (71)$$

A.2. ONE DIMENSION

$$A_2 = 1 - \alpha_2(\partial_{xx}) + \alpha_4(\partial_{xxxx}), \quad (72)$$

$$A_3 = \lambda(\partial_x) - \gamma_3\lambda^3(\partial_{xxx}) + \gamma_5\lambda^5(\partial_{xxxxx}), \quad (73)$$

$$B_2 = \beta_1(\partial_x) - \beta_3(\partial_{xxx}) + \beta_5(\partial_{xxxxx}), \quad (74)$$

$$B_3 = 1 - \gamma_2\lambda^2(\partial_{xx}) + \gamma_2\lambda^4(\partial_{xxxx}), \quad (75)$$

$$S_1 = 1 - c_2\lambda^2(\partial_{xx}) + c_4\lambda^4(\partial_{xxxx}), \quad (76)$$

$$S_2 = \lambda(\partial_x) - s_3\lambda^3(\partial_{xxx}) + s_5\lambda^5(\partial_{xxxxx}). \quad (77)$$

A.3. TWO DIMENSIONS

$$A_{01} = \lambda \partial_x - \gamma_3 \lambda^3 (\partial_{xxx} + \partial_{xyy}) + \gamma_5 \lambda^5 (\partial_{xxxx} + 2\partial_{xxyy} + \partial_{yyyy}) \quad (78)$$

$$A_{02} = \lambda \partial_y - \gamma_3 \lambda^3 (\partial_{xxy} + \partial_{yyy}) + \gamma_5 \lambda^5 (\partial_{xxxxy} + 2\partial_{xxyyy} + \partial_{yyyyy}) \quad (79)$$

$$A_1 = 1 - \alpha_2 (\partial_{xx} + \partial_{yy}) + \alpha_4 (\partial_{xxxx} + 2\partial_{xxyy} + \partial_{yyyy}), \quad (80)$$

$$A_{11} = 1 - \alpha_2 (\partial_{xx}) + \alpha_4 (\partial_{xxxx} + \partial_{xxyy}), \quad (81)$$

$$A_2 = -\alpha_2 (\partial_{xy}) + \alpha_4 (\partial_{xxyy} + \partial_{xyyy}), \quad (82)$$

$$A_{22} = 1 - \alpha_2 (\partial_{yy}) + \alpha_4 (\partial_{xxyy} + \partial_{yyyy}), \quad (83)$$

$$B_0 = 1 - \gamma_2 \lambda^2 (\partial_{xx} + \partial_{yy}) + \gamma_4 \lambda^4 (\partial_{xxxx} + 2\partial_{xxyy} + \partial_{yyyy}), \quad (84)$$

$$B_{11} = \beta_1 (\partial_x) - \beta_3 (\partial_{xxx} + \partial_{xyy}) + \beta_5 (\partial_{xxxx} + 2\partial_{xxyy} + \partial_{xyyy}), \quad (85)$$

$$B_{12} = \beta_1 (\partial_y) - \beta_3 (\partial_{xxy} + \partial_{yyy}) + \beta_5 (\partial_{xxxxy} + 2\partial_{xxyyy} + \partial_{yyyyy}), \quad (86)$$

$$C_{11} = 1 - c_2 \lambda^2 (\partial_{xx}) + c_4 \lambda^4 (\partial_{xxxx} + \partial_{xxyy}), \quad (87)$$

$$C_2 = -c_2 \lambda^2 (\partial_{xy}) + c_4 \lambda^4 (\partial_{xxyy} + \partial_{xyyy}), \quad (88)$$

$$C_{13} = \lambda (\partial_x) - s_3 \lambda^3 (\partial_{xxx} + \partial_{xyy}) + s_5 \lambda^5 (\partial_{xxxx} + 2\partial_{xxyy} + \partial_{xyyy}) \quad (89)$$

$$C_{22} = 1 - c_2 \lambda^2 (\partial_{yy}) + c_4 \lambda^4 (\partial_{xxyy} + \partial_{yyyy}), \quad (90)$$

$$C_{23} = \lambda (\partial_y) - s_3 \lambda^3 (\partial_{xxy} + \partial_{yyy}) + s_5 \lambda^5 (\partial_{xxxxy} + 2\partial_{xxyyy} + \partial_{yyyyy}) \quad (91)$$

Acknowledgements

The first author would like to thank Division of Applied mathematics at Brown University (Providence, USA) for an inspiring visit in the period from Sep 2004 to Mar 2005. Further, the helpful discussions with Professor Chi-Wang Shu of Brown University, USA, Claes Eskilsson of Chalmers University of Technology, Sweden, and David Fuhrman of Technical University of Denmark have been highly appreciated.

References

- Ambrosi, D., and Quartapelle, L. A Taylor-Galerkin Method for Simulating Nonlinear Dispersive Water Waves. *J. Comp. Phys.*, Vol. 146, 546-569, 1998.
- Antunes Do Carmo, J. S., and Seabra Santos, F. J. Surface Waves propagation in shallow water: A finite element model. *J. Numer. Meth. Fluids*, Vol. 16, 447-459, 1993.
- Bassi, F., and Rebay, S. A High-Order Accurate Discontinuous Finite Element Method for the Numerical Solution of the Compressible Navier-Stokes Equations. *J. Comp. Phys.*, Vol. 131, 267-279, 1997.
- Beji, S., and Battjes, J. A. Numerical simulation of nonlinear-wave propagation over a bar. *Coastal Engineering*, Vol. 23, 1-16, 1994.

- Beji, S., and Nadaoka, K. A formal derivation and numerical modelling of the improved Boussinesq equations for varying depth. *Ocean Engng.*, Vol. 23, 691-704, 1996.
- Carpenter, M. H., and Kennedy, C. A. Fourth order 2N-storage Runge-Kutta scheme. NASA-TM-109112, NASA Langley Research Center, VA., 1994.
- Cockburn, B., and Shu, C.-W. Runge-Kutta Discontinuous Galerkin Methods for Convection-Dominated Problems. *J. Scientific Computing*, Vol. 16, No. 3, 173-261, 2001.
- Cooker, M.J., Weidman, P.D., and Bale, D.S. Reflection of a high-amplitude solitary wave at a vertical wall. *J. Fluid Mech.*, Vol. 342, 141-158, 1997.
- Eskilsson, C., and Sherwin, S. J. A Discontinuous Spectral Element Model for Boussinesq-Type Equations. *J. Scient. Comp.*, Vol. 17, 143-152, 2002.
- Eskilsson, C., and Sherwin, S. J. An *hp*/spectral element model for efficient long-time integration of Boussinesq-type equations. *Coastal Engineering Journal*, Vol. 45, 295-320, 2003.
- Eskilsson, C., and Sherwin, S. J. An unstructured spectral/*hp* element model for enhanced Boussinesq-type equations. *submitted to Coastal Engineering Journal*, 2005.
- Eskilsson, C., and Sherwin, S. J. Spectral/*hp* discontinuous Galerkin methods for modelling 2D Boussinesq equations. *J. Comp. Phys.*, *submitted*, 2004.
- Eskilsson, C., and Sherwin, S. J. Discontinuous Galerkin spectral/*hp* element modelling of dispersive shallow water systems. *J. Scient. Comp.*, 22, 279-298, 2005.
- Fischer, P., and Mullen, J. S. Filter-based Stabilization of Spectral Element Methods. *C. R. Acad. Sci. Paris*, 332, 265-270, 2001.
- Fuhrman, D. R., Bingham, H. B., Madsen, P. A., and Thomsen, P. G. Linear and nonlinear stability analysis of high-order Boussinesq equations. *Int. J. Numer. Meth. Fluids*, 2000.
- Fuhrman, D. R., Bingham, H. B., and Madsen, P. A. Nonlinear wave-structure interactions with a high-order Boussinesq model. *Coastal engineering*, 2004.
- Hesthaven, J.S. From electrostatics to almost optimal nodal sets for polynomial interpolation in a simplex. *SIAM J. Numer. Anal.*, Vol. 35, No. 2, 655-676, 1998.
- Hesthaven, J.S., and Warburton, T. High-Order Nodal Methods on Unstructured Grids. I. Time-Domain Solution of Maxwell's Equations. *J. Comput. Phys.*, 181(1), 186-221, 2002.
- Jamois, E., Fuhrman, D. R., and Bingham, H. B. Wave-structure interactions and nonlinear wave processes on the weather side of reflective structures. *Submitted to Coastal Engineering*, 2005.
- Karniadakis, G. E., and Sherwin, S. J. Spectral/*hp* Element Methods for CFD. Oxford University Press, 1999.
- Kirby, R. M., and Karniadakis, G. E. Selecting the Numerical Flux in Discontinuous Galerkin Methods for Diffusion problems. Oxford University Press, 2005.
- Larsen, J., and Dancy, H. Open boundaries in short wave simulations - a new approach. *Coastal Engineering*, 7, 285-297, 1983.
- Langtangen, H. P., and Pedersen, G. Computational models for weakly dispersive nonlinear water waves. *Comput. Methods Appl. Engrg.*, 160, 337-358, 1998. rt, Department of Mathematical Sciences, Loughborough University.
- Li, Y.S, Liu, S.-X., Yu, Y.-X., and Lai, G.-Z. Numerical modelling of Boussinesq equations by finite element method. *Coastal engineering*, 37, 97-122, 1999.

- Luth, H. R., Klopman, B., and Kitou, N. Projects 13G: Kinematics of waves breaking partially on an offshore bar: LDV measurements for waves with and without a net onshore current. *Technical report H1573*, Delft Hydraulics, 1994.
- Madsen, P.A., H.B. Bingham, and H. Liu. A new Boussinesq method for fully nonlinear waves from shallow to deep water. *J. Fluid Mech.* 462, 1–30, 2002.
- Madsen, P.A., and Schäffer, H. A. A review of Boussinesq-type equations for gravity waves. *In Advances in Coastal and Ocean Engineering*, vol 5, 1–95, 1999.
- Madsen, P.A., and Sørensen, O. R. A new form of the Boussinesq equations with improved linear dispersion characteristics. Part 2. A slowly varying bathymetry. *Coastal Engineering*, vol 18, 183–204, 1992.
- Madsen, P.A., and Agnon, Y. Accuracy and convergence of velocity formulations for water waves in the framework of Boussinesq theory. *J. Fluid Mech.*, vol 477, 285–319, 2003.
- Nwogu, O. Alternative form of Boussinesq equations for nearshore wave propagation. *Journal of Waterway, Port, Coastal and Ocean Engineering*, 119 (6), 618–638, 1993.
- Peregrine, D. H. Long waves on a beach. *Journal of Fluid Mechanics*, 27, 815–827, 1967.
- Savitzky, A., and Golay, M. J. E. Smoothing and differentiation of data by simplified least squares procedures. *Anal. Chem.*, 36, 1627–1639, 1964.
- Sherwin, S. Dispersion Analysis of the Continuous and Discontinuous Galerkin Formulations. *Lecture Notes in Computational Science and Engineering: Discontinuous Galerkin Methods - Theory, Computation and Applications*, Springer, 1999.
- Sørensen, O. R., Schäffer, H. A., and Sørensen, L. S. Boussinesq-type modelling using an unstructured finite element technique. *Coastal Engineering*, 50, 181–198, 2004.
- Tanaka, M. The stability of solitary waves. *J. Phys. Fluids*, 29, 650–655, 1986.
- Walkley, M. A. A Numerical Method for Extended Boussinesq Shallow-Water Wave Equations. *PhD. Thesis.*, The University of Leedes, School of Computer Studies.
- Walkley, M. A., and Berzins, M. A finite element method for the one-dimensional extended Boussinesq equations. *Int. J. Numer. Meth. Fluids.*, 29, 143–157, 1999.
- Walkley, M. A., and Berzins, M. A finite element method for the two-dimensional extended Boussinesq equations. *Int. J. Numer. Meth. Fluids.*, 39, 865–885, 2002.

