Journal of Computational Physics 299 (2015) 597–612

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

Artificial boundary layers in discontinuous Galerkin solutions to shallow water equations in channels

D. Wirasaet [a,*], S.R. Brus [a], C.E. Michoski [c], E.J. Kubatko [b], J.J. Westerink [a], C. Dawson [c]

[a] Environmental Fluid Dynamics Group, Department of Civil and Environmental Engineering and Earth Sciences, University of Notre Dame, Notre Dame, IN, 46556, USA
[b] Department of Civil and Environmental Engineering and Geodetic Science, The Ohio State University, Columbus, OH, 43210, USA
[c] Institute for Computational Engineering and Sciences, The University of Texas at Austin, Austin, TX, 78712, USA

ARTICLE INFO

Article history:
Received 17 July 2013
Received in revised form 2 April 2015
Accepted 10 July 2015
Available online 14 July 2015

Keywords:
Discontinuous Galerkin
Finite elements
Shallow water equations
Open channel
Numerical fluxes
No-normal flow boundary condition
Iso- and super-parametric finite elements

ABSTRACT

In this work, we consider the application of Discontinuous Galerkin (DG) solutions to open channel flow problems, governed by two-dimensional shallow water equations (SWE), with solid curved wall boundaries on which the no-normal flow boundary conditions are prescribed. A commonly used approach consists of straightforwardly imposing the no-normal flow condition on the linear approximation of curved walls. Numerical solutions indicate clearly that this approach could lead to unfavorable results and that a proper treatment of the no-normal flow condition on curved walls is crucial for an accurate DG solution to the SWE. In the test case used, errors introduced through the commonly used approach result in artificial boundary layers of one-grid-size thickness in the velocity field and a corresponding over-prediction of the surface elevation in the upstream direction. These significant inaccuracies, which render the coarse mesh solution unreliable, appear in all DG schemes employed including those using linear, quadratic, and cubic DG polynomials. The issue can be alleviated by either using an approach accounting for errors introduced by the geometric approximation or an approach that accurately represents the geometry.

1. Introduction

The shallow water equations (SWE) serve as an excellent model for incompressible flow with horizontal scales much larger than depth. The SWE are used extensively in modeling many environmental flows, such as tides, hurricane induced coastal flooding, open channel and riverine flow. Simulation of these problems often involves large, geometrically complicated domains and integration over a long period of time. Numerical methods to accurately solve the SWE must be able to propagate long waves and accurately simulate convective processes. Successful continuous Galerkin (CG) finite element solutions to the SWE include, but are not limited to, those devised in [1–4]. Discontinuous Galerkin (DG) finite element methods (see [5–7] and references therein for reviews and detailed accounts of DG methods), which excel in the solution of propagation- and convection-dominated problems, have emerged as a powerful alternative for solving the SWE [8–15]. Conceptually similar to finite volume (FV) methods, DG methods inherently posses the property of being conservative on

* Corresponding author.
E-mail address: dwirasae@nd.edu (D. Wirasaet).

http://dx.doi.org/10.1016/j.jcp.2015.07.015

the elemental level, a desirable property when coupling flow and transport equations. Unlike FV methods, high-order DG schemes on unstructured meshes can be constructed in a straightforward manner. Since they employ a piecewise discontinuous approximation, DG methods are able to accommodate non-conforming meshes and allow the use of polynomial approximations of arbitrary order in each element, thus making them naturally well-suited for an *hp*-adaptive discretization. In addition, the parallel implementation of DG schemes is highly scalable when used in conjunction with explicit time integration schemes [10].

While DG methods have numerous favorable properties, one major drawback of DG solutions in comparison to CG solutions on a given mesh is the larger number of degrees of freedom, which directly implies greater computational costs. The performance study of DG and CG methods for the SWE in [16] demonstrates that, for linear elements on identical meshes, the cost per time step of the DG solution [8,9] is approximately four to five times higher than that of the CG solution [2] (the latter solves the generalized wave continuity equation, a reformulated form of the SWE). Such a higher cost is not as alarming as it seems as the subsequent study [10] demonstrates that the DG method has comparable or higher efficiency in terms of obtaining a specified error level for a given computational cost and in terms of scalability on parallel machines. Most SWE solvers are first and second order accurate methods that are based on cell-averaged FV and linear finite elements. Indeed, for problems with smooth solutions, as demonstrated in [17,18], DG solutions offer a significant computational cost-per-accuracy when using high-order elements, i.e. elements with polynomial interpolants of degree p greater than unity.

Developments made over the years, described in a number of papers [8–10,12,19], enable Dawson et al. [13] to apply the linear-element DG methods to a realistic modeling of hurricane-induced coastal and inland flooding. In [13], the results from linear-element DG methods are validated against the observation data and compared with the results from ADvanced Circulation (ADCIRC) code [2], a CG-based SWE solver used extensively in such applications. Solutions, computed with an identical high-resolution mesh and physical parameter values, from these two methods agree well in most of the domain; however, significant disagreement in the results is seen in inland areas, especially in meandering channels. In the channels, the surge level of the ADCIRC solution is in good agreement with the observation data. However, the surge level in the DG solution is damped compared to that of the ADCIRC solution and attenuates at a faster rate in the upstream direction. To a certain degree, this indicates that the DG solution is more diffusive in channels and hinders DG methods from becoming a viable tool in storm-surge applications.

In this work, motivated in part by an attempt to resolve the issue mentioned above, we investigate the effect of curved-wall boundary treatments in DG solutions for SWE to open channel flow. As widely employed in CG calculations, DG calculations simply replace channel curved walls with a linear approximation (see for example [20,21]) and apply the no-normal flow condition on each straight segment in a straightforward manner. In gas dynamics, Bassi and Rebay [22] demonstrate that DG solutions are highly sensitive to the accuracy of the representation of a solid curved wall, a boundary on which the no-normal flow condition is prescribed. Numerical results shown therein (also see [23]) demonstrate that the DG methods under p-refinement fail to yield a numerical solution that converges to the true solution when imposing the no-normal flow (or slip) condition on the linear approximation of the geometry, i.e. on a set of straight segments. Errors introduced by the geometric approximation appear to have a strong effect on the solution away from the boundary. Bassi and Rebay [22] show that this issue can be resolved by approximating the geometry using a polynomial of degree that is at least equal to the degree of the DG polynomial but not less than two, i.e. using at least iso-parametric elements for $p > 1$ and super-parametric elements for $p = 1$ for boundary-mesh elements. As shall be seen in detail in Section 5, simply prescribing the no-normal flow condition on the linear approximation of the solid curved wall of the channel leads to the presence of resolution-dependent artificial boundary layers and an over-prediction of the surface elevation on the upstream side. In this work, in addition to considering the curvilinear iso- and/or super-parametric elements, we employ a so-called curvature-boundary-condition approach, proposed originally for the Euler equations in [23], for the treatment of the no-normal flow condition on solid curved walls. Such an approach adjusts a component enforcing the boundary condition in a DG formulation so that the physical no-normal flow conditions are better approximated on the straight-sided-element mesh.

The remainder of the paper is organized as follows. In Section 2, we provide a description of the two-dimensional SWE. A DG method for SWE described in [8,9] is briefly summarized in Section 3 (also, we briefly discuss considerations to achieve a so-called well-balanced property in high-order DG schemes in this section). Section 4 contains the detailed account of the two different approaches for treating the no-normal flow condition on a solid wall. In this study, a converging/diverging channel problem is used as a test problem and is described in Section 5.1. Section 5.2 presents results from the study on the flow problem. Conclusions are drawn in Section 6.

2. Governing equations

By assuming a hydrostatic pressure distribution and a uniform velocity profile in the vertical direction, flow in a channel can be modeled by two-dimensional shallow water equations (SWE), also known as the St. Venant equation. The SWE consist of the depth-averaged continuity equation and x- and y-momentum equations written here in a conservative form as:

$$\frac{\partial \mathbf{q}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{q}) = \mathbf{s}(\mathbf{q}, \boldsymbol{x}, t), \quad (\boldsymbol{x}, t) \in \Omega \times [0, \infty) \tag{1}$$

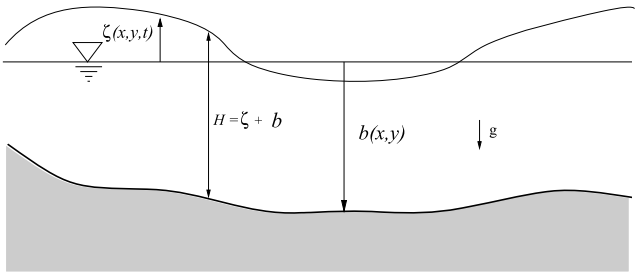

Fig. 1. Schematic diagram of the free surface and bathymetry.

where $\mathbf{q} = (\zeta, uH, vH)^T$ is the vector of conserved variables, $\mathbf{F}(\mathbf{q}) = (\mathbf{f}_1, \mathbf{f}_2)$, which depends on the conserved variables, denotes the flux with

$$\mathbf{f}_1 = \begin{pmatrix} uH \\ u^2H + \frac{1}{2}g(H^2 - b^2) \\ uvH \end{pmatrix} \text{ and } \mathbf{f}_2 = \begin{pmatrix} vH \\ uvH \\ v^2H + \frac{1}{2}g(H^2 - b^2) \end{pmatrix}, \tag{2}$$

and the forcing term $\mathbf{s}$ is given by

$$\mathbf{s} = \left(0,\ g\zeta\frac{\partial b}{\partial x} - \tau_b uH,\ g\zeta\frac{\partial b}{\partial y} - \tau_b vH\right)^T, \tag{3}$$

with

$$\tau_b = C_f\frac{\sqrt{u^2+v^2}}{H} \tag{4}$$

The governing equations are augmented with appropriate initial and boundary conditions. In (2)–(4), ζ denotes the surface elevation measured positive upwards from a specified datum (see Fig. 1 for an illustration), b is the bathymetric depth measured positive downwards from the datum, $H = \zeta + b$ represents the total water column height, u and v are the depth-averaged velocity in the x- and y-directions respectively, g is the magnitude of the gravitational acceleration, τ_b denotes the bottom-stress friction factor, and C_f is the bottom drag coefficient. Here, τ_b is assumed to obey the quadratic friction law (4). Here, we consider the effect of the diffusion of momentum from turbulence negligible and the terms accounting for such an effect are omitted from the equations.

3. Discontinuous Galerkin finite element discretization

3.1. DG discretization

We consider a DG scheme for SWE described in [8,9,12,13,19]. Such a scheme, summarized briefly below, is based on the use of DG for the spatial discretization and the strong-stability-preserving Runge–Kutta (SSPRK) scheme for the time integration.

To discretize (1) in space using DG, the domain Ω is triangulated into a set of non-overlapping triangular elements denoted by $\mathcal{T}_h$. The solution $\mathbf{q}$ is then replaced by a discontinuous approximate solution $\mathbf{q}_h$. The approximate solution is selected so that, when restricted to the element $K \in \mathcal{T}_h$, it belongs to a finite dimensional space $V(K) = (\mathcal{P}^p(K))^3$ where $\mathcal{P}^p(K)$ is a space of polynomials of degree at most p. More precisely, let $\mathbf{\Phi}^K = \{\phi_m^K(\boldsymbol{x})\}_{m=1,\ldots,N_p}$ be a basis of $\mathcal{P}^p(K)$, the solution on the element K is approximated by $\mathbf{q}_h\big|_K = (q_{h,1}^K, q_{h,2}^K, q_{h,3}^K)$ with

$$q_{h,i}^K = \sum_{m=1}^{N_p} \widetilde{q}_{m,i}^K(t)\phi_m^K(\boldsymbol{x}) \tag{5}$$

where $\widetilde{q}_{m,i}^K(t)$ are time-dependent expansion coefficients of the ith-solution component associated with $\phi_m^K(\boldsymbol{x})$. The arising residual in each element is subsequently required to be orthogonal to the local approximation space $V(K)$, yielding the following semi-discrete weak formula

$$\int_K \phi_m^K \frac{\partial q_{h,i}^K}{\partial t} d\boldsymbol{x} - \int_K \nabla\phi_m^K \cdot \mathbf{F}_i(\mathbf{q}_h) d\boldsymbol{x} + \int_{\partial K} \phi_m^K \widehat{\mathbf{F}}_i \cdot \mathbf{n} ds = \int_K \phi_m^K \mathbf{s}_i(\mathbf{q}_h) d\boldsymbol{x},\ \forall \phi_m^K \in \mathbf{\Phi}^K,\ i = 1,2,3 \tag{6}$$

where $\mathbf{n}$ represents the outward-pointing unit normal vector on the element boundary ∂K, $\mathbf{F}_i = \left((\mathbf{f}_1)_i, (\mathbf{f}_2)_i\right)$ denotes the ith-component of the flux term (*e.g.* $\mathbf{F}_1 = (uH, vH)$) and $\mathbf{s}_i$ represents the ith-component of the vector of forcing terms. The

so-called numerical flux $\widehat{\mathbf{F}}$ is employed to resolve the flux $\mathbf{F}(\mathbf{q}_h)$ on the element boundary due to the approximation being discontinuous across the element boundary. The numerical fluxes depend on the traces of $\mathbf{q}_h$ from both sides of the element interface, namely, $(\mathbf{q}^{\text{in}}, \mathbf{q}^{\text{ex}})$. Here, we use $\mathbf{q}^{\text{in}}$ to refer to the value of the solution when approaching a point $\boldsymbol{x} \in \partial K$ from the interior of an element K and $\mathbf{q}^{\text{ex}}$ the value when approaching from the exterior (i.e. from a neighboring element sharing the edge ∂K) of the element K. It is noted that the numerical flux is the term in the DG formula that couples the solution from different elements and is crucial for the stability, convergence, and efficiency of the DG method [5,6]. The numerical flux $\widehat{\mathbf{F}} \cdot \mathbf{n} = \mathcal{F}(\mathbf{q}^{\text{in}}, \mathbf{q}^{\text{ex}}; \mathbf{n})$ can be any locally Lipschitz-continuous, monotone flux that is consistent with the nonlinear flux $\mathbf{F} \cdot \mathbf{n}$ [24]. Unless otherwise indicated, the numerical results shown below are obtained from using the local Lax–Friedrichs flux (LLF)

$$\mathcal{F}(\mathbf{q}^{\text{in}}, \mathbf{q}^{\text{ex}}; \mathbf{n}) = \frac{\mathbf{F}(\mathbf{q}^{\text{in}}) + \mathbf{F}(\mathbf{q}^{\text{ex}})}{2} \cdot \mathbf{n} + \frac{1}{2} C(\mathbf{q}^{\text{in}} - \mathbf{q}^{\text{ex}})$$

where a constant C corresponding to the largest value of the absolute maximum eigenvalue of the normal flux Jacobian matrix, more precisely,

$$C = \max_{\mathbf{s} \in [\mathbf{q}^{\text{in}}, \mathbf{q}^{\text{ex}}]} \left| \lambda \left(n_x \left. \frac{\partial \mathbf{f}_1}{\partial q} \right|_{\mathbf{s}} + n_y \left. \frac{\partial \mathbf{f}_2}{\partial q} \right|_{\mathbf{s}} \right) \right| = \max_{\mathbf{s} \in [(\mathbf{u}, H)^{\text{in}}, (\mathbf{u}, H)^{\text{ex}}]} \left[\left(|\mathbf{n} \cdot \boldsymbol{u}| + \sqrt{gH} \right) \Big|_s \right] \tag{7}$$

where $\lambda(\cdot)$ represents the eigenvalues of the matrix.

In realizing (6), we employ a Dubiner basis [25] on $I = \{(r, s) \mid r, s \geq -1, r + s \leq 0\}$, which is an orthogonal basis and has a dimension of $N_p = (p+1)(p+2)/2$ for a given order p, in defining the test functions ϕ_i^K. More precisely, the test functions are given by

$$\phi_i^K(\boldsymbol{x}) = (\phi_i \circ \boldsymbol{\psi}^{-1})(\boldsymbol{x}), \quad \boldsymbol{x} \in K \tag{8}$$

where ϕ_i denote the Dubiner basis functions and $\boldsymbol{\psi} : I \to K$ maps the coordinates of the reference triangular element I to the physical triangular element K. The area and edge integrals are evaluated using a quadrature rule devised for integration over a triangle [26] and a classical one-dimensional Gauss quadrature, respectively. Note that an $N_p \times N_p$ matrix associated with the first term of (6), is known as an element mass matrix. Since the expansion coordinates from different elements appear only in the numerical flux terms, the global mass matrix associated with the time-derivative term is decoupled-block diagonal and therefore can be inverted in an elementwise fashion. The resulting system of ordinary differential equations is integrated in time using an explicit strong-stability-preserving Runge–Kutta (SSPRK) scheme [27,28] of order $p+1$. The time-step size used in the calculation is selected based on a CFL-type condition

$$\Delta t = \frac{2}{3} C \min_{\boldsymbol{x} \in K, \forall K \in \mathcal{T}_h} \left(\frac{h_K}{(|\mathbf{u}| + \sqrt{gH})(p+1)} \right) \tag{9}$$

where $|\boldsymbol{u}| \equiv \sqrt{u^2 + v^2}$, h_K is a diameter of the element K, and C is a constant. In numerical calculations, we use the diameter of the largest circle inscribed in a triangle to represent the element size.

3.2. Element types

In this work, we consider straight-sided triangular elements as well as curved-edged triangular elements. For the straight-edged element, we use in (8) a standard linear affine mapping for $\boldsymbol{\psi}$ [29], more precisely,

$$\boldsymbol{\psi}(\boldsymbol{r}) = \sum_{i=1}^{3} L_i(\boldsymbol{r}) \boldsymbol{x}_i^K, \; L_1 = -\frac{(r+s)}{2}, \; L_2 = \frac{(r+1)}{2}, \text{ and } L_3 = \frac{(1+s)}{2} \tag{10}$$

where $\boldsymbol{x}_i^K$ denotes a coordinate of the ith-vertex of the element and L_i is a linear Lagrange interpolation basis function on I.

For the curved-edged triangular elements, the geometry of K is represented by a polynomial interpolation of degree $\widetilde{p}$ on I, more precisely,

$$\boldsymbol{x}_K(r, s) = \boldsymbol{\psi}(\boldsymbol{r}) = \sum_{i=1}^{N} L_i(\boldsymbol{r}) \boldsymbol{x}_i^K \tag{11}$$

where $L_i(r, s)$ are the two-dimensional Lagrange polynomial basis functions on I associated with the nodal points $\boldsymbol{r}_i$, $\{\boldsymbol{x}_i^K\}_{i=1,N}$ are the given points of the physical element K, and $N = (\widetilde{p}+1)(\widetilde{p}+2)/2$ (see Fig. 2 for an illustration). The Lagrange polynomial basis functions are unity only at their associated nodal point and zero at other nodal points, i.e. $L_i(r_j) = \delta_{i,j}$, where $\delta_{i,j}$ is the Kronecker delta. Therefore, the above mapping is exact at the grid points $\boldsymbol{x}_i^K$. For $\widetilde{p} = p$, where p is the degree of polynomial used in the discretization, the mapping (11) is known as an iso-parametric interpolation, for $\widetilde{p} > p$ a super-parametric interpolation, and $\widetilde{p} < p$ a subparametric interpolation. Indeed, the mapping for the straight-edged element (10) corresponds to the case where $\widetilde{p} = 1$.

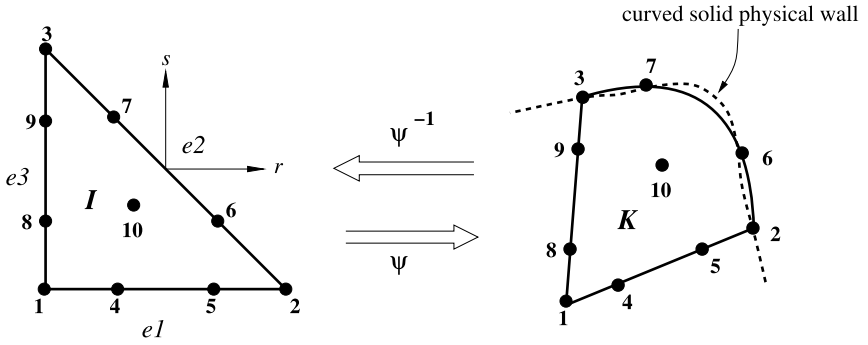

Fig. 2. Schematic diagram of the iso-parametric mapping for $\widetilde{p}=3$. $I=\{(r,s)\mid(r,s)\le -1, r+s\le 0\}$ and K is the physical element. This example shows the case where the edge joining the vertices labeled with 2 and 3 is a curved edge.

Note that it is important to choose the nodal distribution $\{\boldsymbol{r}_i\}$, especially for high $\widetilde{p}$, in a way that the interpolation does not suffer from the Runge phenomenon, i.e. pronounced oscillations occurring when using a high-order interpolation polynomial of the equidistant grids (see [7] for devising such a nodal set and for a systematic way of constructing the Lagrange polynomial basis). A mesh generator utilized in this work provides data for linear finite elements; therefore, information of the grid points $\{\boldsymbol{x}_i\}$ for high-order curvilinear elements is not readily available. In this work, we obtain $\{\boldsymbol{x}_i\}$ from the strategy described in [7] which involves deforming the straight edge(s) and redistributing the interior nodes of the straight-edged elements (the grid points of the straight-edged element are determined by the mapping (10)) to avoid severe distortion of the grid points.

Note that the area integrals in (6) are evaluated by numerically integrating the equivalent transformed integrals in the (r,s) coordinates, more specifically,

$$\int_K f\,d\boldsymbol{x}=\int_I f\,\det(\mathbf{J})\,d\boldsymbol{r},\quad \mathbf{J}=\frac{\partial(x,y)}{\partial(r,s)} \tag{12}$$

$$\int_K f\frac{\partial g}{\partial x}d\boldsymbol{x}=\int_I f\left(\frac{\partial r}{\partial x}\frac{\partial g}{\partial r}+\frac{\partial s}{\partial x}\frac{\partial g}{\partial s}\right)\det(\mathbf{J})\,d\boldsymbol{r}=\int_I f\left(\frac{\partial y}{\partial s}\frac{\partial g}{\partial r}-\frac{\partial y}{\partial r}\frac{\partial g}{\partial s}\right)d\boldsymbol{r} \tag{13}$$

where $\mathbf{J}$ denotes the Jacobian matrix and the derivatives in (x,y) and (r,s) are related through $(\partial/\partial x,\partial/\partial y)^T=\mathbf{J}^{-T}(\partial/\partial r,\partial/\partial s)^T$. The edge integrals are evaluated in a similar manner. Note that the unit vector normal to the element edge is present in the edge integral term of the DG weak formula. The unit normal vector to the edge of the physical element can be calculated from the formula below

$$\hat{\mathbf{n}}=(\hat{n}_x,\hat{n}_y)=\begin{cases}\dfrac{1}{J_1}(\partial y/\partial r,-\partial x/\partial r), & \text{on } \partial K_1,\\ \dfrac{1}{J_2}(-\partial y/\partial r+\partial y/\partial s,\partial x/\partial r-\partial x/\partial s), & \text{on } \partial K_2,\\ \dfrac{1}{J_3}(-\partial y/\partial s,\partial x/\partial s), & \text{on } \partial K_3\end{cases} \tag{14}$$

where J_i is the factor normalizing the vector so that $|\hat{\mathbf{n}}|=1$ and in fact it is the Jacobian for the mapping along the edge ∂K_i $(i=1,2,3)$ of a triangle. Note that the edge i is the edge connecting the vertices ith and $(\mathrm{mod}(i,3)+1)$th (see Fig. 2 for illustration), where mod(,) denotes the modulo operator. For the straight-sided elements, the mapping (10) has a constant Jacobian. Since the Dubiner basis is orthogonal, the element mass matrix is diagonal in this case and therefore can be trivially inverted. Generally, the Jacobian of the mapping (11) is not a constant for $\widetilde{p}>1$. As a consequence, the element mass matrix is no longer diagonal for the curvilinear elements and the non-constant Jacobian (and also a non-constant unit normal vector on the curved edge) dictates the need for using a more accurate quadrature rule for an accurate integration of the integral terms. Here, for the straight-edged elements, the area integrals are evaluated using a $2p$-order accurate quadrature rule and the edge integrals are calculated using a $p+1$-point one-dimensional Gauss quadrature. For curved-edged elements, the quadrature rule integrating exactly polynomials of degree $2p+2\widetilde{p}$ is used in the area integrals and the $(p+\widetilde{p}+1)$-point 1-D Gauss quadrature is employed in the edge integrals. It is noted that the quadrature rules used here are selected such that they integrate the weak formula exactly at the still water state ($\zeta_h=const$, $(\mathbf{u}H)_h=\mathbf{0}$) to ensure that the scheme preserves the solution of the *lake-at-rest* problem, i.e. the scheme is well-balanced.

3.3. Remark on the well-balanced property

In non-flat bed problems, one concern of DG or FV methods that are based on the conservative form of the SWE involves their ability to preserved the lake-at-rest solution

$$\boldsymbol{u}(\boldsymbol{x}) = 0 \quad \text{and} \quad \zeta = H(\boldsymbol{x}) - b(\boldsymbol{x}) = const \tag{15}$$

in the time marching process. Numerical schemes preserving such a state are termed as well-balanced schemes [30,31]. To achieve the well-balanced property, numerical schemes are devised so that the gravity term cancel out the bed term for a given approximate solution of the state at rest solution. In this work, the well-balanced property is attained by following a treatment outlined in [18], which is suitable for the conforming-order and conforming-mesh discretization. In brief, such a treatment consists of (i) replacing the bathymetric depth $b(\boldsymbol{x})$ with a piecewise continuous interpolant of the same degree with the DG polynomial and (ii) exact realization of the DG weak formula at the still-water state. For the steady state at rest solution $\zeta_h = const$, $(\boldsymbol{u}H)_h = \boldsymbol{0}$, and $H_h = \zeta_h + b_h$, the former ensures that $H_h - b_h = const$ everywhere in the element and H_h is single-valued along the inter-element boundary. This leads to the following weak formula

$$\int_K \frac{\partial \mathbf{q}_h}{\partial t} v_h d\boldsymbol{x} = \begin{pmatrix} 0 \\ \int_K \frac{1}{2} g H^* \frac{\partial v_h}{\partial x} d\boldsymbol{x} - \int_{\partial K} \frac{1}{2} g H^* v_h ds + \int_K g \zeta_h v_h \frac{\partial b_h}{\partial x} d\boldsymbol{x} \\ \int_K \frac{1}{2} g H^* \frac{\partial v_h}{\partial y} d\boldsymbol{x} - \int_{\partial K} \frac{1}{2} g H^* v_h ds + \int_K g \zeta_h v_h \frac{\partial b_h}{\partial y} d\boldsymbol{x} \end{pmatrix}, \quad H^* \equiv H_h^2 - b_h^2 \tag{16}$$

where v_h are the test functions. The integrand of each edge integral is a direct consequence of the conservative property of the numerical flux. By integrating the first term of the right hand side by-parts and using $H_h = const + b_h$, it can be verified that the right-hand-side term vanishes, hence yielding the well-balanced property. The requirement (ii) ensures the cancellation of terms is also satisfied in the numerical calculation. A question that may arise naturally concerns whether the requirement (ii) can be achieved in the curvilinear-element case. It can be shown that, owing to (11) being polynomials, an integrand in each integral term of (16) is indeed either a polynomial in (r, s) or in a 1-D parametric variable. Therefore, the DG weak formula at the still water state can be calculated exactly by utilizing sufficiently accurate quadratures. In Appendix A, we demonstrate, through a test problem, the well-balanced property of the DG scheme with curvilinear elements.

4. Implementation of no-normal flow boundary conditions

In this study, we consider open channel flow problems with solid curved walls. On the solid walls, we prescribe the no-normal flow boundary condition also known as the inviscid wall boundary conditions. This condition indicates there is no flow across the wall, i.e. the normal component of velocity vanishes at the wall

$$\boldsymbol{u} \cdot \boldsymbol{N} = 0, \tag{17}$$

where $\boldsymbol{N}$ denotes the unit outward vector normal to the physical boundary. Since grid generators typically generate meshes consisting of only straight-sided elements, the boundary of the computational domain is therefore a piecewise linear approximation of the physical domain (the approximate boundary is often unfortunately the only information available). The no-normal flow condition is then conventionally treated by simply specifying that the normal component of velocity is zero on straight-sided segments of the approximate boundary. This treatment is generally sufficient for finite volume methods and for continuous Galerkin finite element methods with linear elements in that the order of accuracy is not deteriorated by the piecewise linear representation of the boundary. As shall be seen below in Section 5.2, this is not sufficient for DG solution even in the linear element case. The issue can be circumvent by a technique that accounts for errors introduced by the geometric approximation or an approach that accurately represents the geometry.

Below, we first describe the conventional approach in imposing the no-normal flow condition in the DG scheme. Subsequently, we discuss the so-called curvature-boundary-condition (CBC) approach [23] for implementing the no-normal flow conditions on straight-sided element meshes and thereafter realization of (17) in the curvilinear element solution.

4.1. Conventional approach

In the conventional approach, one prescribes the condition

$$\boldsymbol{u} \cdot \mathbf{n} = 0 \tag{18}$$

on an element edge approximating the no-normal flow curved boundary where $\mathbf{n}$ is the unit vector normal to the element edge (see Fig. 3 for illustration). To implement this condition in the DG scheme (6), we use the values of the numerical fluxes that are a result of setting the fictitious exterior states at integration points as follows

$$(\boldsymbol{u}H)^{\text{ex}} \cdot \mathbf{n} = -(\boldsymbol{u}H)^{\text{in}} \cdot \mathbf{n}, \quad (\boldsymbol{u}H)^{\text{ex}} \cdot \boldsymbol{\tau} = (\boldsymbol{u}H)^{\text{in}} \cdot \boldsymbol{\tau}, \quad H^{\text{ex}} = H^{\text{in}} \tag{19}$$

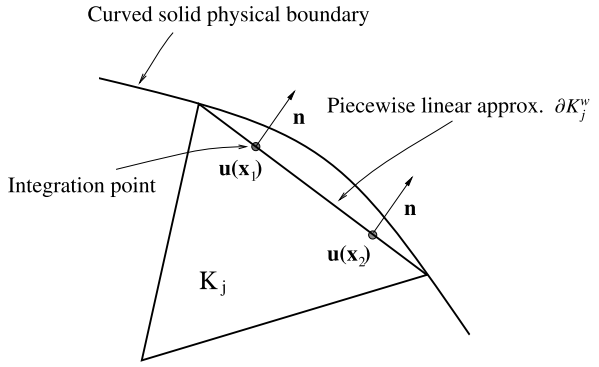

Fig. 3. Straight-sided element with an edge approximating a curved boundary.

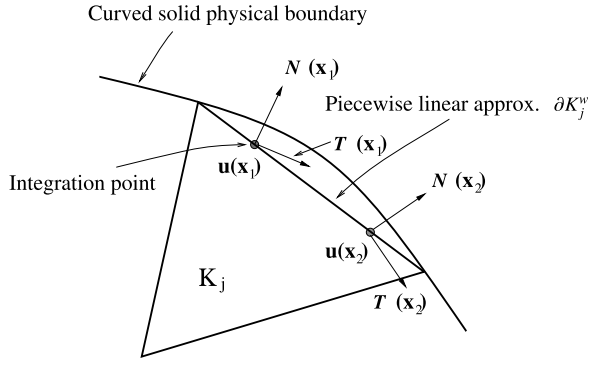

Fig. 4. Curvature-boundary-condition approach for two integration points.

where $\boldsymbol{\tau}$ is the unit-tangential vector of the given edge ($\mathbf{n} = (n_x, n_y)$, $\mathbf{n} \cdot \boldsymbol{\tau} = 0$ and $\mathbf{n} \times \boldsymbol{\tau} = \widehat{\mathbf{k}}$). The velocity of the exterior state given above is the reflection of the interior velocity with respect to $\boldsymbol{\tau}$. Note that this results in the exterior state with the following velocity components

$$u^{\text{ex}} = \left(n_y^2 - n_x^2\right) u^{\text{in}} - 2n_x n_y v^{\text{in}}, \quad v^{\text{ex}} = \left(n_x^2 - n_y^2\right) v^{\text{in}} - 2n_x n_y u^{\text{in}} \tag{20}$$

It can be verified that the exterior data leads to $\widehat{\mathbf{F}} \cdot \mathbf{n}$ with a vanishing component for the continuity equation (note that $\widehat{\mathbf{F}} \cdot \mathbf{n}$ is calculated by passing $(\mathbf{q}^{\text{in}}, \mathbf{q}^{\text{ex}}; \mathbf{n})$ to a Riemann solver). This indicates that the condition (18) is weakly enforced.

4.2. Curvature-boundary-condition (CBC) approach

The CBC approach, devised by [23] for Euler equations, is based on the observation that when a computational boundary does not coincide with the physical boundary, imposing the no-normal flow through the physical boundary implies permitting some flow to enter or leave the computational boundary. Therefore, this approach assumes no-normal flow through the physical boundary in the near vicinity of the surface and imposes at each integration point on the approximate boundary ∂K^w the following condition

$$\boldsymbol{u} \cdot \boldsymbol{N} = 0, \quad \boldsymbol{x} \in \partial K^w \tag{21}$$

where $\boldsymbol{N} = (N_x, N_y)$ is the unit normal to the physical boundary (see Fig. 4 for illustration).

By following [23], the CBC approach for realizing (21) in the DG scheme is carried out by determining the fictitious exterior state at an integration point from

$$(\boldsymbol{u}H)^{\text{ex}} \cdot \boldsymbol{N} = -(\boldsymbol{u}H)^{\text{in}} \cdot \boldsymbol{N}, \quad (\boldsymbol{u}H)^{\text{ex}} \cdot \boldsymbol{T} = (\boldsymbol{u}H)^{\text{in}} \cdot \boldsymbol{T}, \quad H^{\text{ex}} = H^{\text{in}} \tag{22}$$

where $\boldsymbol{T}$ denotes the unit vector tangential to the physical boundary ($\boldsymbol{T} \cdot \boldsymbol{N} = 0$ and $\boldsymbol{T} \times \boldsymbol{N} = \widehat{\mathbf{k}}$). The velocity of the exterior state given above is the reflection of the interior velocity with respect to $\boldsymbol{T}$. The interior state, the fictitious exterior state, and the unit vector normal to the approximate boundary $\mathbf{n}$ are subsequently used in evaluating the numerical flux $\widehat{\mathbf{F}} \cdot \mathbf{n}$. It can be verified that, for non-matching $\boldsymbol{N}$ and $\mathbf{n}$, the continuity-equation component of $\widehat{\mathbf{F}} \cdot \mathbf{n}$ does not vanish; this implies that the CBC approach leads to non-conservation of mass in the computational domain.

Note that a procedure for approximating the vector $\boldsymbol{N}$ from a given straight-sided mesh is described in [23]. In this study, since the analytical expressions of the walls are available (see Section 5.1), we compute the unit normal vector $\boldsymbol{N}$

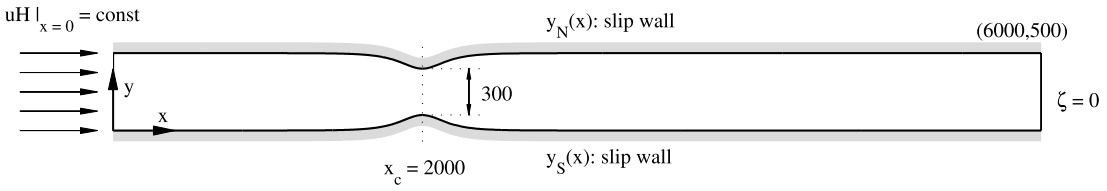

Fig. 5. Configuration of the converging–diverging channel problem.

directly from the analytical expression. Although not reported here in detail, from our numerical experiments with a channel problem with more complicated-geometry walls, we note that using a cubic spline interpolation to fit the boundary curve from a given mesh appears to yield a satisfactory approximate solution.

4.3. Curvilinear element

In the curvilinear element, the boundary condition on the solid curved wall is given by

$$\boldsymbol{u} \cdot \hat{\mathbf{n}} = 0 \tag{23}$$

where $\hat{\mathbf{n}}$ is the spatially-varying unit vector normal to the curved edge approximating the physical solid curved wall. The procedure, similar to that described in Section 4.1 (with the spatially-varying unit normal vector $\hat{\mathbf{n}}$), is sufficient to achieve the condition (23) in the approximate solutions. The unit normal vector at the (mapped) quadrature points on the element edge can be computed by using the formula (14).

5. Application of DG to flow in channels

5.1. Problem description

The converging/diverging channel test problem is illustrated in Fig. 5. The channel has flat bathymetry and is 6000 m long and has variable width which is 500 m wide at the eastern and western ends and 300 m wide at the narrowest part. The narrowest part is located at 2000 m from the western end. The profile of the northern and southern walls are given analytically by

$$\begin{aligned} y_N(x) &= Y_{NW} - z_{\max}\operatorname{sech}\left(4(x - X_c)/L_m\right) \quad \text{and} \\ y_S(x) &= Y_{SW} + z_{\max}\operatorname{sech}\left(4(x - X_c)/L_m\right), \end{aligned} \tag{24}$$

respectively, where $L_m = 500$, $z_{\max} = 100$, $X_c = 2000$, $Y_{NW} = 500$ and $Y_{SW} = 0$. In numerical simulations, we use the quiescent state as the initial condition, more precisely,

$$\boldsymbol{u}H = (0, 0)^T, \ \zeta = 0, \ \text{at } t = 0.$$

The no-normal flow condition is prescribed on the northern and southern walls. The boundary conditions at the western and eastern ends are set to

$$(uH, vH) = (q_w, 0) \ \text{ on } (x = 0, y, t) \quad \text{and} \quad \zeta = 0 \ \text{ on } (x = 6000, y, t),$$

respectively, where q_w is a constant unit-width discharge. These boundary conditions describe constant discharge of flow to an open ocean. A value of the unit-width discharge q_w and the constant bathymetric depth b are chosen in a way that flow is sub-critical everywhere. The bottom drag coefficient C_f is set to a constant value throughout the domain. Note that all numerical results reported below, unless otherwise stated, are for the problem with $b(x, y) = 10$ (m), $q_w = 5$ (m^2/s), $g = 9.81$ (m/s^2) and $C_f = 0.0025$.

Numerical solutions are computed on a sequence of refined meshes. Fig. 6 shows the coarsest mesh (h-mesh) and the two finer meshes ($h/2$- and $h/4$-meshes). Note that the h-mesh has approximately five to six elements across the channel width. In these meshes, the northern and southern walls are a piecewise linear approximation of the exact wall profile (24); they are more accurate as the resolution increases. We consider DG polynomials of orders ranging from $p = 1$ to 3. The $(p+1)$th-order, $(p+2)$-stage SSPRK is used in the time integration with the variable time step size Δt adjusted based on criteria (9). We set the constant C in (9) to a small value, more precisely to 0.25, in order to keep the error from the temporal discretization small in comparison to the spatial discretization error. To avoid introducing sudden forcing in the simulations, the discharge boundary value is gradually introduced as $q_w r(t)$, where $r(t)$ denotes a so-called ramping function; here, we employ a hyperbolic tangent ramping function $r(t) = \tanh(2t/D_r)$ with a ramping parameter $D_r > 0$ for this purpose. The value of D_r is set to 0.08 day in the calculations. The simulations are carried out until $t_f = 4$ days is reached.

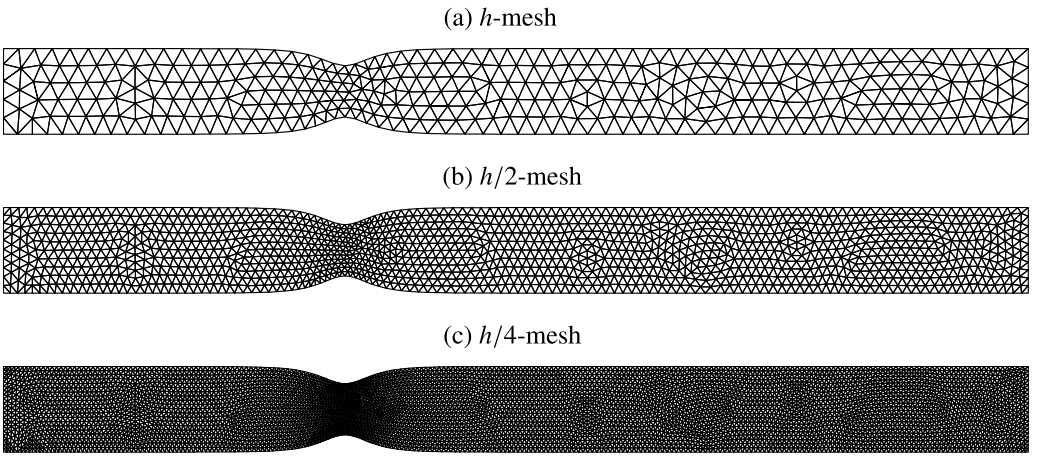

Fig. 6. Computational meshes used in the numerical tests: (a) h-mesh, $N_{el} = 656$, (b) $h/2$-mesh, $N_{el} = 2624$, and (c) $h/4$-mesh, $N_{el} = 10\,496$.

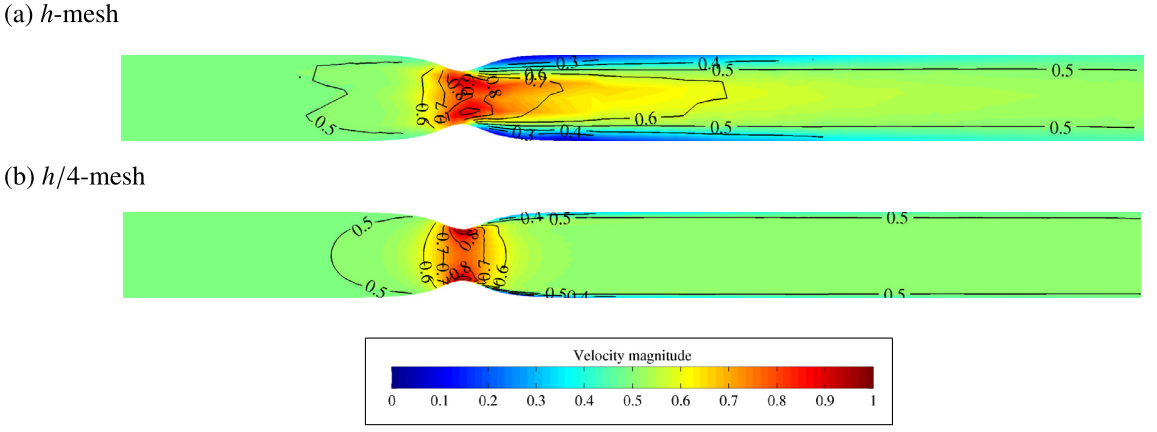

Fig. 7. Velocity magnitude $|\boldsymbol{u}|$ at $t = 2$ days from using the LLF DG scheme with $p = 1$ and the conventional implementation of the no-normal flow boundary condition. Solutions under h-refinement: (a) h-mesh; (b) $h/4$-mesh.

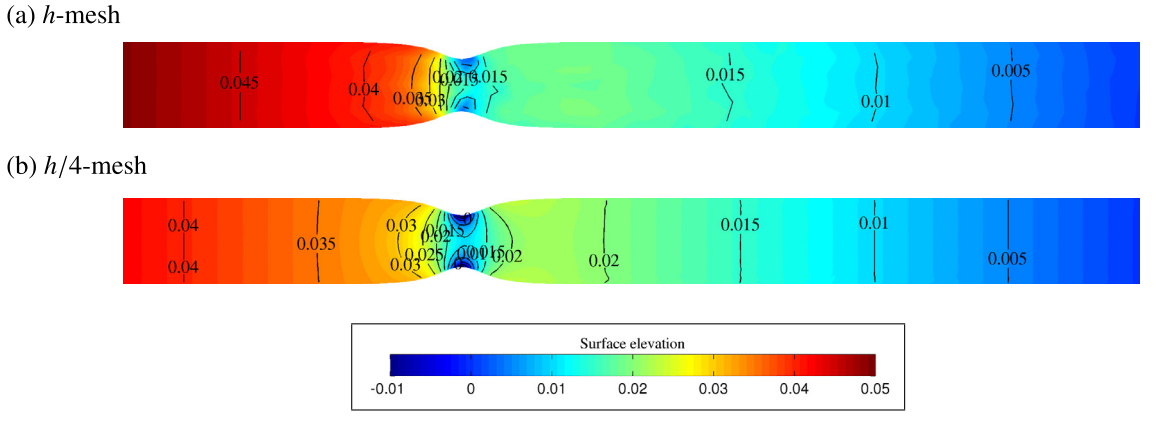

Fig. 8. Surface elevation ζ at $t = 2$ days obtained from using the LLF DG scheme with $p = 1$ and the conventional implementation of the no-normal flow boundary condition. Solutions under h-refinement: (a) h-mesh; (b) $h/4$-mesh.

5.2. Numerical results

5.2.1. Results from the conventional implementation of the no-normal flow boundary condition

We first report on the numerical results using the conventional approach for implementing the no-normal flow boundary conditions. The velocity magnitude $|\boldsymbol{u}| = \sqrt{u^2 + v^2}$ and the surface elevation ζ at $t = 2$ days obtained from using the LLF DG scheme with $p = 1$ on the h- and $h/4$ mesh are plotted in Fig. 7 and Fig. 8, respectively. Note that boundary layers can be

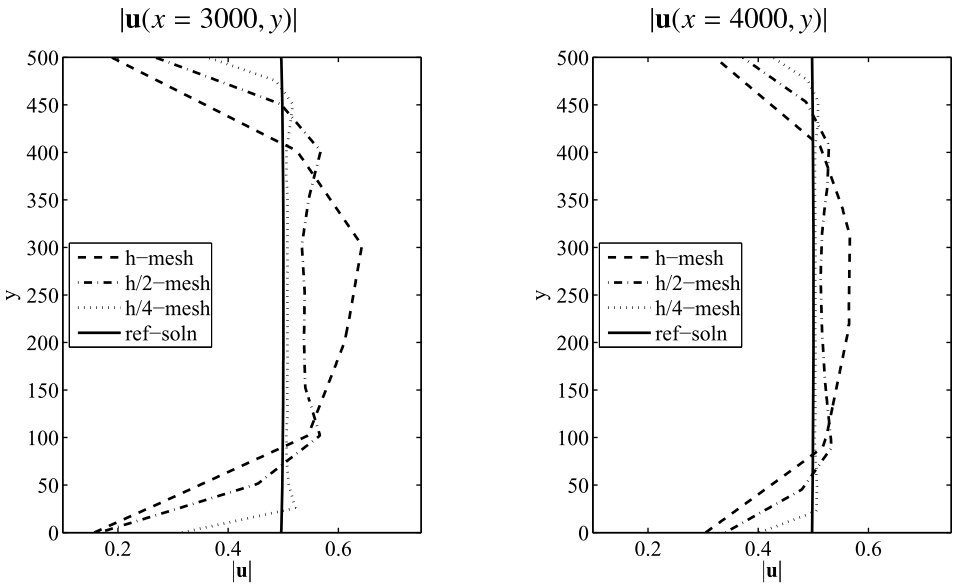

Fig. 9. Solution at $t = 2$ days from using the conventional implementation of the no-normal flow boundary condition. Velocity magnitude at the section $x = 3000$ (left) and $x = 4000$ (right) computed using DG schemes with $p = 1$ on meshes of different resolutions. A solid line depicts the data of the reference solution.

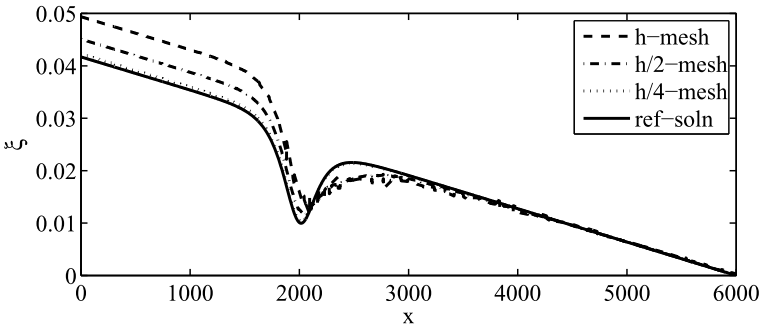

Fig. 10. Solution at $t = 2$ from using the conventional implementation of the no-normal flow boundary condition. Surface elevation along the section $y = 250$, $\zeta(x, y = 250)$ computed using DG schemes with $p = 1$ on meshes of different resolutions. A solid line depicts the data of the reference solution.

clearly observed in the velocity plots (Fig. 7). These types of boundary layers also appear in the previous works of [21] and [20] which use the straight-edged elements in their calculations. Such boundary layers emerge from the channel throat and persist far downstream. In Fig. 9, we plot $|\boldsymbol{u}(3000, y)|$ and $|\boldsymbol{u}(4000, y)|$, the velocity magnitude at the cross section along $x = 3000$ and $x = 4000$, respectively. For comparison, we also plot in these figures results from the reference calculation which is computed by the iso-parametric element DG with $p = 3$ on a high-resolution mesh (the results of curvilinear element calculations will be discussed below in Sections 5.2.2 and 5.2.3). Notice that the thickness of the boundary layers decreases as the grid size decreases and it appears to be approximately one-grid size in width. These observations provide convincing evidence that the boundary layers appearing in the solution are in fact artificial.

The presence of boundary layers, as expected, has an unfavorable effect on the accuracy of the water surface elevation level. In this problem setup, the artificial boundary layers lead to an over-prediction of the surface elevation upstream. We plot in Fig. 10 the surface elevation along the channel's horizontal centerline $\zeta(x, y = 250)$. From the surface elevation plots (see Figs. 8 and 10), it is evident that the level of surface elevation upstream of the channel throat obtained from the coarse-mesh solution is visibly higher than those from the solutions obtained on the finer meshes. The connection between the presence of the artificial boundary layers and over-prediction of the surface elevation level may be explained as follows. With the boundary layers, the downstream portion of the channel is effectively narrowed; this results in higher velocity away from the channel walls in such areas. For this test problem, a steady state solution is a balance of the pressure gradient, convection, and the bottom friction terms. It is obvious that the higher velocity magnitude implies a larger quadratic friction term and thus a higher gradient of the surface elevation; as a consequence, this leads to greater surface elevation on the upstream side. This can be easily seen in the one-dimensional case in which one has $g\frac{\partial H}{\partial x} = -C_f\frac{|u|u}{H}$

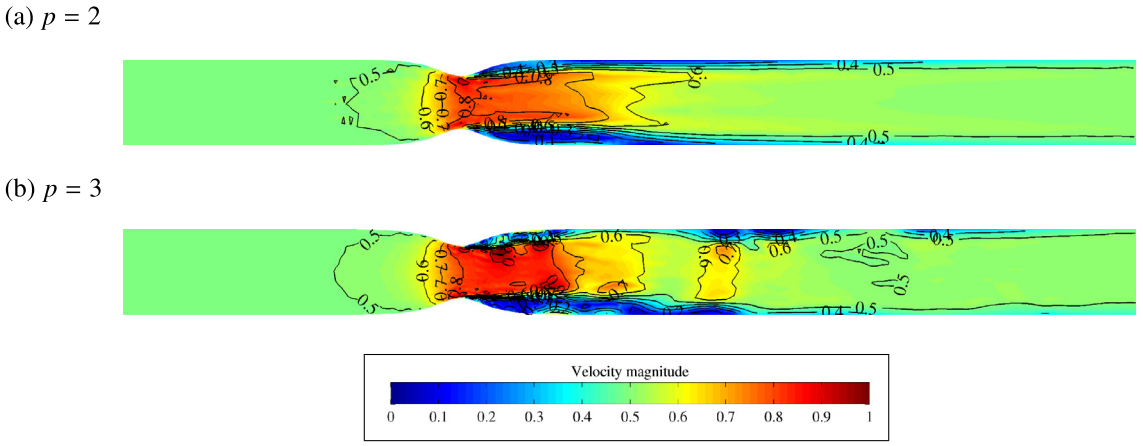

Fig. 11. Velocity magnitude $|\boldsymbol{u}|$ at $t = 2$ days from using the LLF DG scheme and the conventional implementation of the no-normal flow boundary condition. Solutions under p-refinement on h-mesh: (a) $p = 2$; (b) $p = 3$.

(ignoring the convection term for simplicity of explanation) at steady state; it can be easily checked that the gradient $-g\frac{\partial H}{\partial x}$ increases as u increases.

In the above discussion, the resolution of the computation is changed by refining the mesh sizes. The resolution can be also be refined by changing the order of DG polynomials, p, which yields great benefit for a problem with a smooth solution (see for example [17,18] for a performance study of high-order DG methods). Fig. 11 shows the plots of velocity magnitude at $t = 2$ days under p-refinement on the h-mesh. For the LLF DG scheme, the approximate solution from the high-order schemes ($p > 1$) contains visibly stronger, more complicated-structure boundary layers than that with $p = 1$ (see Figs. 11 and 7). Qualitatively, the solution appears to be increasingly poor as p increases. In fact, the LLF DG solution becomes unsteady for $p = 3$.

Although not discussed here in detail, when the upwind-type numerical fluxes such as the Roe flux and Harten–Lax–van Leer-Contact flux are used in the calculations, the artificial boundary layers, although still clearly visible, become noticeably weaker than those appearing in the solutions with the LLF flux shown above. In fact, when using the upwind-type fluxes, the velocity magnitude of the high-order scheme reaches steady state with decreasing thickness of the boundary layers as p increases. This indicates, as one might expect, an advantage of the upwind-type numerical flux over the LLF flux for channel flow applications.

5.2.2. Results from the CBC implementation and curvilinear elements

Below, we report the numerical results from using the CBC approach for implementing the no-normal flow boundary condition and from using curvilinear elements. In curvilinear-element calculations, the iso-parametric or super-parametric elements are used only along the curved walls of the channel. Unless otherwise specified, results reported below for the curvilinear element solution are obtained from the iso-parametric elements ($\widetilde{p} = p$) for high-order elements ($p > 1$) and the super-parametric with $\widetilde{p} = 2$ for the linear elements. Indeed, numerical solutions from using the CBC approach and the curvilinear element are in qualitatively good agreement and virtually indistinguishable to the naked eyes when plotted. We therefore discuss below only the results from the CBC approach. Observations made below for the CBC approach are also what observed in the curvilinear-element solution. The close inspection of errors in these approaches is presented in the subsequently section (Section 5.2.3) which examines convergence rates.

Fig. 12 shows snap shots of the velocity magnitude at $t = 2$ days using the CBC approach in the LLF DG method with $p = 1$ on the h- and $h/4$-mesh. The velocity profiles plotted appear to visually contain no boundary layer. The plots in Fig. 13, which show the velocity magnitude at the cross section $x = 3000$ and $x = 4000$, exhibit no boundary layer in the DG solution. The surface elevation at $t = 2$ days from the CBC approach and the linear-element DG are shown in Fig. 14; Fig. 15 plots the surface elevation along the channel horizontal centerline $\zeta(x, y = 250)$.

It is noted from these plots that the level of surface elevation upstream of the channel throat from the coarse-mesh calculations differ only slightly from the finer-mesh calculations. Indeed, in comparison to ζ obtained from the conventional no-normal flow implementation presented in the previous subsection (see Fig. 8 and Fig. 10), the coarse-mesh (h-mesh) solution of the CBC approach compares well with the fine-mesh ($h/4$-mesh) solution of the conventional no-normal flow implementation.

Next we examine the numerical results of the CBC approach when using the high-order DG elements ($p > 1$). Fig. 16 shows the velocity magnitude obtained from the high-order LLF DG scheme. Clearly, the solution improves qualitatively as the order p increases. No boundary layer is present in these high-order DG solutions. Note that, unlike in the conventional no-normal flow approach, the solution of the high-order LLF DG scheme with the CBC approach reaches steady state for all orders of p considered.

(a) h-mesh

(b) $h/4$-mesh

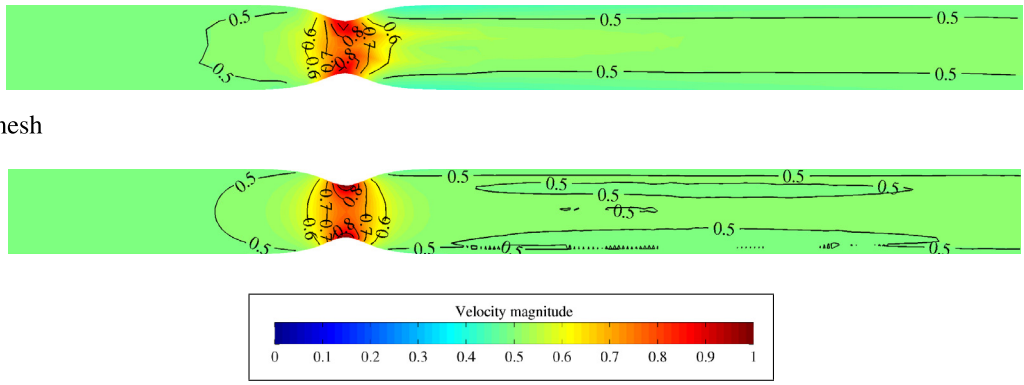

Fig. 12. Velocity magnitude $|\boldsymbol{u}|$ at $t = 2$ days from using the LLF DG scheme with $p = 1$ and the CBC implementation for the no-normal flow boundary conditions. Solution under h-refinement: (a) h-mesh, (b) $h/4$-mesh.

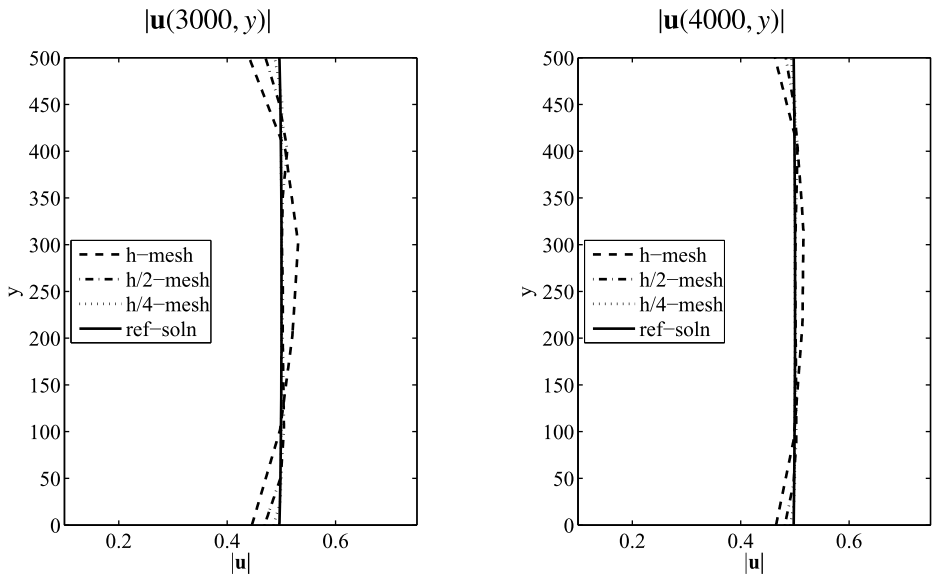

Fig. 13. Solution at $t = 2$ days from using the CBC implementation of the no-normal flow boundary condition. Velocity magnitude at the section $x = 3000$ (left) and $x = 4000$ (right) computed using DG schemes with $p = 1$ on meshes of different resolutions. A solid line depicts the data of the reference solution.

(a) h-mesh

(b) $h/4$-mesh

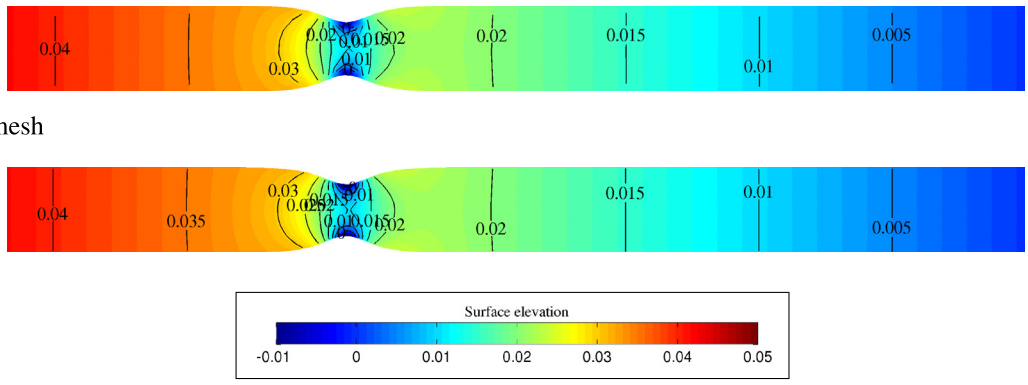

Fig. 14. Surface elevation ζ at $t = 2$ days obtained from using the LLF DG scheme with $p = 1$ and the CBC implementation of no-normal flow boundary conditions. Solutions under h-refinement: (a) h-mesh; (b) $h/4$-mesh.

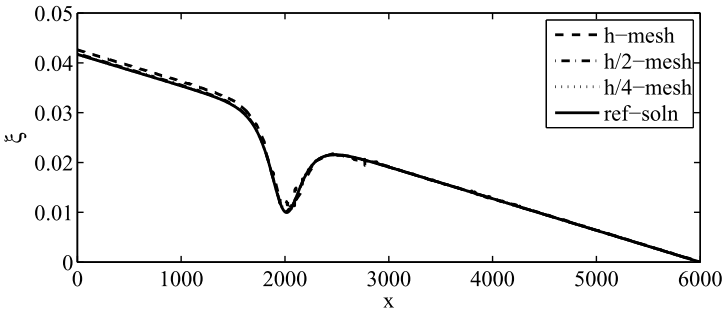

Fig. 15. Solution at $t = 2$ days from using the CBC implementation of the no-normal flow boundary condition. Surface elevation along the section $y = 250$, $\zeta(x, y = 250)$ computed using DG schemes with $p = 1$ on meshes of different resolutions. A solid line depicts the data of the reference solution.

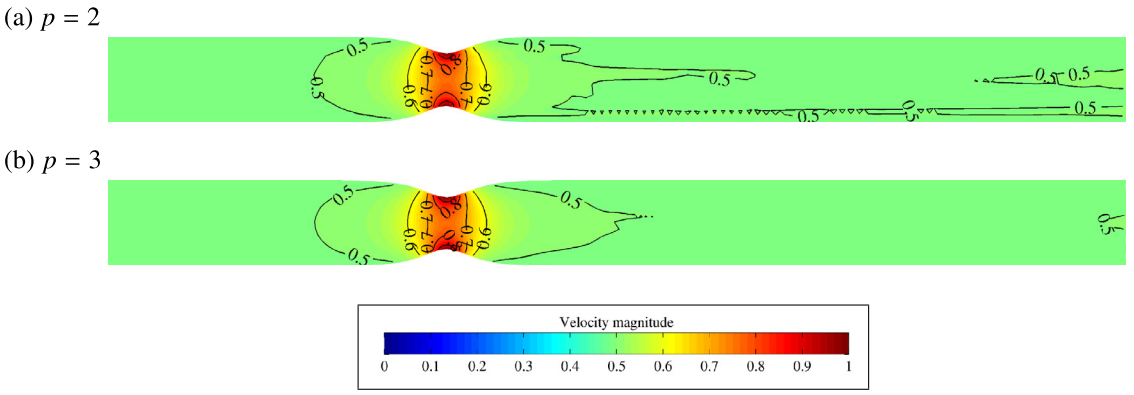

Fig. 16. Velocity magnitude $|\boldsymbol{u}|$ at $t = 2$ days obtained from using the LLF flux and the CBC approach for no-normal flow boundary conditions. Results are solved on h-mesh with (a) $p = 2$, and (b) $p = 3$.

5.2.3. Numerical convergence

Since this problem has no analytical solution, we use an isoparametric-element solution obtained on the $h/8$-mesh (note that the $h/4$-mesh is a de-refinement of the $h/8$ mesh) and $p = 3$ as a reference solution for computing errors in the solution. Errors, measured in the L_2-norm, in the water level ζ_h and the x-directed momentum component $(uH)_h$ are plotted in Fig. 17 and in Fig. 18, respectively. In these figures, the errors are plotted against the discretization resolution in the log–log scale. Here, $\sqrt{N_p N_{el}}$, where $N_p = (p+1)(p+2)/2$ and N_{el} is the number of elements in the computational mesh, is used as an estimate of an inversion of the discretization grid size. The errors in the curvilinear-element DG solutions are depicted with the solid lines. The dashed lines show the errors in the DG solution with the CBC approach. We also include in the plots of the errors in the DG solution using the conventional no-normal flow implementation; the dotted lines, unfilled circles, and unfilled square show errors in the solution from using the linear, quadratic, and cubic elements, respectively. The results from both p- and h-refinement are included these figures (the value of p is placed to the right of each curve). Clearly, the level of errors in the solution using the conventional approach is very poor in comparison to the other two approaches; in addition, this approach fails to converge in the p-refinement, i.e. when varying p while holding the mesh resolution unchanged. It can be observed that the approximate solution becomes more accurate as the finer meshes are used in the calculation, i.e. as the mesh size decreases. In addition, the curvilinear-element DG solution yields significantly more accurate solution for a given discretization grid side as the high-order elements are used, i.e. as p increases. In the CBC approach, the benefit of this type from p-refinement can be clearly noticed as p increases from 1 to 2; however, very little benefit is observed from increasing p from 2 to 3.

Table 1 reports the overall rate of convergence of the approximate solution. Note that, theoretical estimates in [32] show that the LLF DG scheme for scalar conservation laws has the optimal rates of convergence of $p+1/2$. The curvilinear-element DG solution exhibits a numerical convergence of rate between $O(h^{p+1/2})$ and $O(h^{p+1})$. For $p = 1$, and 2, the DG method with the CBC approach yields the approximation solution that converges at the rate of $p + 1/2$ to $p + 1$. However, for $p = 3$, this approach converges at a rate of only approximately 2.1, significant lower than the expected order. Note that although factoring in the effect of the curved boundary to determine the exterior state so that the resulting numerical flux yields a better approximation of the physical no-normal condition, the CBC approach is still based on the use of an approximate boundary (which is a linearly truncated version of the physical boundary) in the calculation. Intuitively, errors introduced by not matching the computational domain to the physical domain could increasingly become a source of major error as p increases. It is thus logical for the CBC no-normal flow approach to not perform in an ideal manner in the p-refinement.

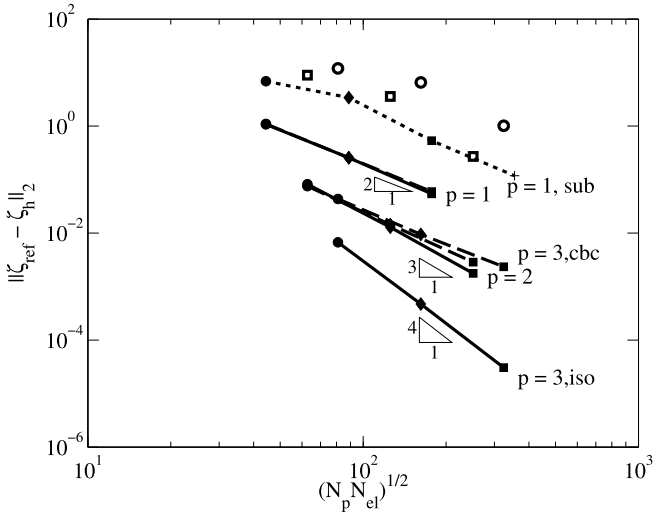

Fig. 17. L_2-errors in ζ_h as a function of $(1/h) \sim \sqrt{N_p N_{el}}$. Solid and dashed lines plot errors in the curvilinear element DG solution and in the DG solution with the CBC approach, respectively. A dotted line, unfilled squares, and unfilled circles show errors in the linear, quadratic, and cubic-element DG solution with conventional no-normal flow implementation, respectively. Symbols indicate meshes used for the calculation: ● h-mesh, ♦ $h/2$-mesh, ■ $h/4$-mesh, + $h/8$-mesh.

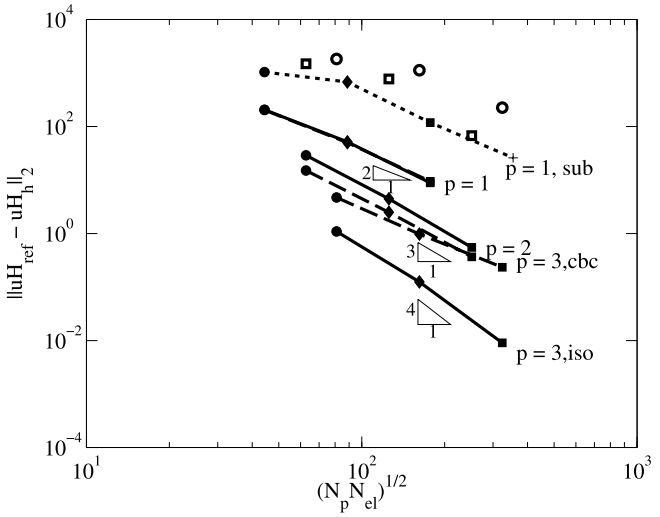

Fig. 18. L_2-errors in $(uH)_h$ as a function of $(1/h) \sim \sqrt{N_p N_{el}}$. Solid and dashed lines plot errors in the curvilinear element DG solution and in the DG solution with the CBC approach, respectively. A dotted line, unfilled squares, unfilled circles show errors in the linear, quadratic, and cubic-element DG solution with conventional no-normal flow implementation, respectively. Symbols indicate meshes used for the calculation: ● h-mesh, ♦ $h/2$-mesh, ■ $h/4$-mesh, + $h/8$-mesh.

Table 1
Overall numerical rate of convergence in the surface water level ζ_h and the x-directed momentum component $(uH)_h$.

p	ζ		uH	
	CBC	Curv. element	CBC	Curv. element
1	2.102	2.159	2.215	2.266
2	2.358	2.766	2.680	2.856
3	2.105	3.890	2.166	3.456

6. Conclusions

In this work, we examine DG solutions of flow in channels governed by SWE with different treatments of the no-normal flow boundary condition at solid curved walls. Numerical investigations provide clear evidence that, as observed in gas dynamics, imposing the no-normal flow boundary on the linear approximation of the solid curved wall can lead to unfavorable

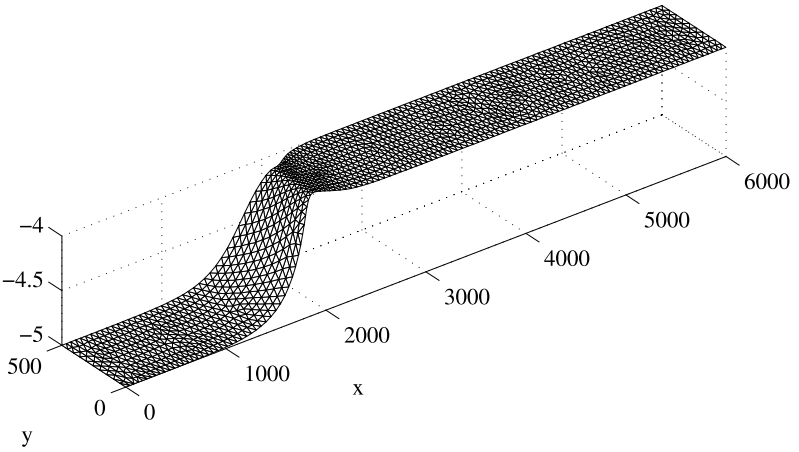

Fig. 19. Profile of the bed $(-b(x))$ used in the well-balanced test.

results. In our test setup, this conventional approach produces artificial boundary layers of one-grid-size thickness and over-prediction of the water surface level upstream of the channel (in practical applications of storm surge, this will increase the computed stage-flow relationships for a channel and artificially over-dampen a hurricane surge propagating up the channel). A proper treatment of no-normal flow boundary conditions on the curved wall is crucial for achieving accurate DG solutions to the SWE. Such a treatment is essential to achieve realistic coarse grid solutions and is especially critical for obtaining a high-order solution from the high-order DG schemes for flow in channels. In this work, we employ both an approach that accounts for errors introduced by the geometric approximation and an approach that accurately represents the geometry. The former, which is done here through the use of the CBC approach, offers a simple solution in an existing computer code built for straighted-edge elements. In our test case, this approach performs well for linear and quadratic elements. However, it offers little to no benefit in terms of accuracy as p is further increased. The reason for this may stem from errors introduced by not matching the computational boundary to the physical boundary in the CBC approach, although evidently less than the conventional no-normal flow approach, may increasingly become the major source of errors as p increases and lead to sub-optimal convergence. Here, an approach that accurately represents the geometry is implemented through the use of curvilinear elements. This approach yields highly desirable numerical accuracy in that the solution converges at the expected rate. The curvilinear element, however, is more computationally expensive than the straighted-edge elements.

Acknowledgements

This work was supported by National Science Foundation grants DMS-1228212, ACI-1339738, DMS-1228243, ACI-1339801, DMS-1217071, DMS-0915118, and DMS-1217218.

Appendix A. Well-balanced test

In this section, we use the test problem below to demonstrate the well-balanced property of the DG scheme with the treatment of the bed term discussed in Section 3.3. Here, we present only the case that involves the curvilinear elements. We consider the lake-at-rest problem defined on the domain described in Section 5.1 with the bathymetric depth

$$b(x,y) = \begin{cases} 5 - \operatorname{sech}\left(\dfrac{x-2000}{200}\right) & \text{for } x \le 2000 \\ 4 & \text{for } x > 2000 \end{cases} \tag{25}$$

The profile of the bed is depicted in Fig. 19. The no-normal flow boundary conditions are prescribed at the domain boundaries and the state at rest as the initial condition

$$\zeta(\boldsymbol{x}, t=0) = \zeta_0, \quad \mathbf{u}H = \mathbf{0} \tag{26}$$

with ζ_0 is set to 1/4 in the numerical calculations. The numerical solution is computed on the $h/2$-mesh shown in Fig. 6 and curvilinear elements are used along the curved-wall boundaries. For high-order elements ($p \ge 2$), the iso-parametric elements ($\widetilde{p} = p$) are used in representing the geometry and for the linear elements ($p = 1$) the super-parametric elements $\widetilde{p} = 2$ are employed. The time integration is performed until $t = 172\,800$ (2 days) is reached. In this study, the bottom friction terms are included in the numerical test (note that the bottom friction terms vanish when $\mathbf{u} = \mathbf{0}$). Table 2 tabulates the L_∞ error in the approximate water surface level and in the x-, y-directed discharge. It can be observed from these results that the scheme processes the well-balanced property.

Table 2
Well-balanced test with curvilinear elements. L_∞-errors in ζ_h and $(\mathbf{u}H)_h$ at $t = 172\,800$.

$(p, \widetilde{p})$	$\|\zeta_h - \zeta_0\|_\infty$	$\|(uH)_h\|_\infty$	$\|(vH)_h\|_\infty$
(1, 2)	1.59e−13	4.95e−11	3.28e−11
(2, 2)	1.16e−13	1.61e−10	1.19e−10
(3, 3)	6.52e−12	2.91e−08	1.89e−08
(4, 4)	5.79e−12	4.78e−08	7.32e−08

References

[1] D.R. Lynch, W.G. Gray, A wave equation model for finite element tidal computations, Comput. Fluids 7 (1979) 207–228.
[2] R.A. Luettich Jr., J.J. Westerink, N.W. Scheffner, ADCIRC: An advanced three-dimensional circulation model for shelves, coasts, and estuaries: theory and methodology of ADCIRC-2DDI and ADCIRC-3DL, Technical report drp-92-6, US Army Corps of Engineers, DC, 1992.
[3] S.W. Bova, G.F. Carey, A symmetric formulation and SUPG scheme for shallow water equations, Adv. Water Resour. 19 (1996) 123–131.
[4] J.C. Galland, N. Goutal, J.M. Hervouet, TELEMAC: a new numerical model for solving shallow water equations, Adv. Water Resour. 14 (1991) 138–148.
[5] B. Cockburn, C.-W. Shu, Runge–Kutta discontinuous Galerkin methods for convection-dominated problems, J. Sci. Comput. 16 (2001) 173–261.
[6] B. Cockburn, Discontinuous Galerkin methods, Z. Angew. Math. Mech. 83 (2003) 731–754.
[7] J.S. Hesthaven, T. Warburton, Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Application, Springer Science+Business Media, LLC, 2008.
[8] V. Aizinger, C. Dawson, A discontinuous Galerkin method for two-dimensional flow and transport in shallow water, Adv. Water Resour. 25 (2002) 67–84.
[9] E.J. Kubatko, J.J. Westerink, C. Dawson, *hp* Discontinuous Galerkin methods for advection dominated problems in shallow water flow, Comput. Methods Appl. Mech. Eng. 196 (2006) 437–451.
[10] E.J. Kubatko, S. Bunya, C. Dawson, J.J. Westerink, C. Mirabito, A performance comparison of continuous and discontinuous finite element shallow water models, J. Sci. Comput. 40 (2009) 315–339.
[11] F.X. Giraldo, T. Warburton, A high-order triangular discontinuous Galerkin oceanic shallow water model, Int. J. Numer. Methods Fluids 56 (7) (2008) 899–925.
[12] S. Bunya, E.J. Kubatko, J.J. Westerink, C. Dawson, A wetting and drying treatment for the Runge–Kutta discontinuous Galerkin solution to the shallow water equations, Comput. Methods Appl. Mech. Eng. 198 (2009) 1548–1562.
[13] C. Dawson, E.J. Kubatko, J.J. Westerink, C. Trahan, C. Mirabito, C. Michoski, N. Panda, Discontinuous Galerkin methods for modeling hurricane storm surge, Adv. Water Resour. 34 (2010) 1165–1176, http://dx.doi.org/10.1016/j.advwatres.2010.11.004.
[14] T. Kärnä, B. de Brye, O. Gourgue, J. Lambrechts, R. Comblen, V. Legat, E. Deleersnijder, A fully implicit wetting–drying method for DG-FEM shallow water models, with an application to the Scheldt estuary, Comput. Methods Appl. Mech. Eng. 200 (2011) 509–524.
[15] C. Michoski, C. Mirabito, C. Dawson, D. Wirasaet, E.J. Kubatko, J.J. Westerink, Dynamic *p*-enrichment schemes for multicomponent reactive flows, Adv. Water Resour. 34 (2011) 1666–1680.
[16] C. Dawson, J.J. Westerink, J.C. Feyen, D. Pothina, Continuous, discontinuous, and coupled discontinuous–continuous Galerkin finite elements methods for the shallow water equations, Int. J. Numer. Methods Fluids 52 (2006) 63–88.
[17] D. Wirasaet, S. Tanaka, E.J. Kubatko, J.J. Westerink, C. Dawson, A performance comparison of nodal discontinuous Galerkin methods on triangles and quadrilaterals, Int. J. Numer. Methods Fluids 64 (2010) 1326–1362.
[18] D. Wirasaet, E.J. Kubatko, C.E. Michoski, S. Tanaka, J.J. Westerink, C. Dawson, Discontinuous Galerkin methods with nodal and hybrid modal/nodal triangular, quadrilateral, and polygonal elements for nonlinear shallow water flow, Comput. Methods Appl. Mech. Eng. 270 (2014) 113–149.
[19] C. Michoski, C. Mirabito, C. Dawson, D. Wirasaet, E.J. Kubatko, J.J. Westerink, Adaptive hierarchic transformations for dynamically *p*-enriched slope-limiting over discontinuous Galerkin systems of generalized equations, J. Comput. Phys. 230 (22) (2011) 8028–8056.
[20] C. Mirabito, C. Dawson, E.J. Kubatko, J.J. Westerink, S. Bunya, Implementation of a discontinuous Galerkin morphological model on two-dimensional unstructured meshes, Comput. Methods Appl. Mech. Eng. 200 (2011) 189–207.
[21] P.A. Tassi, C.A. Vionnet, S. Rhebergen, O. Bokhove, A discontinuous Galerkin finite element model for river bed evolution under shallow flows, Comput. Methods Appl. Mech. Eng. 197 (2008) 2930–2947.
[22] F. Bassi, S. Rebay, High-order accurate discontinuous finite element solution of the 2D Euler equations, J. Comput. Phys. 138 (1997) 251–285.
[23] L. Krivodonova, M. Berger, High-order accurate implementation of solid wall boundary conditions in curved geometries, J. Comput. Phys. 211 (2006) 492–512.
[24] B. Cockburn, C.-W. Shu, The local discontinuous Galerkin method for time-dependent convection–diffusion systems, SIAM J. Numer. Anal. 35 (1998) 2440–2463.
[25] M. Dubiner, Spectral methods on triangle and other domains, J. Sci. Comput. 6 (1991) 345–390.
[26] D.A. Dunavant, High degree efficient symmetrical Gaussian quadrature rules for the triangles, Int. J. Numer. Methods Eng. 21 (1985) 1129–1148.
[27] S. Gottlieb, C.-W. Shu, E. Tadmor, Strong stability-preserving high-order time discretization methods, SIAM Rev. 43 (2001) 89–112.
[28] E.J. Kubatko, J.J. Westerink, C. Dawson, Semi-discrete discontinuous Galerkin methods and stage-exceeding-order, strong-stability-preserving Runge–Kutta time discretizations, J. Comput. Phys. 228 (2007) 832–848.
[29] P. Šolín, Partial Differential Equations and the Finite Element Method, John Wiley & Sons, NJ, 2006.
[30] R.J. LeVeque, Balancing source terms and flux gradients in high-resolution Godunov methods: the quasi-steady wave-propagation algorithm, J. Comput. Phys. 146 (1) (1998) 346–365.
[31] E. Audusse, F. Bouchut, M.-O. Bristeau, R. Klein, B. Perthame, A fast and stable well-balanced scheme with hydrostatic reconstruction for shallow water flows, SIAM J. Sci. Comput. 25 (6) (2004) 2050–2065.
[32] Q. Zhang, C.-W. Shu, Error estimates to smooth solutions of Runge–Kutta discontinuous Galerkin methods for scalar conservative laws, SIAM J. Numer. Anal. 42 (2004) 641–666.