Available online at www.sciencedirect.com
ScienceDirect
Comput. Methods Appl. Mech. Engrg. 196 (2007) 1984–1996
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

A Runge–Kutta discontinuous Galerkin method for linear free-surface gravity waves using high order velocity recovery

S.K. Tomar [a,*], J.J.W. van der Vegt [b]

[a] Johann Radon Institute for Computational and Applied Mathematics (RICAM), Austrian Academy of Sciences, Altenbergerstrasse 69, A-4040 Linz, Austria
[b] Department of Applied Mathematics, University of Twente, P.O. Box 217, 7500AE Enschede, The Netherlands

Received 23 February 2006; received in revised form 30 October 2006; accepted 1 November 2006

Abstract

We present a higher order accurate discontinuous Galerkin finite element method for the simulation of linear free-surface gravity waves. The method uses the classical Runge–Kutta method for the time-discretization of the free-surface equations and the discontinuous Galerkin method for the space-discretization. In order to circumvent numerical instabilities arising from an asymmetric mesh a stabilization term is added to the free-surface equations. In combination with a higher order velocity recovery technique this stabilizes the numerical discretization with minimal effect on the accuracy of the wave computations. A stability analysis of the semi and fully-discrete scheme is presented, which suggests that for a suitable choice of the stabilization constant a relatively large time step can be chosen for accurate simulations over a long period of time. Numerical examples of a number of problems are also presented.

Keywords: Discontinuous Galerkin method; Gravity waves; Free-surface; High order gradient recovery

1. Introduction

Numerical computations of free-surface waves require a higher order accurate discretization, both in space and time, in order to obtain accurate solutions with minimal dispersion and dissipation errors, in particular for long time simulations. For wave problems in complex domains, finite element methods provide an attractive numerical technique to achieve high order accuracy, in particular for nonlinear waves. Also, finite element methods result in a sparse matrix which can be efficiently solved with fast iterative solvers. Recently, discontinuous Galerkin (DG) finite element methods have become very popular due to their suitability for mesh adaptation, using local mesh refinement and coarsening or the adjustment of the polynomial order of the basis functions, while maintaining high order accuracy on unstructured meshes. In this paper we further investigate their suitability for the simulation of wave problems.

In [9] we have proposed an unconditionally stable DG method for the numerical simulation of linear free-surface gravity waves. A key feature of this numerical algorithm is the introduction of the free-surface boundary condition directly into the weak formulation, which eliminates a separate computation of the velocity field. This DG algorithm is unconditionally stable and does not suffer from a weak instability originating from the free-surface boundary condition when a non-uniform mesh is used in the interior domain, as occurs for instance in the higher order spectral method proposed in [8].

The algorithm in [9] is higher order accurate in space, but only second order accurate in time. When the polynomial order of the finite element basis functions is two or higher then a severe time step limitation occurs if one wants to have the same accuracy in time as in space. Also, a

* Corresponding author. Tel.: +43 732 2468 5220; fax: +43 732 2468 5212.
E-mail addresses: satyendra.tomar@ricam.oeaw.ac.at (S.K. Tomar), j.j.w.vandervegt@math.utwente.nl (J.J.W. van der Vegt).

0045-7825/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.cma.2006.11.007

straightforward extension of this algorithm to nonlinear problems cannot be accomplished by completely eliminating the velocity field.

In this paper an alternative DG method is presented where we treat the free-surface boundary conditions separately, i.e. they are not included into the weak formulation. In this approach, once the velocity potential is computed in the domain, the time dependent free-surface equations can be discretized using high order explicit schemes, e.g. the classical Runge–Kutta method. We combine this technique with an L^2 projection based high order velocity recovery technique [7], which ensures that the velocity field has the same order of accuracy as the velocity potential. This approach results in a uniformly higher order accurate finite element discretization, which we demonstrate for linear free-surface waves in an inviscid incompressible fluid. The study of this model problem is an essential step in the development of a finite element method for nonlinear waves.

Unfortunately, the algorithm discussed in this paper is not unconditionally stable on non-uniform meshes. To overcome this, as suggested in [8], we add a stabilization term to the free-surface boundary condition. Using Fourier analysis, we will demonstrate that in combination with the higher order velocity recovery technique this has minimal effect on the wave properties. Also, for the same level of accuracy the presented algorithm is more efficient than the approach discussed in [9], which was only second order accurate in time and requires a significantly smaller time step for a comparable accuracy as the algorithm discussed in this article.

Furthermore, the core of the approach presented here, the weak formulation of the Laplace equation in the fluid domain, will not have to be altered for the fully nonlinear problem, only the discretization of the free-surface boundary conditions has to be changed. Thus, the system of equations resulting from the weak formulation for the velocity potential will preserve the symmetry and positive-definiteness for the fully nonlinear problem, which can therefore be solved efficiently by many fast iterative solvers.

We now briefly summarize the contents of this paper. In Section 2 we present the governing equations, tessellation, and function spaces followed by the trace and lifting operators. In Section 3 we discuss the discontinuous Galerkin method and present the weak formulation. Section 4 presents the discrete form of the weak formulation for the potential and velocity field. This is followed by Section 5 where the discrete formulation of the linear free-surface equations is presented and the time-discretization scheme is discussed. Some of the matrix notations introduced here will be needed in the stability analysis. In Section 6 we first present the semi-discrete stability analysis of the linear problem, which provides the dispersion and dissipation properties of the scheme. This is followed by a fully-discrete stability analysis of the classical Runge–Kutta scheme, giving an estimate for the maximum time step necessary for a stable scheme. Numerical results are presented in Section 7, followed by the conclusions in Section 8. Finally, further algorithmic details are given in the appendix.

2. Preliminaries

2.1. Governing equations

Let us assume the fluid to be incompressible, inviscid, with the velocity field irrotational. Let (x,z) denote the regular Cartesian coordinate system in $\mathbb{R}^d$, with $d=2$ or 3. In case $d=3$, x will represent (x,y) wherever required. We consider the equations for linear free-surface waves in a domain $\Omega \subset \mathbb{R}^d$. The boundary is split into two parts Γ_D and Γ_N with $\Gamma_D \cap \Gamma_N = \emptyset$. Γ_D represents the boundary where the free-surface conditions are applied. The surface Γ_D is taken as (part of) the $z=0$ plane for convenience. The following equations then govern the motion of linear free-surface waves [9,8,11,12]

$$-\Delta\phi = 0 \quad \text{in } \Omega, \tag{2.1a}$$

$$\nabla\phi \cdot n = g_N \quad \text{at } \Gamma_N, \tag{2.1b}$$

$$\frac{\partial\phi}{\partial t} = -\zeta \quad \text{at } z=0, \tag{2.1c}$$

$$\frac{\partial\zeta}{\partial t} = \frac{\partial\phi}{\partial z} \quad \text{at } z=0, \tag{2.1d}$$

where ϕ denotes the velocity potential, ζ the wave height, t time, $n \in \mathbb{R}^d$ the unit outward normal vector to $\partial\Omega$, and g_N the normal velocity at solid surfaces. Eqs. (2.1a)–(2.1d) are non-dimensionalized using the transformations

$$\phi \to H(g_c H)^{1/2}\phi, \quad (x,z) \to H(x,z), \quad \zeta \to H\zeta, \quad \text{and}$$
$$t \to (H/g_c)^{1/2} t,$$

where H represents the average water depth and g_c the gravitational constant.

Now, let us assume that at a given instance in time, say t_n, the potential on the free-surface boundary Γ_D be equal to ϕ^D. We then obtain the velocity potential at the next time level by solving the following boundary value problem:

$$-\Delta\phi = 0 \quad \text{in } \Omega, \quad \phi = \phi^D \quad \text{at } \Gamma_D,$$
$$\nabla\phi \cdot n = g_N \quad \text{at } \Gamma_N. \tag{2.2}$$

Using (2.1c) and (2.1d), the velocity potential thus obtained will be updated to give ϕ^D for the next time level. An example of a flow domain is given in Fig. 1, where a model basin with a wave-maker on the left side and solid walls on bottom and right sides are depicted. As discussed in Section 7.2, at the wave-maker a time periodic Neumann boundary condition is applied.

2.2. Tessellation and function spaces

Let $\mathscr{T}_h$ be the tessellation of Ω and N_e the total number of elements in the mesh at a given time t_n. We denote the

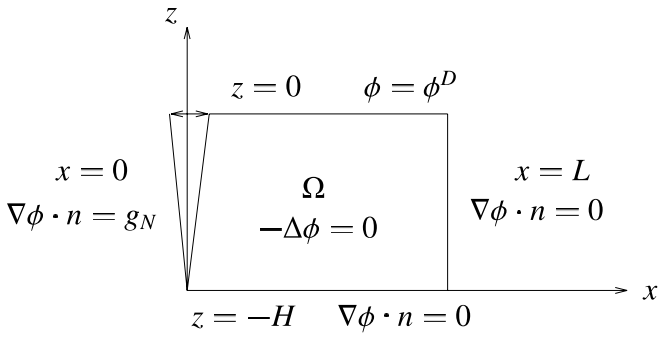

Fig. 1. Example of a flow domain for wave computations.

elements of $\mathscr{T}_h$ by K, with $h = \max_{K\in\mathscr{T}_h}\mathrm{diam}(K)$, the union of the faces of the elements K of $\mathscr{T}_h$ by Γ, i.e. $\Gamma = \bigcup_{K=1}^{N_e}\partial K$, and the union of the internal faces by $\Gamma_0 = \Gamma\backslash\partial\Omega$. We further denote the set of all faces of $\mathscr{T}_h$ by $\mathscr{F}^A$, the set of internal faces by $\mathscr{F}^I$, the set of faces on Γ_D by $\mathscr{F}^D$, the set of faces on Γ_N by $\mathscr{F}^N$, and the set of faces on $\partial\Omega$ by $\mathscr{F}^\partial$.

We define the usual Sobolev spaces as follows:

$$H^1(\Omega) := \{\psi \in L^2(\Omega) : \mathscr{D}\psi \in L^2(\Omega)\},$$

where $\mathscr{D}$ denotes the distributional derivative and $L^2(\Omega)$ the space of square integrable functions on Ω. In addition, we introduce the broken Sobolev spaces $H^1(\mathscr{T}_h) = \prod_{K\in\mathscr{T}_h}H^1(K)$. The *traces* of functions in $H^1(\mathscr{T}_h)$ belong to $T(\Gamma) := \prod_{K\in\mathscr{T}_h}H^{1/2}(\partial K)$. Functions in $T(\Gamma)$ are thus double-valued on Γ_0 and single-valued on $\partial\Omega$.

Next, we introduce the finite element spaces associated with the tessellation $\mathscr{T}_h$ of the domain Ω. We assume the elements K to be shape-regular [3]. We set

$$\Psi_h := \{\psi \in L^2(\Omega) : \psi|_K \in P(K), \forall K \in \mathscr{T}_h\},$$
$$V_h := \{v \in [L^2(\Omega)]^d : v|_K \in V(K) \text{ and } \nabla\psi|_K \in V(K), \forall\psi \in \Psi_h, \forall K \in \mathscr{T}_h\},$$

where $P(K) = \mathscr{P}_p(K)$ is the space of polynomial functions of degree at most $p \geqslant 1$ on K and $V(K) = [\mathscr{P}_p(K)]^d$. Note that V_h additionally contains the gradients of the functions in Ψ_h.

2.3. Trace and lifting operators

To deal with multivalued traces at the element boundary faces in a DG discretization we introduce some trace operators to manipulate the numerical fluxes and to define the primal DG formulation. For $\varphi \in T(\Gamma)$, we define the *average* $\{\varphi\}$ and *jump* $[\![\varphi]\!]$ of φ as follows:

Let $\mathscr{F}$ be an interior face shared by elements L and R. Define the unit normal vectors n_L and n_R on $\mathscr{F}$ pointing exterior to L and R, respectively. With $\varphi_{L/R} := \varphi|_{\partial K_{L/R}}$ we set

$$\{\varphi\} = \frac{1}{2}(\varphi_L + \varphi_R), \quad [\![\varphi]\!] = \varphi_L n_L + \varphi_R n_R \quad \text{on } \mathscr{F} \in \mathscr{F}^I.$$

For $q \in [T(\Gamma)]^d$ we define q_L and q_R analogously, and set

$$\{q\} = \frac{1}{2}(q_L + q_R), \quad [\![q]\!] = q_L \cdot n_L + q_R \cdot n_R \quad \text{on } \mathscr{F} \in \mathscr{F}^I.$$

For $\mathscr{F} \in \mathscr{F}^\partial$ each $\varphi \in T(\Gamma)$ and $q \in [T(\Gamma)]^d$ have a uniquely defined restriction on $\mathscr{F}$; we set

$$[\![\varphi]\!] = \varphi n, \quad \{q\} = q \quad \text{on } \mathscr{F} \in \mathscr{F}^\partial,$$

where n is the outward unit normal at $\partial\Omega$. Since we do not require either of the quantities $\{\varphi\}$ or $[\![q]\!]$ on boundary faces, we leave them undefined.

In the sequel we will also need the lifting operator $\mathscr{R}_0 : [L^2(\Gamma_0 \cup \Gamma_D)]^d \to V_h$, defined by

$$\int_\Omega \mathscr{R}_0(q) \cdot v \,\mathrm{d}x = \int_{\Gamma_0\cup\Gamma_D} q \cdot \{v\} \,\mathrm{d}s, \quad \forall v \in V_h. \tag{2.3}$$

Further, to get the most compact stencil in the discretization matrix we also need a *local* lifting operator $\mathscr{R}_\mathscr{F} : [L^2(\mathscr{F})]^d \to V_h$, where $\mathscr{F} \subset \Gamma_0 \cup \Gamma_D$, defined by

$$\int_\Omega \mathscr{R}_\mathscr{F}(q) \cdot v \,\mathrm{d}x = \int_\mathscr{F} q \cdot \{v\} \,\mathrm{d}s, \quad \forall v \in V_h. \tag{2.4}$$

Note that $\mathscr{R}_\mathscr{F}(q)$ vanishes outside the union of the one or two elements connected to $\mathscr{F}$ and that $\mathscr{R}_0(q) = \sum_{\mathscr{F}\in\mathscr{F}^I+\mathscr{F}^D}\mathscr{R}_\mathscr{F}(q)$ for all $q \in [L^2(\Gamma)]^d$.

3. Discontinuous Galerkin method

To solve the boundary value problem (2.2) for the velocity potential and the velocity field using a discontinuous Galerkin method we introduce the velocity field $u = \nabla\phi$ as an auxiliary variable into our system. Eqs. (2.1a) and (2.1b) can now be rewritten as

$$u - \nabla\phi = 0 \quad \text{in } \Omega, \quad -\nabla\cdot u = 0 \quad \text{in } \Omega, \tag{3.1a}$$
$$\phi = \phi^D \quad \text{at } \Gamma_D, \quad u\cdot n = g_N \quad \text{at } \Gamma_N. \tag{3.1b}$$

Using the DG discretization technique proposed by Brezzi et al. [4,5] the primal DG formulation for the potential ϕ_h can now be formulated as:

Find a $\phi_h \in \Psi_h$ such that for all $\psi_h \in \Psi_h$ the following relation holds:

$$B_h(\phi_h, \psi_h) = L_h(\psi_h), \tag{3.2a}$$

where the bilinear form $B_h(\phi_h, \psi_h) : \Psi_h \times \Psi_h \to \mathbb{R}$ and the linear form $L_h(\psi_h) : \Psi_h \to \mathbb{R}$ are defined as

$$\begin{aligned} B_h(\phi_h, \psi_h) := {} & \int_\Omega \nabla_h\phi_h \cdot \nabla_h\psi_h \,\mathrm{d}x \\ & - \int_{\Gamma_0\cup\Gamma_D} ([\![\phi_h]\!] \cdot \{\nabla_h\psi_h\} + \{\nabla_h\phi_h\} \cdot [\![\psi_h]\!]) \,\mathrm{d}s \\ & + \eta_\mathscr{F} \sum_{\mathscr{F}\in\mathscr{F}^I+\mathscr{F}^D} \int_\Omega \mathscr{R}_\mathscr{F}([\![\phi_h]\!]) \cdot \mathscr{R}_\mathscr{F}([\![\psi_h]\!]) \,\mathrm{d}x, \end{aligned} \tag{3.2b}$$

$$\begin{aligned} L_h(\psi_h) := {} & \int_{\Gamma_D} \phi^D n \cdot \nabla_h\psi_h \,\mathrm{d}s \\ & + \eta_\mathscr{F} \sum_{\mathscr{F}\in\mathscr{F}^D} \int_\mathscr{F} \phi^D n \cdot \mathscr{R}_\mathscr{F}([\![\psi_h]\!]) \,\mathrm{d}s + \int_{\Gamma_N} g_N\psi_h \,\mathrm{d}s. \end{aligned}$$

The parameter $\eta_\mathscr{F}$ must be chosen such that $\eta_\mathscr{F} > n_f$, the number of faces of an element. For the derivation the read-

er is referred to [9,1]. For the discretization of (3.1a) and (3.1b) one can also use other stable and consistent/adjoint-consistent DG methods as listed in [1], for example Bassi et al. [2] or the local discontinuous Galerkin method by Cockburn and Shu [6].

Furthermore, the following relation gives the velocity field $u_h \in V_h$:

$$u_h = \nabla_h \phi_h + \mathscr{R}_0([\![\hat{\phi} - \phi_h]\!]), \tag{3.3}$$

with the lifting operator $\mathscr{R}_0$ defined in (2.3).

4. Discrete formulation

The weak formulation for the Laplace equation (3.2) is transformed into a set of algebraic equations by introducing the polynomial expansion for ϕ_h and ψ_h in each element K, given by

$$\phi_{K,h} = \sum_{j=0}^{p_K} \tilde{\phi}_{K,j} \mathscr{N}_{K,j}(x), \quad \psi_{K,h} = \sum_{j=0}^{p_K} \tilde{\psi}_{K,j} \mathscr{N}_{K,j}(x), \quad x \in K \subset \mathbb{R}^d, \tag{4.1}$$

into the weak formulation equation (3.2). Here $\tilde{\phi}_{K,j} \in \mathbb{R}^{p_K+1}$ and $\tilde{\psi}_{K,j} \in \mathbb{R}^{p_K+1}$ are the expansion coefficients of the potential ϕ_h and the test function ψ_h in the element K, respectively, and $\mathscr{N}_{K,j}$ the $p_K + 1$ polynomial basis functions of order p_K, which can vary in each element.

After some lengthy algebra the linear problem can be represented in matrix form as

$$A\phi = X, \quad \text{with } A \in \mathbb{R}^{N\times N}, \quad \phi, X \in \mathbb{R}^N, \tag{4.2}$$

where $N = \sum_{K=1}^{N_e}(p_K + 1)$ is the total number of degrees of freedom in the discretization. The details of the construction of the matrix A are given in Appendix A.2. The matrix A, consisting N_e^2 blocks $[A] \in \mathbb{R}^{(p_K+1)\times(p_K+1)}$, is a sparse symmetric positive definite matrix since the bilinear form (3.2b) is symmetric and coercive on $H^1(\mathscr{T}_h)$, for a proof see e.g. [1]. Further, the vectors ϕ, X consist of blocks $[\phi], [X] \in \mathbb{R}^{(p_K+1)}$, respectively. The linear system can be solved straight-forwardly with either a sparse direct method or a preconditioned conjugate gradient method. For the update of the linear free-surface boundary conditions we do not need, however, the potential in the whole domain, only the gradient in the z-direction at the free-surface is needed. This is particularly useful in the stability analysis discussed in Section 6. In Appendix A.3 we discuss the details of the construction of the discrete Dirichlet to Neumann operator which maps the discrete free-surface potential ϕ_h^D to the vertical velocity component of u_h at the free-surface.

4.1. Higher order accurate velocity recovery

The free-surface boundary conditions (2.1c) and (2.1d) require the vertical velocity at the free-surface. Using the discretization for the Laplace equation (4.2) with a given free-surface potential ϕ^D, we can compute this velocity using (3.3). In this section we describe the DG algorithm to compute the velocity field. Since the velocity field depends on the gradient of the potential ϕ_h, we loose one order of accuracy. This would also affect the accuracy of the wave computations. In the second part of this section we describe therefore a higher order velocity recovery technique which improves the accuracy in the velocity field and also the stability of the numerical scheme, the latter is demonstrated in Section 6.

We express the velocity field in an element K as

$$u_{K,k_d,h} = \sum_{j=0}^{p_K} \tilde{u}_{K,k_d,j} \mathscr{N}_{K,j}(x), \quad x \in K, \tag{4.3}$$

where $\tilde{u}_{K,k_d,j}$ are the expansion coefficients for the k_dth component of $u_{K,h}$. We now define the matrices $D^K \in \mathbb{R}^{p_K\times p_K}$, $E^{LR} \in \mathbb{R}^{p_L\times p_R\times d}$ and $M^K \in \mathbb{R}^{p_K\times p_K\times d}$ as

$$D_{ij}^K = \int_K \mathscr{N}_{K,i}\mathscr{N}_{K,j}\,\mathrm{d}x, \quad E_{ijk_d}^{LR} = \int_{\mathscr{F}} \mathscr{N}_{L,i}\mathscr{N}_{R,j} n_{R,k_d}\,\mathrm{d}s,$$
$$M_{ijk_d}^K = \int_K \mathscr{N}_{K,i} \frac{\partial \mathscr{N}_{K,j}}{\partial x_{k_d}}\,\mathrm{d}x \tag{4.4}$$

$\forall k_d = 1,\ldots,d$. Here L and R denote the elements connected to the face $\mathscr{F}$ and n_{K,k_d} is the k_dth component of the unit outward normal vector at the boundary of element K. Substituting the approximation (4.3) into (3.3), multiplying with arbitrary test functions $v_h \in V_h$, integrating over element K and finally using the relation for the lifting operator given in the appendix, we obtain the following equation for the expansion coefficients of the velocity field in element K:

$$\sum_{j=0}^{p_K} \tilde{u}_{K,k_d,j} D_{ij}^K = \sum_{j=0}^{p_K} \tilde{\phi}_{K,j} M_{ijk_d}^K + \sum_{\mathscr{F}\in\mathscr{F}^D} \sum_{j=0}^{p_K} \tilde{\phi}_{K,j}^D E_{ijk_d}^{KK} - \frac{1}{2} \sum_{\substack{\mathscr{F}\in\mathscr{F}^I+\mathscr{F}^D \\ \mathscr{F}\subset\partial K}} \left(\sum_{j=0}^{p_L} \tilde{\phi}_{L,j} E_{ijk_d}^{LL} + \sum_{j=0}^{p_R} \tilde{\phi}_{R,j} E_{ijk_d}^{LR} \right). \tag{4.5}$$

These equations can be solved independently in each element, since given the coefficients for the potential $\tilde{\phi}_K$ and the free-surface potential $\tilde{\phi}_K^D$, they are fully uncoupled from the velocity field in neighboring elements. We note that for the linear problem we only need to compute the z-component ($k_d = d$) of the velocity field.

The velocity potential computed using Eq. (3.2a) has an order of accuracy $O(h^{p+1})$ in the L^2 norm, where h denotes the mesh size and p the approximating polynomial order [9,4,1]. Without any treatment the velocity field computed using the technique described above can at best achieve an order of accuracy $O(h^p)$. However, achieving super convergence for the gradient of a finite element solution has been an active research area for quite some time [7,13,14,10,15] (and the references therein). Here we use the L^2 projection based higher order gradient recovery technique proposed by Heimsund et al. [7]. In this approach two meshes are required, a finer mesh with mesh size h_f and polynomial order p and a coarser mesh h_c with

polynomial order r, with $h_c > h_f$ and $r > p$. In [7] a detailed comparison has been given with respect to various polynomial orders and mesh sizes. To achieve a high order of accuracy improvement with increasing p it is required to maintain an appropriate ratio between p and r. For example, for a combination of $p = 1$ and $r = 2$ the accuracy improvement for the gradient will be higher than for the combination of $p = 3$ and $r = 4$. The latter can then be increased by choosing a higher value for r in an appropriate ratio, e.g. $r = 6$. We perform the computations of the velocity potential on the finer mesh and use a coarser mesh for computing the velocity field. Note that this technique is required in the z direction only since for the linear problem we only need the z-component of the velocity field.

5. DG discretization of the free-surface boundary conditions and time integration

In this section we provide the fully-discrete explicit in time DG scheme for the free-surface problem. Let $\mathscr{T}_h$ denote the tessellation of the domain Ω into elements $K = 1, 2, \ldots, N_e$. Let the elements $K \in \mathscr{T}_h$ be numbered such that the first N_F elements $K_F = 1, \ldots, N_F$ are at the free-surface. The mesh points in the domain Ω are denoted as $(x_G, G = 0, \ldots, N_G)$ and ordered such that $(x_{G_S}, G_S = 0, \ldots, N_{G_F})$ are the mesh points at the free-surface. At the free-surface we also denote the free-surface elements $(S_{K_S}, K_S = 1, \ldots, N_F)$, which are equal to the free-surface boundary of the elements $K_F = 1, \ldots, N_F$. Finally, the total degrees of freedom on the free-surface elements are denoted as $N^F = \sum_{K=1}^{N_F}(p_K + 1)$.

We now consider the free-surface potential $\phi^D(x,t)$ at $z = 0$, which provides a Dirichlet type boundary condition at the free-surface for the potential ϕ in the flow domain. Let $w(x,t)$ denote the z-component of the velocity field at the free-surface:

$$w(x,t) = \frac{\partial \phi}{\partial z}(x,0,t).$$

In each free-surface element we use a polynomial approximation for ϕ^D, ζ and w similar to (4.1) and by $\tilde{w}_{K,j}$, $\tilde{\phi}^D_{K,j}$, $\tilde{\zeta}_{K,j}$ we denote the expansion coefficients of w, ϕ^D, and ζ, respectively, in the element K connected to the free-surface. Let the matrix N^D denote the discrete Dirichlet to Neumann operator (see Appendix A.3 for the definition). For ease of notation we introduce the sub-matrices $N^{D,K,K'}_{j,i}$ as

$$N^{D,K,K'}_{j,i} = N^D_{(p_K+1)(K-1)+j+1,(p_K+1)(K'-1)+i+1}. \tag{5.1}$$

We can express the coefficients $\tilde{w}_{K,j}$ in terms of the coefficients of the free-surface potential $\tilde{\phi}^D_{K,j}$ using the discrete Dirichlet to Neumann operator N^D as

$$\tilde{w}_{K,j} = \sum_{K'=1}^{N_F}\sum_{i=0}^{p_K} N^{D,K,K'}_{j,i}\tilde{\phi}^D_{K',i}. \tag{5.2}$$

The DG discretization of the free-surface conditions in each free-surface element can now be written as

$$\sum_{j=0}^{p_K}\frac{\mathrm{d}\tilde{\phi}^D_{K,j}}{\mathrm{d}t}\int_K \mathscr{N}_{K,i}(x)\mathscr{N}_{K,j}(x)\,\mathrm{d}x = -\sum_{j=0}^{p_K}\tilde{\zeta}_{K,j}\int_K \mathscr{N}_{K,i}(x)\mathscr{N}_{K,j}(x)\,\mathrm{d}x,$$

$$\sum_{j=0}^{p_K}\frac{\mathrm{d}\tilde{\zeta}_{K,j}}{\mathrm{d}t}\int_K \mathscr{N}_{K,i}(x)\mathscr{N}_{K,j}(x)\,\mathrm{d}x = \sum_{j=0}^{p_K}\tilde{w}_{K,j}\int_K \mathscr{N}_{K,i}(x)\mathscr{N}_{K,j}(x)\,\mathrm{d}x,$$

for all $i = 0, \ldots, p_K$. Using the expression for $\tilde{w}_{K,j}$ in terms of $\tilde{\phi}^D_{K,j}$, given by (5.2), and the fact that the matrix $D^K_{i,j}$ is non-singular we obtain, for $K = 1, \ldots, N_F$ and $j = 0, \ldots, p_K$, the following system of expansion coefficients for the DG discretization of the free-surface potential and the wave height

$$\frac{\mathrm{d}\tilde{\phi}^D_{K,j}}{\mathrm{d}t} = -\tilde{\zeta}_{K,j}, \quad \frac{\mathrm{d}\tilde{\zeta}_{K,j}}{\mathrm{d}t} = \tilde{w}_{K,j}(\tilde{\phi}^D) = \sum_{K'=1}^{N_F}\sum_{i=0}^{p_K} N^{D,K,K'}_{j,i}\tilde{\phi}^D_{K',i}. \tag{5.3}$$

If we introduce $\Phi^C = ([\Phi^D][\zeta])^{\mathrm{T}}$, with $\zeta \in \mathbb{R}^{N^F}$, then the coefficients satisfy the equation

$$\frac{\mathrm{d}\Phi^C}{\mathrm{d}t} = P\Phi^C, \quad \text{where } P = \begin{bmatrix} 0 & -I \\ N^D & 0 \end{bmatrix}. \tag{5.4}$$

For the time integration of these coupled equations for the expansion coefficients we use the classical 4th-order four stage Runge–Kutta method. We briefly summarize the steps required to solve the coupled system of equation (5.3): Given ϕ_n and ζ_n we compute

$$\phi^D_{n+1} = \phi^D_n + \frac{1}{6}(k_{1\phi} + 2k_{2\phi} + 2k_{3\phi} + k_{4\phi}),$$

$$\zeta_{n+1} = \zeta_n + \frac{1}{6}(k_{1\zeta} + 2k_{2\zeta} + 2k_{3\zeta} + k_{4\zeta}),$$

where

$$k_{1\phi} = -\Delta t\zeta_n, \quad k_{1\zeta} = \Delta t w(\phi^D_n),$$

$$k_{2\phi} = -\Delta t\left(\zeta_n + \frac{1}{2}k_{1\zeta}\right), \quad k_{2\zeta} = \Delta t w\left(\phi^D_n + \frac{1}{2}k_{1\phi}\right),$$

$$k_{3\phi} = -\Delta t\left(\zeta_n + \frac{1}{2}k_{2\zeta}\right), \quad k_{3\zeta} = \Delta t w\left(\phi^D_n + \frac{1}{2}k_{2\phi}\right),$$

$$k_{4\phi} = -\Delta t(\zeta_n + k_{3\zeta}), \quad k_{4\zeta} = \Delta t w(\phi^D_n + k_{3\phi}).$$

6. Stability analysis

In this section we discuss the stability of the numerical scheme on a non-uniform mesh. We first discuss the semi-discrete scheme which suggests a sufficient value for the stabilization constant to guarantee the stability of the scheme. This is followed by a fully-discrete stability analysis which helps us to choose a suitable time step for a stable explicit time-discretization scheme.

6.1. Semi-discrete analysis: dispersion and dissipation of the numerical scheme

In this section we perform a semi-discrete Fourier analysis of the DG scheme. In [8] the dependence of the spatial stability on the mesh type was discussed within the framework of spectral methods. It was found that the asymmetry of the mesh causes instability in the numerical scheme. To overcome this, one of the promising techniques suggested was to add a small stabilization term to the free-surface boundary condition (2.1d), see equation (53) of [8]. The explicit Runge–Kutta DG scheme also exhibits a dependence of the stability on the mesh type and hence, as a remedy we also add a small stabilization term. However, it will be shown that the higher order gradient recovery technique for the velocity field, as discussed in Section 4.1, has a favorable effect on the size of the stabilization constant μ (see (6.1)), and therefore, we can choose very small values of μ for higher polynomial degrees to keep the scheme stable. This results in a very low dissipation.

Note here that we perform the Fourier analysis only on the free-surface equation (5.4) and not on the whole domain. Let k be the wave number and ω the frequency. Let us assume that we have a uniform mesh at the free-surface, with mesh size Δx, equipped with periodic boundary conditions in the x-direction. The interior mesh, however, can be non-uniform. We introduce the Fourier ansatz

$$\tilde{\phi}^D_{K,j} = \tilde{\phi}^F_j \mathrm{e}^{\imath(Kk\Delta x-\omega t)}, \quad \tilde{\zeta}_{K,j} = \tilde{\zeta}^F_j \mathrm{e}^{\imath(Kk\Delta x-\omega t)},$$

with $k\Delta x \in [0,\pi)$, since any wave with $k\Delta x \geqslant \pi$ cannot be represented on the mesh, and $\imath = \sqrt{-1}$. Now we need to determine the relation between k and ω for the numerical scheme. Introducing the Fourier ansatz in the free-surface discretization we have

$$(-\imath\omega\tilde{\phi}^F_j + \tilde{\zeta}^F_j)\mathrm{e}^{\imath(Kk\Delta x-\omega t)} = 0,$$
$$\left(-\sum_{K'=1}^{N_F}\sum_{i=0}^{p_K} \mathrm{e}^{\imath(K'-K)k\Delta x} N^{D,K,K'}_{j,i}\tilde{\phi}^F_i - \imath\omega\tilde{\zeta}^F_j\right)\mathrm{e}^{\imath(Kk\Delta x-\omega t)} = 0,$$

for $K = 1,\ldots,N_F$, $j = 0,\ldots,p_K$. Using $\tilde{\zeta}^F_j = \imath\omega\tilde{\phi}^F_j$ from the first equation in the second we get

$$\omega^2\tilde{\phi}^F_j - \sum_{K'=1}^{N_F}\sum_{i=0}^{p_K} \mathrm{e}^{\imath(K'-K)k\Delta x} N^{D,K,K'}_{j,i}\tilde{\phi}^F_i = 0.$$

Note that for a uniform mesh and periodic boundary conditions in the x direction the matrices $N^{D,K,K'}_{j,i}$ are circular permutations in K and the dispersion will therefore be independent of K. Let I_j denote the (p_K+1) identity matrix, then the dispersion relation is given by

$$\omega^2 I_j\tilde{\phi}^F_E - \sum_{K'=1}^{N_F} \mathrm{e}^{\imath(K'-K)k\Delta x} N^{D,K,K'}\tilde{\phi}^F_E = 0,$$

with $\tilde{\phi}^F_E \in \mathbb{R}^{p_K+1}$ denoting the Fourier coefficients in an element. Non-trivial solutions $\tilde{\phi}^F_E$ are obtained for those values of ω and k for which

$$\det\left(\omega^2 I_j - \sum_{K'=1}^{N_F} \mathrm{e}^{\imath(K'-K)k\Delta x} N^{D,K,K'}\right) = 0,$$

which represents an eigen-value problem with $\lambda^s = \omega^2$. If we introduce $\mu(k\Delta x) := \mathrm{e}^{\imath k\Delta x}$ then we obtain the following eigen-value problem:

$$\det\left(\lambda^s I_j - \sum_{K'=1}^{N_F} \mu^{K'-K} N^{D,K,K'}\right) = 0,$$

solving which we get p_K+1 eigen-values $\lambda^s_j(k\Delta x)$ with $\omega_j = \sqrt{\lambda^s_j(k\Delta x)}$. Since the Fourier ansatz assumes $\tilde{\phi}^D_{K,j} = \tilde{\phi}^F_j \mathrm{e}^{\imath(Kk\Delta x-\omega t)}$, the real and imaginary parts of λ^s_j are related to the frequency and dissipation of mode j, respectively. The mode with $\omega_j(k\Delta x)$ closest to the exact mode $\omega(k) = \sqrt{k\tanh(k)}$ and with the smallest dissipation is called the physical mode, the other modes are spurious or numerical modes.

We now show the effect of the asymmetry of the mesh on the stability of the scheme. In order to test the dependence of the stability of the numerical scheme on the mesh smoothness we use a uniform mesh with square elements and a randomly generated mesh (a mesh of square elements with the vertices randomly displaced up to a maximum of 30% of the edge length). However, since we need to have the same domain for the coarse mesh as well as the fine mesh (see the discussion in Section 4.1), we choose a coarse mesh with polynomial order $r = 2p$ for a fine mesh with polynomial order p. Thus, the node points of the coarse mesh will exactly match with the nodes of the fine mesh. Note that in general the algorithm is perfectly suited for an unstructured mesh with any reasonable choice of h_c, h_f, p and r. In order to investigate the effect of the higher order velocity recovery technique we first consider cases where this technique is *not* applied. Figs. 2 and 3 show that while the dispersion is unaffected by the asymmetry in the mesh, it causes the imaginary part of the physical mode to be non-negligibly positive, which results in instability of the numerical scheme, see Figs. 4 and 5.

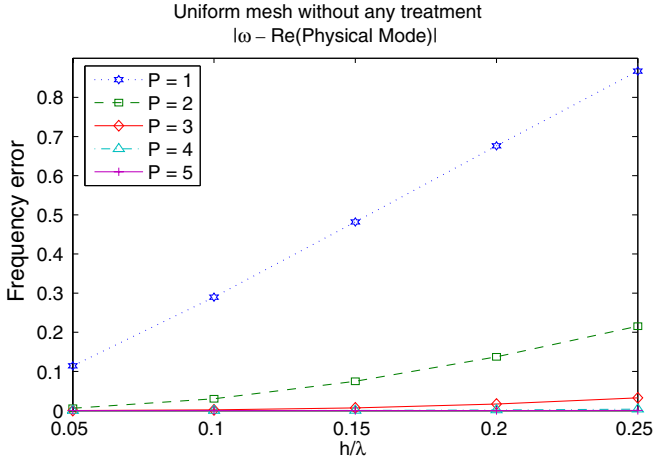

Fig. 2. Dispersion error of the scheme on the uniform mesh.

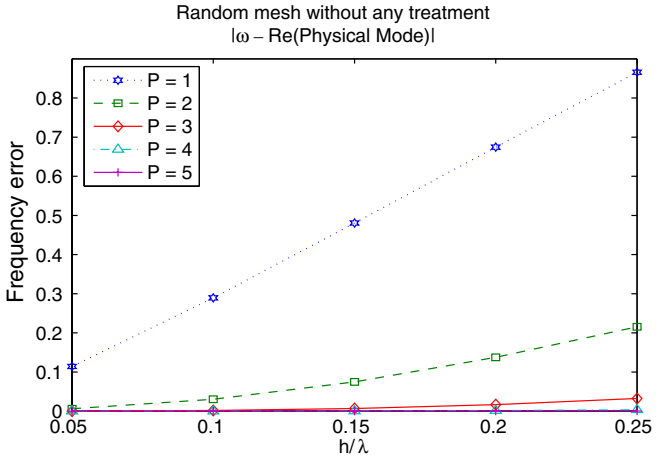

Fig. 3. Dispersion error of the scheme on the random mesh.

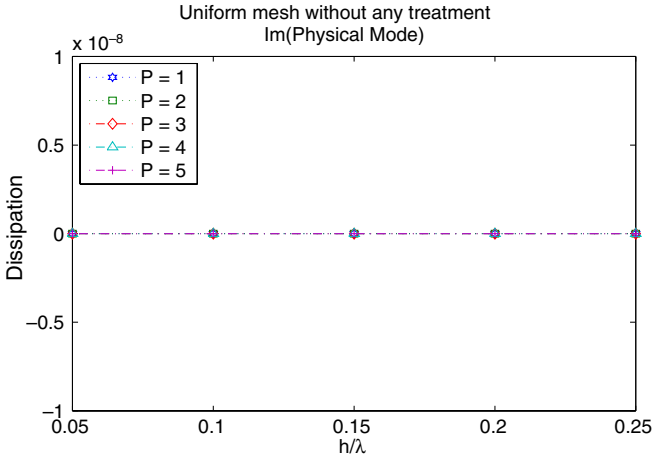

Fig. 4. Dissipation of the scheme on the uniform mesh.

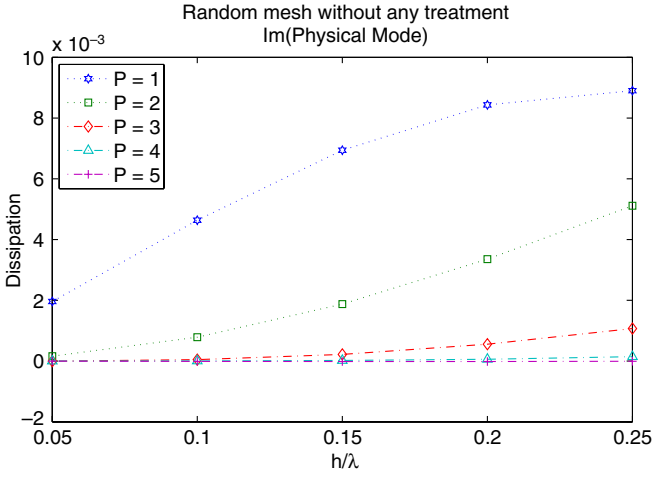

Fig. 5. Dissipation of the scheme on the random mesh.

To ensure stability we now add stabilization term to the free-surface condition:

$$\frac{\partial \zeta}{\partial t} = \frac{\partial \phi}{\partial z} + \mu \frac{\partial^2 \zeta}{\partial x^2}, \tag{6.1}$$

where μ is called the stabilization constant. Let μ_d denote the smallest value of μ to produce a stable solution. To discretize $\frac{\partial^2 \zeta}{\partial x^2}$ we again use the DG method, which straight-forwardly results in the stabilization term P^d. Eq. (5.4) is now changed to

$$\frac{\mathrm{d}\Phi^C}{\mathrm{d}t} = P\Phi^C, \quad \text{where } P = \begin{bmatrix} 0 & -I \\ N^D & -\mu_d P^d \end{bmatrix}. \tag{6.2}$$

We do not repeat the qualitative analysis of the dependence of the stabilization constant μ_d on h, p and the mesh skewness and refer the reader to [8] for details, where a relationship of the form $\mu_d = C\sigma^{\alpha p}$ was obtained, with C and α constants, p the polynomial degree and σ the mesh skewness. Only those values which are small enough to make the scheme stable for our choice of h and p are mentioned here. It is easy to see, however, that Eq. (6.1) satisfies the consistency requirement since the mesh skewness σ is proportional to the mesh size h for different sizes of similar shaped elements and hence, μ_d decays as $h^{\alpha p}$. In all cases hereafter we consider only the random mesh.

The effect of adding the stabilization term with $\mu_d = 0.0005$ is shown in Fig. 6, where the negative value of the dissipation indicates stability. However, the dissipation causes a decay in the wave amplitude which is undesirable, in particular for numerical simulations over a long period of time.

We now study the effect of higher order velocity recovery and suitably chosen μ_d in minimizing the wave dissipation. Fig. 7 shows that for small values of h/λ we can reduce the dissipation up to a factor of about 3 for $p = 1$, a factor of about 4 for $p = 2$ and a factor of about 10 for $p = 3$ with $\mu_d = 0.0005$, 0.00007 and 0.000042, respectively. Fig. 8 shows the value of μ_d for various polynomial orders which shows a pattern similar to Fig. 17 of [8]. Hence, while the stabilization operator has no effect on the spurious modes, which essentially have zero dissipation, see

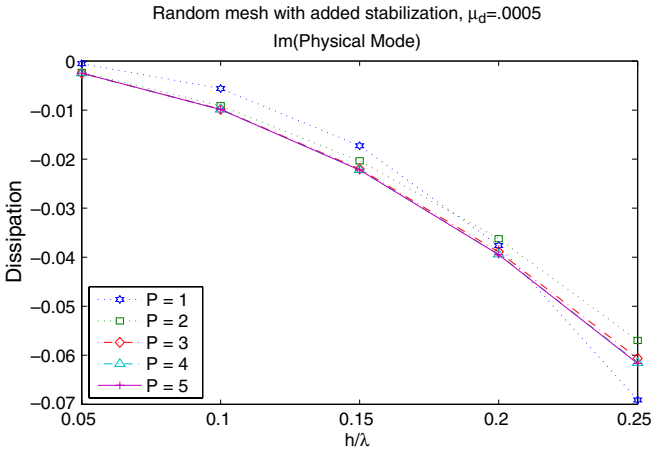

Fig. 6. Dissipation of the scheme with added stabilization.

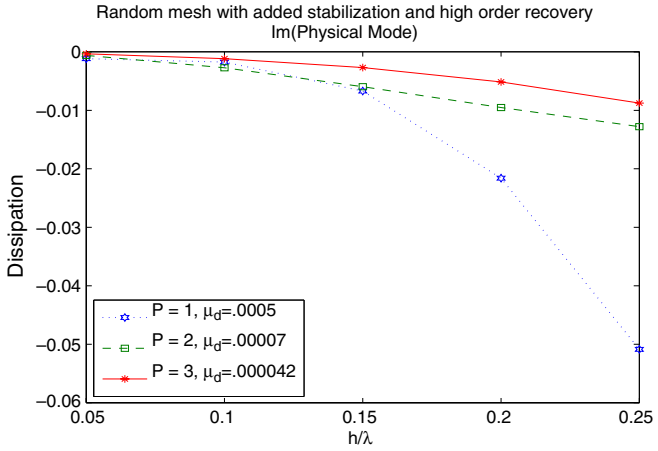

Fig. 7. Dissipation of the scheme with added stabilization, higher order velocity recovery and varying μ_d for higher p.

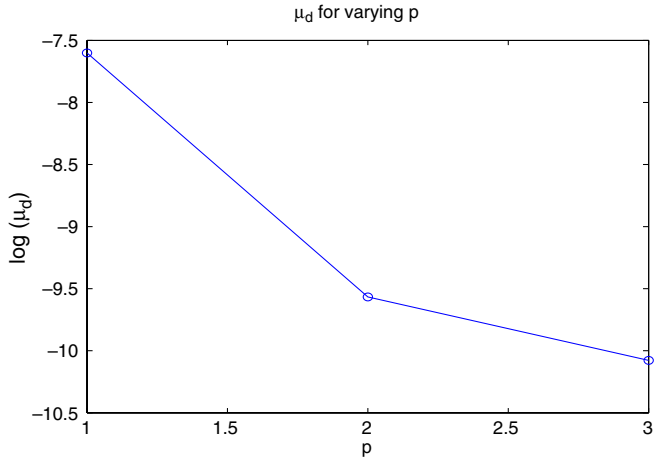

Fig. 8. The stabilization constant μ_d for varying p and $h = 0.1$.

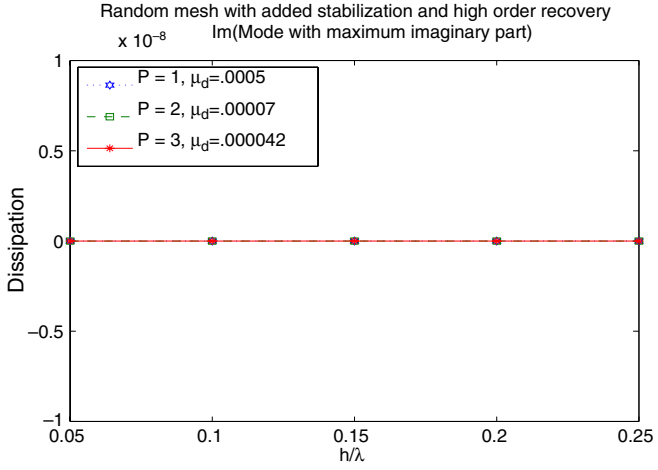

Fig. 9. Dissipation of the spurious modes with added stabilization, higher order velocity recovery and varying μ_d for higher p.

Fig. 9, we can achieve a very low dissipation for large polynomial order p using the higher order velocity recovery

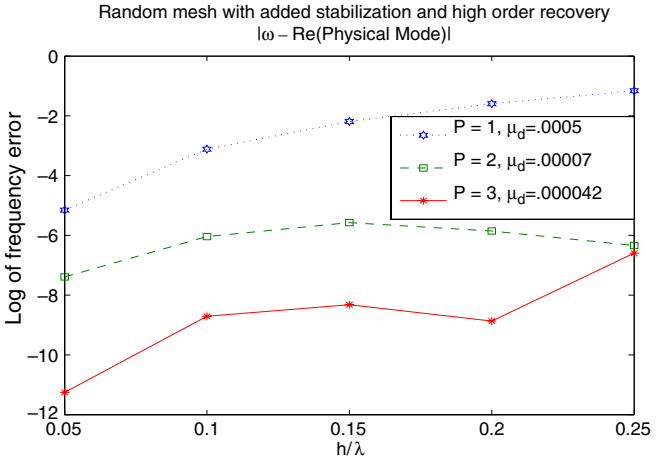

Fig. 10. Dispersion error with added stabilization, higher order velocity recovery and varying μ_d for higher p.

technique and a suitable choice of the stabilization constant, compare Figs. 6 and 7. Fig. 10 shows, on a logarithmic scale, the effect of the stabilization operator on the dispersion error when higher order velocity recovery technique and suitably chosen stabilization constants are used, which is very small for the values of $p \geqslant 3$.

6.2. Fully-discrete analysis

In this section we conduct a fully-discrete stability analysis of the scheme which helps us to find the largest time step within the stability region.

Following (5.4) we can express the fully-discrete RK4 scheme as

$$\Phi^C_{n+1} = Q(\Delta t)\Phi^C_n, \tag{6.3a}$$

where

$$Q(\Delta t) = \left(I + P\Delta t + \frac{(P\Delta t)^2}{2!} + \frac{(P\Delta t)^3}{3!} + \frac{(P\Delta t)^4}{4!}\right), \tag{6.3b}$$

where P includes the stabilization term, see (6.2). Introducing the Fourier ansatz

$$\tilde{\phi}^D_{K,j} = \tilde{\phi}^F_j \mathrm{e}^{\imath(Kk\Delta x - n\omega\Delta t)}, \quad \tilde{\zeta}_{K,j} = \tilde{\zeta}^F_j \mathrm{e}^{\imath(Kk\Delta x - n\omega\Delta t)},$$

and the matrix P^F as

$$P^F = \begin{bmatrix} 0 & -I \\ \sum_{K'=1}^{N_F} \mu^{K'-K} N^{D,K,K'} & -\mu_d P^d \end{bmatrix},$$

we obtain the following eigen-value problem for the RK4 scheme

$$(\mathrm{e}^{-\imath\omega\Delta t} I_{2j} - Q^F) \begin{pmatrix} \tilde{\phi}^F \\ \tilde{\zeta}^F \end{pmatrix} = 0,$$

where

$$Q^F = \left(I + P^F \Delta t + \frac{(P^F \Delta t)^2}{2!} + \frac{(P^F \Delta t)^3}{3!} + \frac{(P^F \Delta t)^4}{4!} \right), \qquad (6.4)$$

and I_{2j} is the $2(p_K+1)\times 2(p_K+1)$ identity matrix. If we introduce $\lambda^t = \mathrm{e}^{-\imath\omega\Delta t}$ then the eigen-values λ^t are determined by solving

$$\det(Q^F - \lambda^t I_{2j}) = 0,$$

and the numerical scheme is stable for $\Delta t \in (0, \Delta t_{\max})$ if $|\lambda^t| \leqslant 1$ for all values of $k\Delta x \in [0, \pi)$. We begin with noting that the explicit RK4 scheme is not unconditionally stable. Figs. 11–13 show the stability region (with amplification factor one) of the fully-discrete RK4 scheme on a random mesh with added stabilization and higher order velocity recovery for the minimal values of μ_d to ensure stability. It is clear that the scheme is stable for a reasonably large values of Δt. This is confirmed by the numerical examples in the next section.

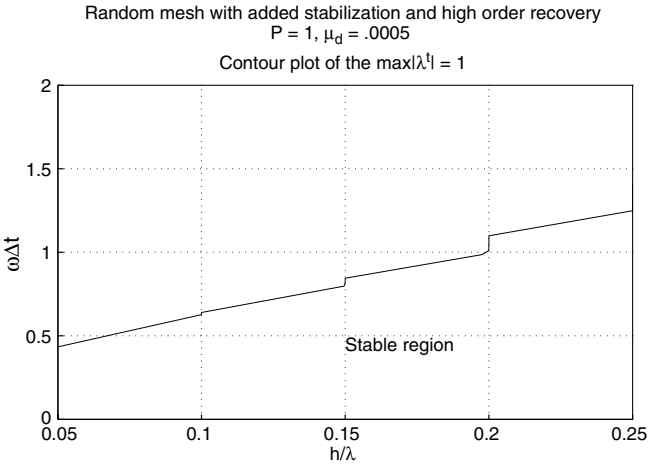

Fig. 11. Stability region of the fully-discrete scheme on a random mesh for $p = 1$.

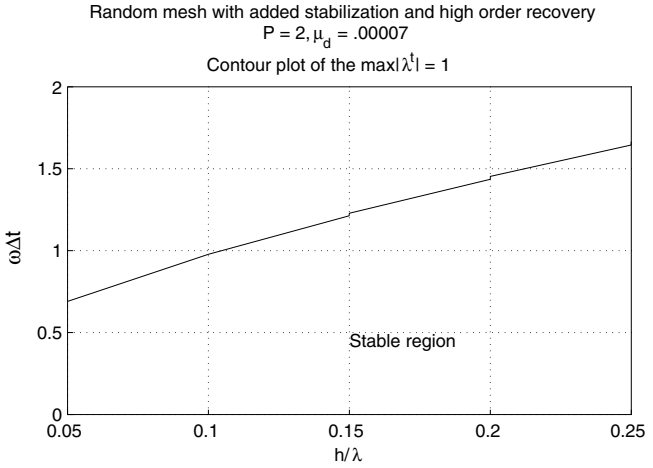

Fig. 12. Stability region of the fully-discrete scheme on a random mesh for $p = 2$.

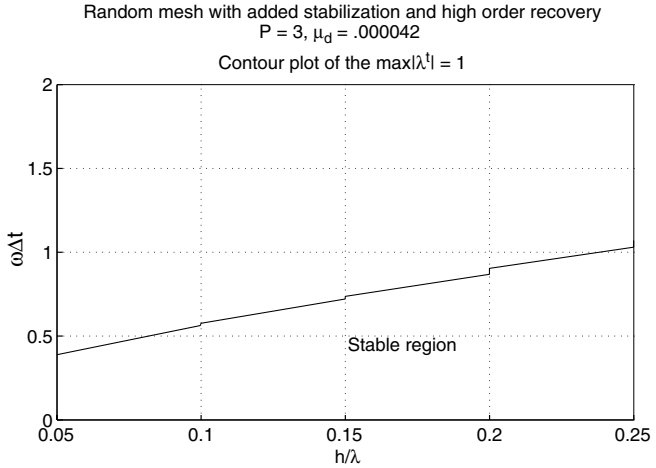

Fig. 13. Stability region of the fully-discrete scheme on a random mesh for $p = 3$.

7. Numerical examples

In this section we provide numerical results of some model problems which support the theoretical analysis. We first verify the spatial accuracy of the velocity potential and the free-surface height. For a precise estimate of the improvement in the accuracy of the computed velocity field by the L^2 projection based higher order recovery for a given h and p the reader can refer to [7]. Next, we consider the examples of [9] with the approach presented in this paper.

7.1. Time-harmonic waves in an unbounded domain

To verify the analysis and the accuracy of the proposed scheme we first consider as a model problem a harmonic wave in an unbounded domain. We choose a domain $[-1,1]\times[-1,0]$ with periodic boundary Γ_P at both ends $x = \pm 1$ of the domain.

The boundary conditions at Γ_P are given by

$$\phi(x + L_x, z, t) = \phi(x, z, t),$$

with L_x denoting the length of the periodic domain in the x-direction. Faces at Γ_P are considered internal faces, which connect the external part of Γ_P to the interior of the domain Ω. At the bottom of the domain $z = -1$ we consider a homogenous Neumann boundary condition. As initial free-surface we use the projection of the analytic solution onto the finite element basis functions. The analytic solution of this problem is given by

$$\phi = \phi_0 \cosh(k(z+1)) \cos(\omega t - kx), \qquad (7.1)$$

where ϕ_0 denotes the maximum amplitude of the velocity potential, k the wave number, which is related to the wavelength λ as $k = 2\pi/\lambda$, and ω the frequency of the oscillations. The frequency and wave number satisfy the following dispersion relation:

$$\omega^2 = k \tanh(k).$$

All quantities considered here are in dimensionless form. The initial conditions for the wave height at $z = 0$ is

$$\zeta = \phi_0 \omega \cosh(k) \sin(kx). \tag{7.2}$$

For all our examples in this section we choose a mesh of quadrilateral elements with random (up to a maximum of 30% of the edge length) displacement of vertices, the approximating polynomial order p up to 3, and ϕ_0 such that the maximum amplitude of the free-surface height is 5% of the water depth. Note that with the higher order velocity recovery technique this requires $p = 6$ on the coarse mesh with $2h$ mesh size.

As a first step we check the accuracy achieved in the computation of the free-surface height. To have an error only influenced by the space-discretization (and negligible effects from the time-discretization) we choose $\Delta t = 0.0001$. We consider polynomial degrees ranging from $p = 1$ to $p = 3$. We compute the error after the first time step and the mesh size h is chosen as 1/8, 1/16, 1/24 and 1/32 for all p. To keep the L^2 error within reasonable bounds for $p = 1, 2, 3$ we take λ as $1, 1/2, 1/3$, respectively. Figs. 14 and 15 show the h-convergence on a *log–log* scale for the velocity potential and the free-surface height. The convergence for the velocity potential is slightly less than the optimal rate of $O(h^{p+1})$. Without higher order velocity recovery the convergence rate for the free-surface height cannot be higher than $O(h^p)$. However, using the L^2 projection based gradient recovery technique the minimum convergence achieved for the free-surface height is $O(h^{p+1})$ (in fact towards ultra-convergent for $p = 1$ and $p = 3$).

As a next step we choose the time step $\Delta t = 0.1$, the mesh size $h = 0.1$, and the wave length $\lambda = 1$. Fig. 16 shows the free-surface height after 20 periods. The dispersion error for $p = 1$ is large and the wave is moving slower than the actual wave. The dissipation for $p = 1$ and $p = 2$ are almost the same, which confirms the observation of the semi-discrete analysis. The dissipation for $p = 3$ is further reduced and is quite small. For $p = 2$ the wave appears to be moving a little faster than the actual wave, however, for $p = 3$ the dispersion error is almost negligible.

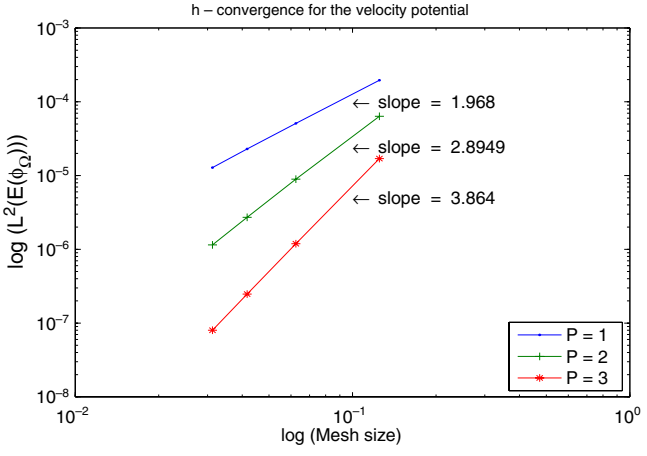

Fig. 14. h-convergence for the velocity potential of a wave in an unbounded domain.

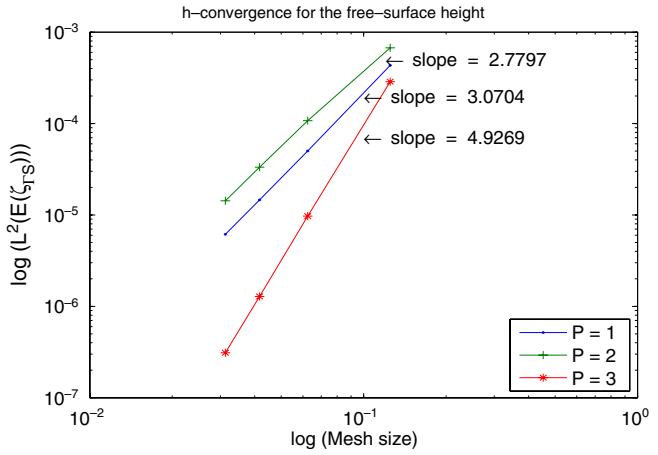

Fig. 15. h-convergence for the free-surface height of a wave in an unbounded domain.

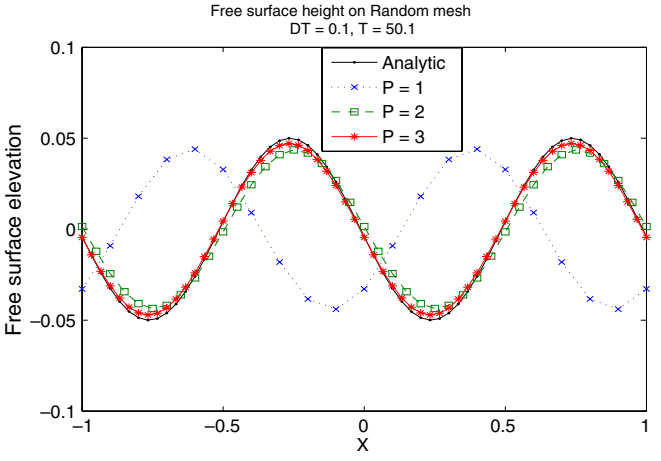

Fig. 16. Wave profile after 20 periods, harmonic wave in an unbounded domain.

7.2. Waves generated by a wave-maker

We now present the numerical simulation of waves generated by a wave-maker in a model basin. For experimental and other numerical results the reader can refer to [11]. The domain is considered as $[0, 10] \times [-1, 0]$ and the wave-maker is at $x = 0$. We assume homogenous Neumann boundary conditions at the bottom $z = -1$ and at the wall opposite to the wave-maker at $x = 10$. The initial free-surface height and the velocity potential are zero, and a time periodic Neumann boundary condition, governing the normal velocity, is applied at the wave-maker. The normal velocity profile is linear, starting with zero at the bottom, and has a maximum amplitude of 0.02. The frequency of the time-harmonic motion is 1.8138. Fig. 17 shows the wave profile at $T = 20$, when the wave starts approaching the wall against the wave-maker. The waves with $p = 2$ and $p = 3$ are identical whereas $p = 1$ shows a small

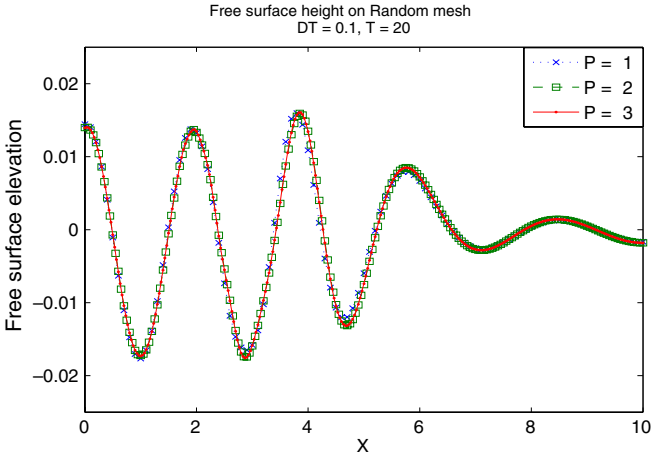

Fig. 17. Wave profile at $T = 20$, wave generated by a wave-maker.

difference with the remaining two. Fig. 18 shows the wave profile at $T = 38.2$, when the wave gains full height against the wall opposite to the wave-maker. The waves with $p = 2$ and $p = 3$ are still almost identical and for $p = 1$ the effect of the damping becomes visible at $X = 10$. Fig. 19 shows

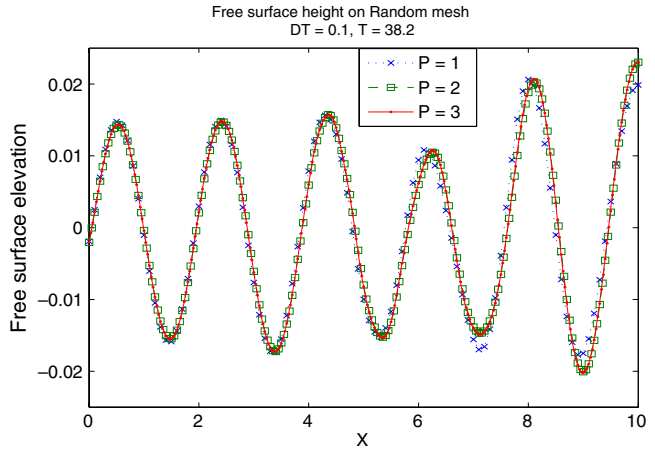

Fig. 18. Wave profile at $T = 38.2$, wave generated by a wave-maker.

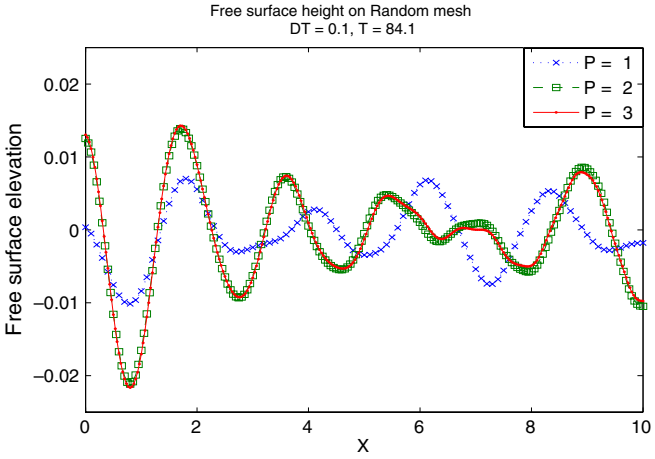

Fig. 19. Wave profile at $T = 84.1$, wave generated by a wave-maker.

the wave profile at $T = 84.1$, when the wave is traveling back into the domain and affects the pattern of the generated wave. The waves with $p = 2$ and $p = 3$ are still close but the wave for $p = 1$ clearly shows a different pattern. The results presented here with $\Delta t = 0.1$ show an excellent agreement with the results in [9], where $\Delta t = 0.02$ and for $p = 2$ and $p = 3$ the accuracy of the scheme in [9] has a strong dependence on the time step since this method is only second order accurate in time.

8. Conclusions

We have presented an alternative approach based on an explicit Runge–Kutta DG method for linear free-surface gravity waves compared to the implicit DG method discussed in [9], which is only second order accurate in time. To ensure the stability of the scheme it is required to add a small stabilization term to the free-surface condition. Using a higher order velocity recovery technique and a suitable choice of stabilization constants the damping of the wave amplitude can be made negligibly small to guarantee a sufficiently accurate simulation over a long period of time. For the nonlinear problem the algorithm requires only a straight-forward extension of the free-surface boundary condition without compromising the symmetry or positive-definiteness of the DG discretization matrix. Hence, most of the efficient iterative solvers can be applied to solve the resulting system of algebraic equations. In fact, the core of the method, i.e. the solution of the Laplace problem, is entirely unchanged for the nonlinear problem, and only the free-surface boundary conditions are changed.

Acknowledgement

The research of the first author is supported by the Austrian Academy of Sciences.

Appendix A. Derivation of discrete Dirichlet to Neumann operator

In this appendix we introduce the various matrix notations and give the equations for the expansion coefficients of the velocity potential. Subsequently, we define the matrix N^D, the discrete Dirichlet to Neumann operator, which is required for the representation of the expansion coefficients of the velocity field in terms of the expansion coefficients of the velocity potential at the free-surface.

A.1. Computation of the local lifting operator $\mathscr{R}_{\mathscr{F}}$

First, we consider the computation of the local lifting operator $\mathscr{R}_{\mathscr{F}}$ (2.4). We approximate $\mathscr{R}_{\mathscr{F}}(\llbracket \phi_h \rrbracket)$ and v_h for a given element K using the polynomial basis functions $\mathscr{N}_{Kj}$ as follows:

$$\mathscr{R}_{\mathscr{F},K,k_d}([\![\phi_h]\!]) = \sum_{j=0}^{p_K} \widetilde{\mathscr{R}}_{K,k_d,j}\mathscr{N}_{K,j}(x),$$

$$v_{K,k_d,h} = \sum_{j=0}^{p_K} \tilde{v}_{K,k_d,j}\mathscr{N}_{K,j}(x), \quad x \in K,$$

where $\widetilde{\mathscr{R}}_{K,k_d,j}, \tilde{v}_{K,k_d,j}$ are the expansion coefficients for the k_dth component of $\mathscr{R}_{\mathscr{F},K}$ and $v_{K,h}$, respectively, in element K. Using the definition of the trace operators and the polynomial approximation into (2.4) and the notations introduced above, after some algebraic manipulations, we get

$$\widetilde{\mathscr{R}}_{L,k_d,n} = \frac{1}{2}\left(\sum_{j=0}^{p_L} F^{LL}_{njk_d}\tilde{\phi}_{L,j} + \sum_{j=0}^{p_R} F^{LR}_{njk_d}\tilde{\phi}_{R,j}\right),$$

$$\widetilde{\mathscr{R}}_{R,k_d,n} = \frac{1}{2}\left(\sum_{j=0}^{p_L} F^{RL}_{njk_d}\tilde{\phi}_{L,j} + \sum_{j=0}^{p_R} F^{RR}_{njk_d}\tilde{\phi}_{R,j}\right),$$

where

$$F^{MN}_{njk_d} = \sum_{i=0}^{p_M} [D^M_{ni}]^{-1} E^{MN}_{ijk_d}. \tag{A.1}$$

For a Dirichlet face of an element K this will be reduced to

$$\widetilde{\mathscr{R}}_{K,k_d,n} = \sum_{j=0}^{p_K} F^{KK}_{njk_d}\tilde{\phi}_{K,j}.$$

We now show that a further simplification of $\mathscr{R}_{\mathscr{F}}$ terms is possible. First, we introduce $\mathscr{F}^{LMR} \in \mathbb{R}^{p_L \times p_R}$, which will be needed in the sequel:

$$\mathscr{F}^{LMR}_{ij} = \sum_{k_d=1}^{d}\sum_{n=0}^{p_M} E^{ML}_{nik_d}\sum_{k_d=1}^{d}\left(\sum_{i'=0}^{p_M}[D^M_{ni'}]^{-1}E^{MR}_{i'jk_d}\right). \tag{A.2}$$

We demonstrate the simplification of the computation of the $\mathscr{R}_{\mathscr{F}}$ terms by considering the following term:

$$\begin{aligned}\int_{\mathscr{F}} \mathscr{R}_{\mathscr{F},R}([\![\phi_h]\!])\cdot\psi_{L,h}n_L\,\mathrm{d}s &= \frac{1}{2}\sum_{i=0}^{p_L}\tilde{\psi}_{L,i}\int_{\mathscr{F}}\left(\sum_{k_d=1}^{d}\mathscr{N}_{L,i}n_{L,k_d}\right)\\ &\times\sum_{k_d=1}^{d}\sum_{n=0}^{p_R}\left(\sum_{j=0}^{p_L}F^{RL}_{njk_d}\tilde{\phi}_{L,j}+\sum_{j=0}^{p_R}F^{RR}_{njk_d}\tilde{\phi}_{R,j}\right)\mathscr{N}_{R,n}\,\mathrm{d}s.\end{aligned}$$

Let T_1 denote the first term in the integral on the right-hand side. Using the definition of $F^{RL}_{njk_d}$ and rearranging the terms we have

$$\begin{aligned}T_1 &= \left(\sum_{k_d=1}^{d}\sum_{n=0}^{p_R}\int_{\mathscr{F}}\mathscr{N}_{L,i}\mathscr{N}_{R,n}n_{L,k_d}\,\mathrm{d}s\right)\\ &\times\left(\sum_{k_d=1}^{d}\sum_{j=0}^{p_L}\tilde{\phi}_{L,j}\sum_{i'=0}^{p_R}[D^R_{ni'}]^{-1}\int_{\mathscr{F}}\mathscr{N}_{R,i'}\mathscr{N}_{L,j}n_{L,k_d}\,\mathrm{d}s\right).\end{aligned}$$

Using the definitions of $\mathscr{F}^{LMR}_{ij}$ from (A.2) we get

$$\begin{aligned}T_1 &= \sum_{j=0}^{p_L}\tilde{\phi}_{L,j}\sum_{k_d=1}^{d}\sum_{n=0}^{p_R}E^{RL}_{nik_d}\sum_{k_d=1}^{d}\left(\sum_{i'=0}^{p_R}[D^R_{ni'}]^{-1}E^{RL}_{i'jk_d}\right)\\ &= \sum_{j=0}^{p_L}\tilde{\phi}_{L,j}\mathscr{F}^{LRL}_{ij}.\end{aligned}$$

Similarly, from the definitions of $F^{MN}_{njk_d}$, and $E^{MN}_{ijk_d}$ the other $\mathscr{R}_{\mathscr{F}}$ terms can be simplified. This gives us

$$\int_{\mathscr{F}}\mathscr{R}_{\mathscr{F},R}([\![\phi_h]\!])\cdot\psi_{L,h}n_L\,\mathrm{d}s = \frac{1}{2}\left(\sum_{j=0}^{p_L}\tilde{\phi}_{L,j}\mathscr{F}^{LRL}_{ij}+\sum_{j=0}^{p_R}\tilde{\phi}_{R,j}\mathscr{F}^{LRR}_{ij}\right).$$

The $\mathscr{R}_{\mathscr{F}}$ term on the Dirichlet boundary faces in the linear form can be analogously simplified to take the form

$$\int_{\mathscr{F}}\mathscr{R}_{\mathscr{F},K}([\![\phi_h]\!])\cdot\psi_{K,h}n_K\,\mathrm{d}s = \sum_{j=0}^{p_K}\tilde{\phi}^D_{K,j}\mathscr{F}^{KKK}_{ji}.$$

A.2. Construction of the DG matrix for the velocity potential

In this section we will describe the construction of the matrix A for the DG discretization of the velocity potential function ϕ_h.

The DG discretization of the velocity potential (3.2a) can be represented in a matrix form as

$$A\phi = X, \quad \text{with } A \in \mathbb{R}^{N\times N}, \quad \phi, X \in \mathbb{R}^N, \tag{A.3}$$

where the matrix A consists of N_e^2 blocks $[A] \in \mathbb{R}^{(p_K+1)\times(p_K+1)}$, and the vectors ϕ, X of blocks $[\phi], [X] \in \mathbb{R}^{(p_K+1)}$, respectively. Before proceeding further we introduce the following matrices: $C^K \in \mathbb{R}^{p_K\times p_K}$, $B^{LR} \in \mathbb{R}^{p_L\times p_R}$, and $\mathscr{E}^{LR} \in \mathbb{R}^{p_L\times p_R}$, with the components

$$C^K_{ij} = \int_K \nabla_h\mathscr{N}_{K,i}\cdot\nabla_h\mathscr{N}_{K,j}\,\mathrm{d}x, \quad B^{LR}_{ij} = \int_{\mathscr{F}}\mathscr{N}_{L,i}n_L\cdot\nabla_h\mathscr{N}_{R,j}\,\mathrm{d}s,$$

$$\mathscr{E}^{LR}_{ij} = \int_{\mathscr{F}}\mathscr{N}_{L,i}\mathscr{N}_{R,j}\,\mathrm{d}s. \tag{A.4}$$

Now the matrix A is constructed in a way suitable for unstructured meshes as follows:

(a) Initialize $A = 0$, (A.5)

(b) $[A_{ij}]^{KK} = C^K_{ij}$,

(c) $[A_{ij}]^{LL} \leftarrow \sum_{\mathscr{F}\in\mathscr{F}^I}\left\{[A_{ij}]^{LL} - \frac{1}{2}(B^{LL}_{ji}+B^{LL}_{ij}) + \frac{1}{4}\eta_{\mathscr{F}}(\mathscr{F}^{LLL}_{ij}+\mathscr{F}^{LRL}_{ij})\right\}$,

(d) $[A_{ij}]^{LR} \leftarrow \sum_{\mathscr{F}\in\mathscr{F}^I}\left\{[A_{ij}]^{LR} - \frac{1}{2}(B^{RL}_{ji}+B^{LR}_{ij}) + \frac{1}{4}\eta_{\mathscr{F}}\left(\mathscr{F}^{LLR}_{ij}+\mathscr{F}^{LRR}_{ij}\right)\right\}$,

(e) $[A_{ij}]^{RL} \leftarrow \sum_{\mathscr{F}\in\mathscr{F}^I}\left\{[A_{ij}]^{RL} - \frac{1}{2}(B^{LR}_{ji}+B^{RL}_{ij}) + \frac{1}{4}\eta_{\mathscr{F}}(\mathscr{F}^{RLL}_{ij}+\mathscr{F}^{RRL}_{ij})\right\}$,

(f) $[A_{ij}]^{RR} \leftarrow \sum_{\mathscr{F}\in\mathscr{F}^I}\left\{[A_{ij}]^{RR} - \frac{1}{2}(B^{RR}_{ji}+B^{RR}_{ij}) + \frac{1}{4}\eta_{\mathscr{F}}(\mathscr{F}^{RLR}_{ij}+\mathscr{F}^{RRR}_{ij})\right\}$,

(g) $[A_{ij}]^{KK} \leftarrow \sum_{\mathscr{F}\in\mathscr{F}^D}\{[A_{ij}]^{KK} - (B^{KK}_{ji}+B^{KK}_{ij}) + \eta_{\mathscr{F}}\mathscr{F}^{KKK}_{ij}\}$.

Similarly, the right-hand side vector X is constructed as follows:

$$\text{(a)}\quad \text{Initialize } X = 0, \tag{A.6}$$

$$\text{(b)}\quad [X_i]^K \leftarrow \sum_{\mathscr{F}\in\mathscr{F}^D}\sum_{j=0}^{p_K}\tilde{\phi}^D_{K,j}(-B^{KK}_{ji} + \eta_{\mathscr{F}}\mathscr{F}^{KKK}_{ji}),$$

$$\text{(c)}\quad [X_i]^K \leftarrow \sum_{\mathscr{F}\in\mathscr{F}^N}\left([X_i]^K + \sum_{j=0}^{p_K}\tilde{g}^N_{K,j}\mathscr{E}^{KK}_{ij}\right),$$

where $\tilde{\phi}^D_{K,j}\in\mathbb{R}^{p_K+1}$ and $\tilde{g}^N_{K,j}\in\mathbb{R}^{p_K+1}$ are the expansion coefficients of ϕ^D and g_N in element K, respectively. Note that the Dirichlet and Neumann boundary data are also approximated using polynomial basis functions, see (4.1).

A.3. Construction of discrete Dirichlet to Neumann operator

An essential ingredient in the analysis of the numerical algorithm is the discrete Dirichlet to Neumann operator which links the free-surface potential to the normal velocity at the free-surface. In this appendix we summarize the construction of N^D, the discrete Dirichlet to Neumann operator.

We rewrite (A.3) as

$$\sum_{j'=1}^{N} A_{i'j'}\Phi_{j'} = X_{i'}, \quad \forall i' = 1,\ldots,N,$$

with $\Phi_{(p_K+1)(K-1)+j+1} = \tilde{\phi}_{K,j}$, $K\in\{1,\ldots,N_e\}$, $j\in\{0,\ldots,p_K\}$. The vector $X\in\mathbb{R}^N$ depends on the Dirichlet boundary condition $\phi^D(x,t)$ at the free-surface and the Neumann boundary condition which has been set to zero in the stability analysis. We can express the vector X as $X = G^D([\Phi^D][0])^{\mathrm{T}}$, where T denotes the transpose, and $0\in\mathbb{R}^{N-N^F}$ the zero vector, $\Phi^D\in\mathbb{R}^{N^F}$ the vector of expansion coefficients of the Dirichlet boundary data, and $G^D\in\mathbb{R}^{N\times N}$ the associated matrix. Hence, $\Phi^D = ([\tilde{\phi}^D_1][\tilde{\phi}^D_2],\ldots,[\tilde{\phi}^D_{N_F}])^{\mathrm{T}}$, with $[\tilde{\phi}^D_K]$ a $\mathbb{R}^{(p_K+1)}$ vector with the expansion coefficients in each element. The expansion coefficients of the velocity potential in the DG discretization can thus be expressed as

$$\Phi = A^{-1}X = G([\Phi^D][0])^{\mathrm{T}}, \quad \text{where } G = A^{-1}G^D. \tag{A.7}$$

The DG discretization for the k_dth component of the velocity field (4.5) can be similarly written in matrix form as

$$\sum_{j'=1}^{N^F} L_{i'j'}W_{j'} = Y_{i'}, \quad \forall i' = 1,\ldots,N^F,$$

with $W_{(p_K+1)(K-1)+j+1} = \tilde{w}_{K,j}$, $K\in\{1,\ldots,N_F\}$, $j\in\{0,\ldots,p_K\}$. We also have $L\in\mathbb{R}^{N^F\times N^F}$ and $Y\in\mathbb{R}^{N^F}$. Further, since we need the data from all the neighboring elements for the integrals $E^{LR}_{ijk_d}$ in (4.5), the vector Y will have a contribution from the potential in the elements K_F attached to the free-surface and those elements which are connected to these elements opposite to the free-surface face. We define $G_T\in\mathbb{R}^{2N^F\times N^F}$, the sub-matrix of G related to these $2N_F$ elements, as

$$G_T = G(1,\ldots,2N^F,1,\ldots,N^F).$$

Hence, the vector Y can be expressed as $Y = R\Phi^{D_2}$, where $\Phi^{D_2} = G_T\Phi^D$, and $R\in\mathbb{R}^{N^F\times 2N^F}$ is a matrix with its blocks formed by M^K and E^{LR}, see (4.5). We can now express the expansion coefficients of the k_dth component of the velocity field in terms of Φ^D as

$$W = L^{-1}RG_T[\Phi^D]^{\mathrm{T}}. \tag{A.8}$$

If we define the matrix $N^D\in\mathbb{R}^{N^F\times N^F}$ as $N^D = L^{-1}RG_T$ we get the discrete Dirichlet to Neumann operator N^D:

$$W = N^D[\Phi^D]^{\mathrm{T}}. \tag{A.9}$$

References

[1] D.N. Arnold, F. Brezzi, B. Cockburn, L.D. Marini, Unified analysis of discontinuous Galerkin Methods for Elliptic Problems, SIAM J. Numer. Anal. 39 (5) (2002) 1749–1779.

[2] F. Bassi, S. Rebay, G. Mariotti, S. Pedinotti, M. Savini, A high-order accurate discontinuous finite element method for inviscid and viscous turbomachinery flows, in: R. Decupere, G. Dibelius (Eds.), Proceedings of 2nd European Conference on Turbomachinery, Fluid Dynamics and Thermodynamics, Technologisch Instituut, Antwerpen, Belgium, 1997, pp. 99–108.

[3] S.C. Brenner, L.R. Scott, The Mathematical Theory of Finite Element Methods, second ed., Springer-Verlag, New York, 2002.

[4] F. Brezzi, G. Manzini, D. Marini, P. Pietra, A. Russo, Discontinuous finite elements for diffusion problems, Atti Convegno in onore di F. Brioschi (Milan, 1997), Instituto Lombardo, Accademia di Scienze e Lettere, Milan, Italy, 1999, pp. 197–217.

[5] F. Brezzi, G. Manzini, D. Marini, P. Pietra, A. Russo, Discontinuous Galerkin approximations for Elliptic Problems, Numer. Methods Partial Differ. Equat. 16 (4) (2000) 365–378.

[6] B. Cockburn, C.W. Shu, The local discontinuous Galerkin method for time-dependent convection-diffusion systems, SIAM J. Numer. Anal. 35 (6) (1998) 2440–2463.

[7] B.O. Heimsund, X.C. Tai, J. Wang, Superconvergence for the gradient of finite element approximations by L^2 projections, SIAM J. Numer. Anal. 40 (4) (2002) 1263–1280.

[8] I. Robertson, S.J. Sherwin, Free-surface flow simulation using *hp*/spectral elements, J. Comput. Phys. 155 (1999) 26–53.

[9] J.J.W. van der Vegt, S.K. Tomar, Discontinuous Galerkin method for linear free-surface gravity waves, J. Sci. Comput. 22 (1) (2005) 541–577.

[10] J. Wang, A superconvergence analysis for finite element solutions by the least-squares surface fitting on irregular meshes for smooth problems, J. Math. Study 33 (2000) 229–243.

[11] J.H. Westhuis, The numerical simulation of nonlinear waves in a hydrodynamic model test basin, Ph.D. thesis, University of Twente, Enschede, The Netherlands, 2001.

[12] G.B. Whitham, Linear and Nonlinear Waves, Wiley, New York, 1974.

[13] Z. Zhang, Ultraconvergence of the patch recovery technique, Math. Comput. 65 (216) (1996) 1431–1437.

[14] Z. Zhang, Ultraconvergence of the patch recovery technique II, Math. Comput. 69 (229) (1999) 141–158.

[15] Z. Zhang, Polynomial preserving gradient recovery and *a posteriori* estimate for bilinear element on irregular quadrilaterals, Int. J. Numer. Anal. Modell. 1 (1) (2004) 1–24.