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, U.S.A.*
[b] *Department of Civil and Environmental Engineering and Geodetic Science, The Ohio State University, Columbus, OH, 43210, U.S.A.*
[c] *Institute for Computational Engineering and Sciences, The University of Texas at Austin, Austin, TX, 78712, U.S.A.*

Abstract

In this work, we examine realizations of no-normal flow boundary conditions on curved boundaries for discontinuous Galerkin (DG) solutions to channel problems governed by two-dimensional shallow water equations (SWE). Two approaches, both of which use a straight-sided-element mesh, are considered. The first approach, which is commonly used, consists simply of imposing the no-normal flow condition directly on the boundary of a straight-sided-element mesh. The second approach is a so-called curvature boundary condition which approximates a no-normal flow condition on the physical curved wall. Numerical solutions to sub-critical flow in a converging-diverging channel clearly indicate that, as in gas dynamics, 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 problem, errors introduced through the straightforward application of the no-normal flow condition on the linear approximation of curved walls result in artificial boundary layers in the velocity field and a corresponding over-prediction of the surface elevation in the upstream direction. These significant inaccuracies appear in flow problems without or with a lateral momentum diffusion term and in all DG schemes employed including those using linear, quadratic, and cubic DG polynomials, and both the local Lax-Friedrichs and Roe numerical fluxes.

Keywords: Discontinuous Galerkin, finite elements, shallow water equations, open channel, numerical fluxes, no normal flow boundary condition

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 modelling many environmental flows, such as, tides, hurricane induced flooding, open channel and riverine flow. Simulation of these problems often involve 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 convection processes. Successful continuous Galerkin (CG) finite element solutions to the SWE include, but are not limited to, those devised in [1, 2, 3, 4]. Discontinuous Galerkin (DG) finite element methods (see [5, 6, 7] and references therein for detailed account and reviews of DG methods), which excel in the solution of propagation- and convection-dominated problems, have emerged

---

*Corresponding author
Email address:* `dwirasae@nd.edu` (D. Wirasaet)

as a powerful alternative for solving the SWE [8, 9, 10, 11, 12, 13, 14, 15]. Conceptually similar to finite volume (FV) methods, DG methods inherently posses the property of being conservative on 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 approximation 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.

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 [10, 8] on serial machines is approximately four to five times more expensive than that of the CG solution [2](the latter solves the generalized wave continuity equation, a reformulated form of the SWE). However, the subsequent study in [10] shows 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. In fact, for problems with smooth solutions, DG solutions may gain a significant computational advantage when using polynomial interpolants of degree p greater than unity. Note that successful SWE CG-based solvers are mostly based on linear interpolation. Numerical investigations in [17, 18], which employ test problems with smooth solutions, demonstrate a remarkable advantage of the high-order DG methods ($p > 1$) in terms of cost-per-accuracy performance. For test problems used therein, the cost required for a moderate level of error in DG solutions with quadratic or bi-quadratic interpolants are typically two orders of magnitude lower than that of the linear-element DG methods. The benefit gained from using a scheme with $p + 1$ over a scheme p however diminishes as the interpolation order p increases. Note that coastal flow problems usually contain terms involving spatially varying bathymetry, curved boundaries, bottom friction, surface wind stress, and time-varying boundary conditions. High-order DG solutions may fail to yield high-order accuracy if these terms are not treated properly. Various aspects of obtaining high-order accuracy DG solutions are examined in [19].

In gas dynamics, Bassi and Rebay [20] 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 imposed. Numerical results shown therein (also see [21]) demonstrate that, when the solid curved wall is approximated by its linear approximation—a set of straight segments, the DG methods under p-refinement fail to yield a numerical solution that converges to the true solution. Errors introduced by the geometric approximation appear to have a strong effect on the solution away from the boundary. Bassi and Rebay [20] 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.

In this paper, we investigate the effect of curved-wall boundary treatments in DG solutions for SWE. We are particularly interested in open channel flow problems, which are an integral part of inland and coastal flow and transport computations. Typically, as widely employed in CG calculations, DG calculations replace channel curved walls with a linear approximation (see for example [22, 23]) and apply the no-normal flow condition on each straight segment in a straightforward manner. As shall be detailed below in Section 5, this leads to the presence of one-grid-size artificial boundary layers for $p = 1$ and h/p boundary layers for $p > 1$ in flow through a converging-diverging

2

channel and an over-prediction of the surface elevation on the upstream side. In this work, instead of implementing curvilinear elements, we employ a so-called curvature-boundary-condition approach, proposed originally for Euler equations in [21], 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 formula so that the physical no-normal flow conditions are better approximated on the straight-sided-element mesh (see [21] and also in Section 5.2.2). Furthermore, we examine the effect of different numerical fluxes as well as h- and p-refinement on the accuracy of the solution.

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, 10] is briefly summarized in Section 3. Section 4 contains the detailed account of the two different approaches for treating the no-normal flow condition on a solid wall: the conventional approach and the curvature-boundary-condition approach. In this study, a converging/diverging channel problem is used as a test problem and is described in Section 5.1. Sections 5.2 and 5.3 present results from the study on the flow problem without and with the lateral momentum diffusion, respectively. 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:

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

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

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

$\mathbf{F}^d(\mathbf{q}, \nabla \mathbf{q}) = (\mathbf{f}_1^d, \mathbf{f}_2^d)$ is the flux term accounting for an effect of the lateral momentum diffusion from turbulence and the forcing term $\mathbf{s}$ is given by

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

Here, we use the form of the diffusion term as given in [2], more precisely, $\mathbf{F}^d$ with

$$\mathbf{f}_1^d = \begin{pmatrix} 0 \\ \nu_T \frac{\partial uH}{\partial x} \\ \nu_T \frac{\partial vH}{\partial x} \end{pmatrix} \text{ and } \mathbf{f}_2^d = \begin{pmatrix} 0 \\ \nu_T \frac{\partial uH}{\partial y} \\ \nu_T \frac{\partial vH}{\partial y} \end{pmatrix}, \tag{4}$$

In (2)-(4), ζ denotes the surface elevation measured positive upwards from a specified datum (see Figure 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

3

magnitude of the gravitational acceleration, v_T represents the lateral eddy viscosity, and τ_b denotes the bottom-stress friction factor. In this study, we use the quadratic bottom friction law for τ_b, namely,

$$\tau_b = C_f \frac{\sqrt{u^2 + v^2}}{H} \tag{5}$$
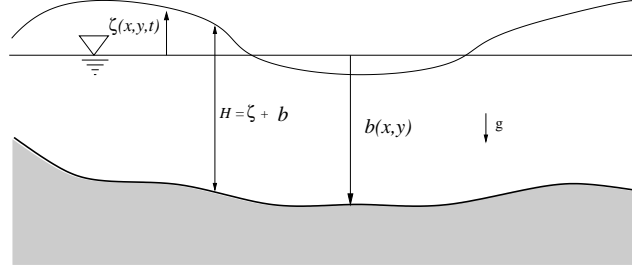
where C_f is a bottom drag coefficient.

Figure 1: Schematic diagram of the free surface and bathymetry

3. Discontinuous Galerkin finite element discretization

We consider a DG scheme for SWEs described in [8, 9, 12, 13, 24]. Such a scheme, summarized in brief below, is based on the use of a DG method with an orthogonal polynomial basis on triangles for the spatial discretization and the strong-stability-preserving Runge-Kutta (SSPRK) scheme for the time integration.

The DG discretization is based on the equations that are a result of reformulating (1) into a system of first order equations, namely,

$$\begin{cases} \dfrac{\partial \mathbf{q}}{\partial t} + \nabla \cdot [\mathbf{F}(\mathbf{q}) + \mathbf{Z}(\mathbf{z})] = \mathbf{s}(\mathbf{q}, \boldsymbol{x}, t) \\ \\ \mathbf{z} = -\nabla \mathbf{U} \end{cases}, \tag{6}$$

where $\mathbf{U} = (U_1 \equiv q_2, U_2 \equiv q_3) = (uH, vH)$ and

$$\mathbf{Z} = \begin{pmatrix} 0 & 0 \\ v_T z_{1,1} & v_T z_{2,1} \\ v_T z_{1,2} & v_T z_{2,2} \end{pmatrix},$$

where $z_{i,j} = \partial U_j / \partial x_i$ denotes the $(i, j)^{\text{th}}$-component of the auxiliary variable $\mathbf{z}$ (note that $x_1 \equiv x$ and $x_2 \equiv y$). To discretize (6) in space using the DG method, the domain Ω is triangulated into a set of non-overlapping triangular elements denoted by $\mathcal{T}_h$. The solution $(\mathbf{q}, \mathbf{z})$ is then replaced by a discontinuous approximate solution $(\mathbf{q}_h, \mathbf{z}_h)$. Such approximate solutions are selected so that, when restricted to the element $K \in \mathcal{T}_h$, each solution component belongs to a finite dimensional space $V(K) = (\mathcal{P}^p(K))$ where $\mathcal{P}^p(K)$ is the space of polynomials of degree at most p. More precisely, letting $\mathbf{\Phi}^K = \{\phi_m^K(\boldsymbol{x})\}_{m=1,...,N_p}$ be a basis of $\mathcal{P}^p(K)$, the solution on the element K is approximated by

4

$(\mathbf{q}_h|_K = (q_{h,i}^K)_{i=1,3}, \mathbf{z}_h|_K = (z_{h,i,j}^K)_{i,j=1,2}$ with

$$q_{h,i}^K = \sum_{m=1}^{N_p} \widetilde{q}_{m,i}^K(t)\phi_m^K(\boldsymbol{x}), \quad z_{h,i,j}^K = \sum_{m=1}^{N_p} \widetilde{z}_{m,i,j}^K(t)\phi_m^K(\boldsymbol{x}), \tag{7}$$

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

$$\int_K \phi_m^K \frac{\partial q_{h,l}^K}{\partial t} d\boldsymbol{x} - \int_K \nabla\phi_m^K \cdot [\mathbf{F}_l(\mathbf{q}_h) + \mathbf{Z}_l(\mathbf{z}_h)] \, d\boldsymbol{x} +$$
$$\int_{\partial K} \phi_m^K(\widehat{\mathbf{F}}_l + \widehat{\mathbf{Z}}_l) \cdot \mathbf{n} ds = \int_K \phi_m^K \mathbf{s}_l(\mathbf{q}_h) d\boldsymbol{x}, \quad l = 1, 2, 3 \tag{8}$$

$$\int_K \phi_m^K z_{h,i,j}^K d\boldsymbol{x} - \int_K U_{h,j} \frac{\partial \phi_m^K}{\partial x_i} d\boldsymbol{x} + \int_{\partial K} \widehat{U}_{h,j} n_i \phi_m^k ds = 0, \quad i, j = 1, 2 \tag{9}$$

for all $\phi_m^K \in \boldsymbol{\Phi}^K$ and $K \in \mathcal{T}_h$. In the above formula, $\mathbf{n} = (n_x, n_y)$ represents the outward-pointing unit normal vector to the boundary ∂K (note that in (9), $n_1 \equiv n_x$ and $n_2 \equiv n_y$), $\mathbf{F}_l = ((\mathbf{f}_1)_l, (\mathbf{f}_2)_l)$ denotes the lth-component of the flux term e.g. $\mathbf{F}_1 = (uH, vH))$ and $\mathbf{s}_l$ represents the lth-component of the vector of forcing terms. The so-called numerical fluxes $\widehat{\mathbf{F}}$, $\widehat{\mathbf{Z}}$, and $\widehat{\mathbf{U}}$ are employed to resolve $\mathbf{F}(\mathbf{q}_h)$, $\mathbf{Z}(\mathbf{z}_h)$, and $\mathbf{U}_h$ on the element boundary due to the approximation being discontinuous across the element interface. They are crucial for the stability and convergence of the DG method [5, 6]. The numerical fluxes depend on the traces of $(\mathbf{q}_h, \mathbf{z}_h)$ from both sides of the element interface, namely, $(\mathbf{q}^{\text{in}}, \mathbf{q}^{\text{ex}})$ and $(\mathbf{z}^{\text{in}}, \mathbf{z}^{\text{ex}})$. Here, we use $\mathbf{w}^{\text{in}}$, $\mathbf{w} \equiv \mathbf{q}_h, \mathbf{z}_h$, 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 $\mathbf{w}^{\text{ex}}$ the value when approaching from the exterior (i.e. from a neighboring element sharing the edge ∂K) of the element K. Note that the numerical fluxes are the terms in the DG formula that couple the solution between neighboring elements.

The numerical flux $\widehat{\mathbf{F}} \cdot \mathbf{n}$ can be any locally Lipschitz-continuous, monotone flux that is consistent with the nonlinear flux $\mathbf{F} \cdot \mathbf{n}$ [5]. Note that such a numerical flux is defined through the given data $(\mathbf{q}^{\text{in}}, \mathbf{q}^{\text{ex}}; \mathbf{n})$. In this study, we consider two-widely used fluxes: the Local Lax-Friedrichs flux (LLF) and the Roe numerical flux. We provide a formula for these fluxes in Appendix A. The LLF flux is simple and computationally inexpensive. The Roe numerical flux is an approximate Riemann solver, which solves exactly the Roe linearized problem of the normal Riemann problem [25, 26]. It is an upwind flux and, for a problem governed by conservation laws, is expected to be superior to the LLF flux in terms of convergence [27]. As demonstrated later in section 5.2, the Roe numerical flux indeed yields more accurate results for a channel flow problem without the diffusion term.

In general, the diffusive fluxes $\widehat{\mathbf{Z}}$ and $\widehat{\mathbf{U}}$ may depend on both $(\mathbf{q}^{\text{in}}, \mathbf{q}^{\text{ex}})$ and $(\mathbf{z}^{\text{in}}, \mathbf{z}^{\text{ex}})$ data. However, solving a system of differential algebraic equations (8) and (9) can be made computationally simpler by considering local DG fluxes [5, 6]. These fluxes define the flux $\widehat{\mathbf{U}}$ to be dependent only on the data $(\mathbf{q}^{\text{in}}, \mathbf{q}^{\text{ex}})$. With this choice of flux, (9) can be solved element-by-element for $\mathbf{z}_h$ in terms of $\mathbf{q}_h$. As a consequence, the solution procedure for (8) and (9) consists of solving (9) in an elementwise fashion for the auxiliary variable $\mathbf{z}_h$ and subsequently integrate in time the

system of ODEs (8) for the solution $\mathbf{q}_h$. The local DG fluxes used in this study are simple central fluxes [28], namely,

$$\widehat{\mathbf{U}} = \frac{\mathbf{U}^{\text{in}} + \mathbf{U}^{\text{ex}}}{2}, \quad \widehat{\mathbf{Z}} = \frac{\mathbf{Z}(\mathbf{z}^{\text{in}}) + \mathbf{Z}(\mathbf{z}^{\text{ex}})}{2} \tag{10}$$

(as a reminder note that $\mathbf{U} \equiv (q_2, q_3)$).

In realizing (8)-(9), we employ a Dubiner basis [29] for $\mathbf{\Phi}^K$. Such a basis is orthogonal and has a dimension of $N_p = (p + 1)(p + 2)/2$ for a given order p. Consequently, an $N_p \times N_p$ matrix associated with the first term of these equations, also known as an element mass matrix, is diagonal. In (8), 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 trivially inverted in an elementwise fashion. The area integrals are evaluated using a $2p$-order accurate quadrature rule devised for integration over a triangle [30]. The edge integrals are calculated using a $p + 1$-point one-dimensional Gauss quadrature. The resulting system of ordinary differential equations is integrated in time using an explicit strong-stability-preserving Runge-Kutta (SSPRK) scheme [31, 32] of order $p + 1$. The time-step size used in the calculation is selected based on a CFL-type condition

$$\Delta t = \frac{2}{3} C \min_{x \in K, \forall K \in \mathcal{T}_h} \left(\frac{h_K}{(|\mathbf{u}| + \sqrt{gH})(p + 1)} \right) \tag{11}$$

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.

4. Implementation of no-normal flow boundary conditions

In this study, we consider open channel flow problems with solid curved walls. Here, the no-normal flow boundary condition is prescribed on the solid walls. This condition indicates there is no flow across the wall. In other words, the normal component of velocity vanishes at the wall, namely,

$$\boldsymbol{u} \cdot \boldsymbol{N} = 0, \tag{12}$$

where $\boldsymbol{N}$ denotes the unit outward vector normal to the physical boundary. Note that grid generators typically generate meshes consisting of only straighted-sided elements. For these meshes, the boundary of the computational domain is a piecewise linear approximation of the physical domain (the approximate boundary is often 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 the sense that the order of accuracy is not deteriorated by the piecewise linear representation of the boundary. In gas dynamics, Bassi and Rebay [20] demonstrate that an accurate representation of the curved boundary is crucial for achieving optimal accuracy in DG solutions. Isoparametric and superparametric curvilinear elements are used in their investigation. Here, instead of implementing the curvilinear elements, we consider instead a curvature-boundary-condition (CBC) approach, originally devised by Krivodonova and Berger [21] for Euler equations. The CBC approach uses straight-sided elements and takes into account error introduced by the piecewise linear representation of the boundary. This approach offers a simple solution for a DG code that is built for straight-sided element meshes.

Below, we describe the conventional approach and the CBC approach for implementing the no-normal flow conditions on straight-sided element meshes.

4.1. Conventional approach

In the conventional approach, one prescribes the condition

$$\boldsymbol{u} \cdot \mathbf{n} = 0 \tag{13}$$

on an element edge approximating the no-normal flow curve boundary where $\mathbf{n}$ is the unit vector normal to the element edge (see Figure 2 for illustration). To implement this condition in the DG scheme (8) and (9), we use the values of
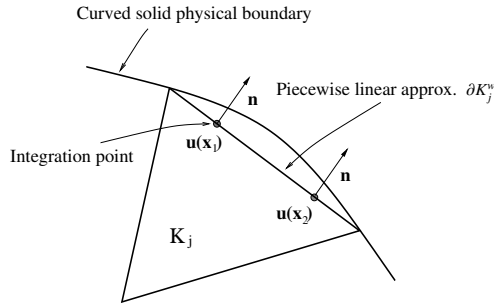
Figure 2: Straight-sided element with an edge approximating a curve boundary

the numerical fluxes that are a result of setting the fictitious exterior states at integration points as follows

$$(\boldsymbol{u}H)^{\text{ex}} \cdot \mathbf{n} = -(\boldsymbol{u}H)^{\text{in}} \cdot \mathbf{n}, \quad (\boldsymbol{u}H)^{\text{ex}} \cdot \boldsymbol{\tau} = (\boldsymbol{u}H)^{\text{in}} \cdot \boldsymbol{\tau}, \quad H^{\text{ex}} = H^{\text{in}}, \quad \mathbf{z}^{\text{ex}} = \mathbf{z}^{\text{in}} \tag{14}$$

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

$$u^{\text{ex}} = \left(n_y^2 - n_x^2\right)u^{\text{in}} - 2n_x n_y v^{\text{in}}, \quad v^{\text{ex}} = \left(n_x^2 - n_y^2\right)v^{\text{in}} - 2n_x n_y u^{\text{in}} \tag{15}$$

It can be verified that the exterior data leads to $\widehat{\mathbf{F}} \cdot \mathbf{n}$ with a vanishing component for the continuity equation and to $\widehat{\mathbf{U}} \cdot \mathbf{n}$ of zero value (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 and the numerical diffusion fluxes $\widehat{\mathbf{U}}$ and $\widehat{\mathbf{Z}}$ are defined through (10)). This indicates that the condition (13) is weakly enforced.

4.2. Curvature-boundary-condition (CBC) approach

The CBC approach, devised by [21] 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

$$\boldsymbol{u} \cdot \boldsymbol{N} = 0, \quad \boldsymbol{x} \in \partial K^w \tag{16}$$

7

where $N = (N_x, N_y)$ is the unit normal to the physical boundary (see Figure 3 for illustration).
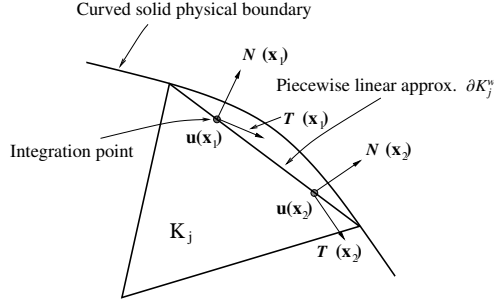

Figure 3: Curvature-boundary-condition approach for two integration points

By following [21], the CBC approach for realizing (16) in the DG scheme is carried out by setting the fictitious exterior state at an integration point to

$$(\boldsymbol{u}H)^{\text{ex}} \cdot \boldsymbol{N} = -(\boldsymbol{u}H)^{\text{in}} \cdot \boldsymbol{N}, \quad (\boldsymbol{u}H)^{\text{ex}} \cdot \boldsymbol{T} = (\boldsymbol{u}H)^{\text{in}} \cdot \boldsymbol{T}, \quad H^{\text{ex}} = H^{\text{in}}, \quad \mathbf{z}^{\text{ex}} = \mathbf{z}^{\text{in}} \tag{17}$$

where $\boldsymbol{T}$ denotes the unit tangential vector ($\boldsymbol{T} \cdot \boldsymbol{N} = 0$ 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 $\boldsymbol{T}$. Note that (17) amounts to specifying the velocity components of the exterior state as

$$u^{\text{ex}} = \left(N_y^2 - N_x^2\right) u^{\text{in}} - 2 N_x N_y v^{\text{in}}, \quad v^{\text{ex}} = \left(N_x^2 - N_y^2\right) v^{\text{in}} - 2 N_x N_y u^{\text{in}} \tag{18}$$

The interior state, fictitious exterior state given above, and the unit vector normal to the approximate boundary $\mathbf{n}$ are used in evaluating the normal numerical convective flux $\widehat{\mathbf{F}} \cdot \mathbf{n}$ and the numerical diffusion fluxes $\widehat{\mathbf{U}}$ and $\widehat{\mathbf{Z}}$. It can be verified that, for non-matching $\mathbf{N}$ and $\mathbf{n}$, the continuity-equation component of $\widehat{\mathbf{F}} \cdot \mathbf{n}$ does not vanish (nor does $\widehat{\mathbf{U}} \cdot \mathbf{n}$); this implies that the CBC approach leads to non-conservation of mass in the computational domain. In addition, the resulting numerical flux is not orthogonal to the normal vector $\mathbf{N}$. Krivodonova and Berger [21] propose also an approach for determining the exterior state yielding the numerical flux that is exactly orthogonal to the vector $\mathbf{N}$. In their numerical calculations which employ the Roe numerical flux, such an approach produces almost identical results to the approach described above.

Note that a procedure for approximating the vector $\mathbf{N}$ from a given straight-sided mesh is described in [21]. In this study, since the analytical expressions of the walls are available (see subsection 5.1), we compute the unit normal vector $\mathbf{N}$ directly from the analytical expression. Although not reported here, from our numerical experiments with a channel problem with more complicate-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.

5. Application of DG to flow in channels

5.1. Problem description

The converging/diverging channel test problem is illustrated in Figure 4. 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 point located at 2000 m from the western end. The profile of the northern and southern walls are given analytically by

$$y_N(x) = Y_{NW} - z_{\max}\operatorname{sech}(4(x - X_c)/L_m) \quad \text{and}$$

$$y_S(x) = Y_{SW} + z_{\max}\operatorname{sech}(4(x - X_c)/L_m),$$

(19)

respectively, where $L_m = 500$, $z_{max} = 100$, $X_c = 2000$, $Y_{NW} = 500$ and $Y_{SW} = 0$. In numerical simulations, we use the
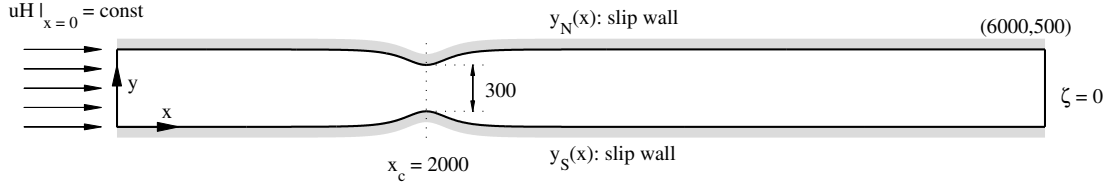

Figure 4: Configuration of converging-diverging channel problem.

quiescent state as the initial condition, more precisely,

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

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 of the problem with $b(x, y) = 10$ (m), $q_w = 5$ (m²/s), and $C_f = 0.0025$.

Numerical solutions are computed on a sequence of refined meshes. Figure 5 shows the coarsest mesh (h-mesh) and the two finer meshes ($h/2$- and $h/4$-meshes). Note that, in these meshes, the northern and southern walls are a piecewise linear approximation of the exact wall profile (19); 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 (11). We set the constant C in (11) 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. The simulations are carried out until $t_f = 2$ days is reached. Here, we consider the flow problem with both zero and non-zero values of eddy viscosity ν_T.
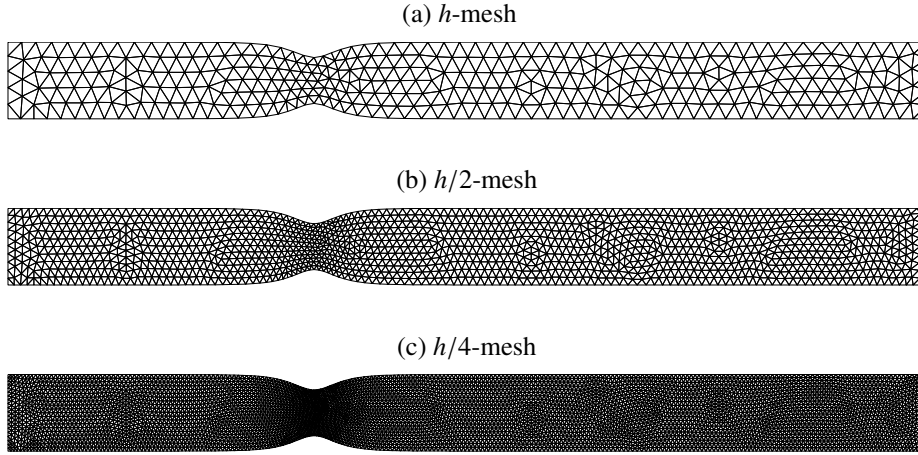
9

(a) *h*-mesh

(b) *h*/2-mesh

(c) *h*/4-mesh

Figure 5: Computational meshes used in numerical tests: (a) *h*-mesh, N_{el} = 656, (b) *h*/2-mesh, N_{el} = 2624, and (c) *h*/3-mesh, N_{el} = 10496

5.2. Numerical results for problem with zero eddy viscosity ($v_T = 0$)

In this section, we present results of the study on the flow problem with zero eddy viscosity ($v_T = 0$). The following subsection discusses the DG solution with the conventional implementation of the no-normal flow boundary conditions. The numerical results of the CBC approach are subsequently given in subsection 5.2.2.

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 Figure 6 and Figure 7, respectively.

(a) *h*-mesh
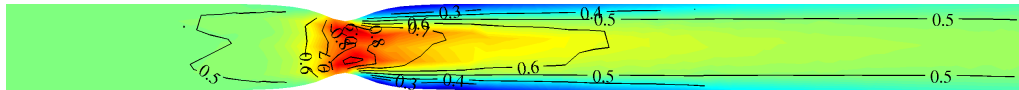
(b) *h*/4-mesh
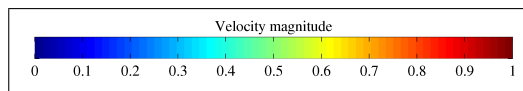
Figure 6: Velocity magnitude $|\boldsymbol{u}|$ at $t = 2$ days from using the LLF DG scheme with $p = 1$ and the conventional implementation of the no-normal flow boundary condition. Solutions under *h*-refinement: (a) *h*-mesh; (b) *h*/4-mesh

The linear-element DG solutions at the same time level using the Roe numerical flux are shown in Figure 8 and 9. Note that, regardless of the type of numerical flux used, boundary layers can be clearly observed in the velocity plots

10
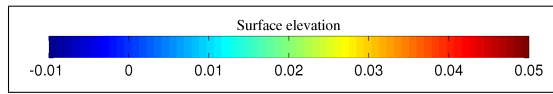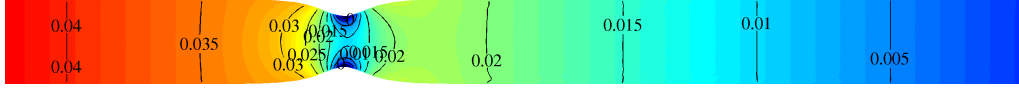
(a) *h*-mesh

(b) *h*/4-mesh

Figure 7: 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.

(a) *h*-mesh

(b) *h*/4-mesh
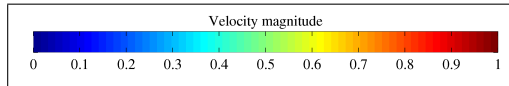
Figure 8: Velocity magnitude $|\boldsymbol{u}|$ at $t = 2$ days obtained from using the Roe-flux DG scheme with $p = 1$ and the conventional implementation of no-normal flow boundary conditions. Solutions under *h*-refinement: (a) *h*-mesh; (b) *h*/4-mesh.

(see Figure 6 and 8 and see also the results reported in [23] and [22]). Such boundary layers emerge from the channel throat and persist far downstream. In Figure 10, 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. Notice that the thickness of the boundary layers decreases as the grid size decreases and it appears to be of approximately one-grid size. 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 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 Figure 11 the surface elevation along the channel's horizontal centerline $\zeta(x, y = 250)$. From the surface elevation plots (see Figure 7, 9, and 11), 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
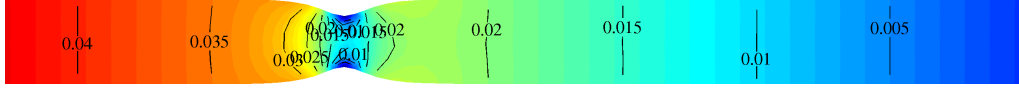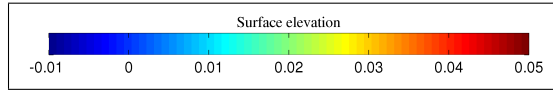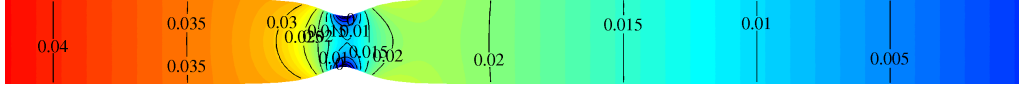
(a) *h*-mesh

(b) *h*/4-mesh

Surface elevation

Figure 9: Surface elevation ζ at t = 2 days obtained from using the Roe-flux DG scheme with p = 1 and the conventional implementation of no-normal flow boundary conditions. Solutions under h-refinement: (a) h-mesh; (b) h/4-mesh.

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

Note that, when comparing the LLF DG solution and Roe-flux DG solution, the boundary layers appearing in the Roe-flux solution are noticeably weaker than those in the solutions with the LLF flux (the LLF DG solutions have a larger dip in the velocity magnitude near the channel wall; see Figure 10). The surface elevation computed from the Roe numerical flux is lower on the upstream side of the channel throat. The lower surface elevation is inter-connected to the weaker boundary layers in the Roe flux solutions. Although not reported here, we note that numerical experiments using other upwind fluxes, more precisely, the exact-Riemann-solver flux and the Hartan-Lax-van Leer-Contact flux yield results that are almost indistinguishable from the Roe numerical flux. These numerical results demonstrate an advantage of the upwind-type numerical flux over the LLF flux for channel flow applications.

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). Figures 12 and 13 show 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 Figures 12 and 6). Qualitatively, the solution appears to be increasingly poor as p increases. Note that the LLF DG solution becomes unsteady for p = 3. For the Roe-flux solution, the velocity magnitude of the high-order scheme agrees qualitatively with the linear-element scheme. Note that all Roe-flux DG solutions reach steady state. In addition, the thickness of the boundary layers decreases as p increases.

(a) LLF DG

$|\mathbf{u}(x = 3000, y)|$

$|\mathbf{u}(x = 4000, y)|$

(b) Roe-flux DG

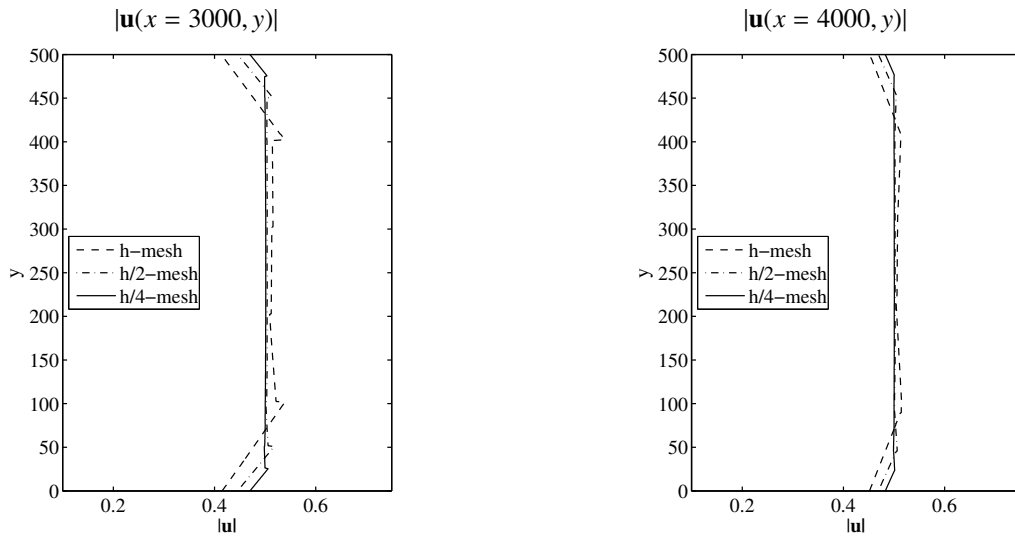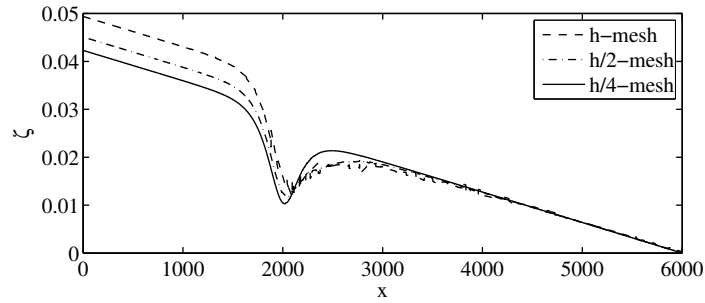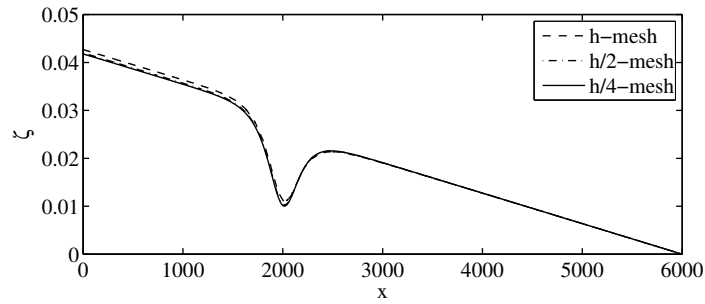$|\mathbf{u}(x = 3000, y)|$

$|\mathbf{u}(x = 4000, y)|$
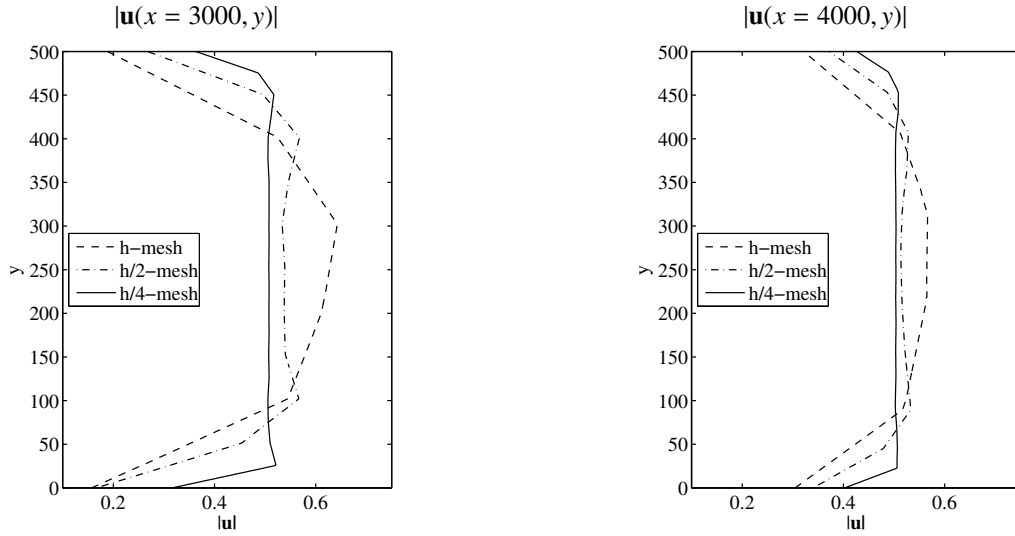
Figure 10: Solution at $t = 2$ days from using the conventional implementation of the no-normal flow boundary condition. Velocity magnitude at the section $x = 3000$ (left) and $x = 4000$ (right) computed using DG schemes with $p = 1$ on meshes of different resolutions. Top: LLF DG solution; (bottom) Roe-flux DG solution.

5.2.2. Results from the CBC implementation of the no-normal flow boundary condition

Below, we report on the numerical results using the CBC approach for implementing the no-normal flow boundary condition. Figure 14 shows snap shots of the velocity magnitude at $t = 2$ days using this approach in the LLF DG method with $p = 1$ on the h- and $h/4$-mesh. We show the velocity magnitude computed using the linear-element Roe-flux DG method and the CBC approach in Figure 15. The velocity profiles plotted appear to visually contain no boundary layer. The plots in Figure 16, which show the velocity magnitude at the cross section $x = 3000$ and
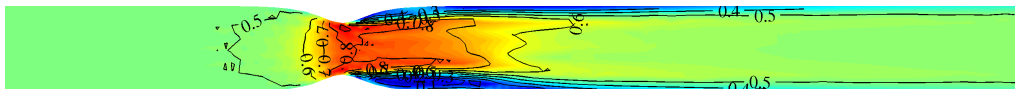
(a) LLF DG

(b) Roe-flux DG

Figure 11: 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) LLF DG solution; (b) Roe-flux DG solution.
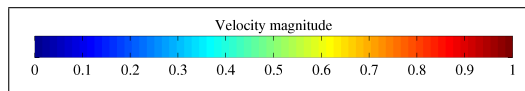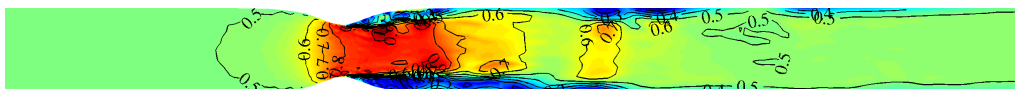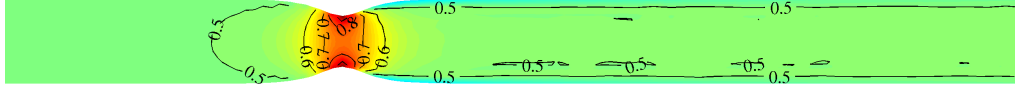
(a) $p = 2$

(b) $p = 3$

Figure 12: 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) $p = 3$.

$x = 4000$, exhibits no boundary layer in the the DG solution (note the LLF solution appears to be more diffusive in comparison to the Roe-flux solution).

The surface elevation at $t = 2$ days from the CBC approach and the linear-element DG are shown in Figure 17 and
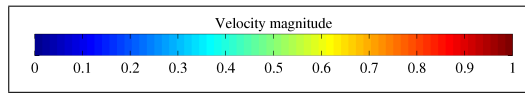
14

(a) $p = 2$

(b) $p = 3$

Figure 13: Velocity magnitude $|\boldsymbol{u}|$ at $t = 2$ from using the Roe-flux DG scheme and the conventional implementation of the no-normal flow boundary condition. Solutions under p-refinement on h-mesh: (a) $p = 2$; (b) $p = 3$.
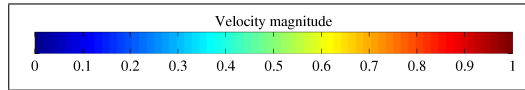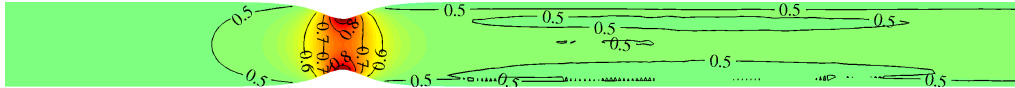
(a) h-mesh

(b) $h/4$-mesh

Figure 14: 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.

18. Note that Figure 17 illustrates the surface elevation of the Roe-flux DG scheme and Figure 18 shows 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 Figure 7, 9 and Figure 11), 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 used in the high-order DG schemes ($p > 1$). Figure 19 and Figure 20 show the velocity magnitude obtained from the high-order LLF DG scheme and Roe-flux DG scheme, respectively. 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 LLF high-order DG scheme with the CBC approach reaches steady state for all orders of p considered.
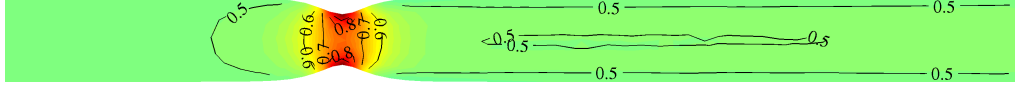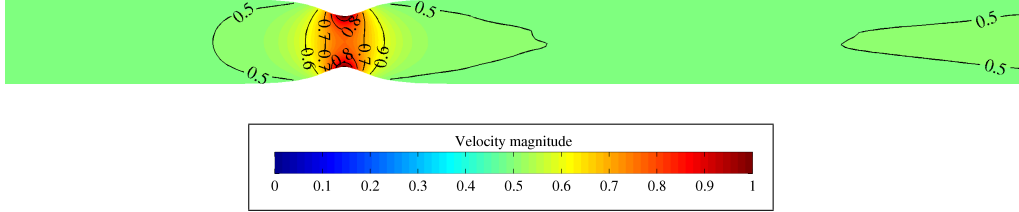
15

(a) *h*-mesh

(b) *h*/4-mesh

Velocity magnitude

| 0 | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 | 1 |

Figure 15: Velocity magnitude |$\boldsymbol{u}$| at $t = 2$ days from using the Roe-flux 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.

.

Since this problem has no analytical solution, we use a Roe-flux DG solution obtained with the CBC no-normal flow approach 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 L_2- and L_∞-norm are reported in Table 1 and Table 2, respectively. We also report in these tables numerical rates of convergence. Note that, from estimates in [27], Roe-flux DG schemes and LLF DG scheme for scalar conservation laws have, in the L_2-norm, the optimal rates of convergence of $p + 1$ and $p + 1/2$, respectively. It can be observed from the data reported that the DG schemes with linear-elements ($p = 1$) yield the approximate solution that converge approximately at $O(h^2)$. For the high-order schemes ($p > 1$), data clearly indicates that the schemes produce the convergent approximate solutions, however, at sub-optimal convergence rates. The convergence rates of the DG schemes with $p = 2$ are approximately of order $O(h^{2.5})$ (which is approximately half an order less than the theoretical optimal order for the Roe-flux DG scheme). The schemes with $p = 3$ appear to produce solutions that converge at a rate of only 2.3 to 2.4, significant lower than the expected order. We 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 no-normal flow approach is still based on the use of an approximate boundary (which is a linearly chopped/patched version of 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 for *p*-refinement.

5.3. Numerical results for problem with non-zero eddy viscosity

We conduct a study on the effect of the treatment of the no-normal flow boundary conditions of the DG solution to the flow problem with the lateral momentum diffusion terms included. For brevity, unless otherwise indicated, we present results from the DG scheme with the numerical convective flux $\widehat{\mathbf{F}} \cdot \mathbf{n}$ of the LLF type. Figures 21 and 22 show, respectively, the velocity magnitude |$\mathbf{u}$| and the surface elevation ζ at $t = 2$ day of the flow problem with $\nu_T = 5$ and 10 obtained on the *h*/4-mesh (a fine-resolution mesh) and the DG scheme with $p = 1$ and the conventional implementation of the no-normal flow boundary condition. The boundary layers downstream from the channel throat
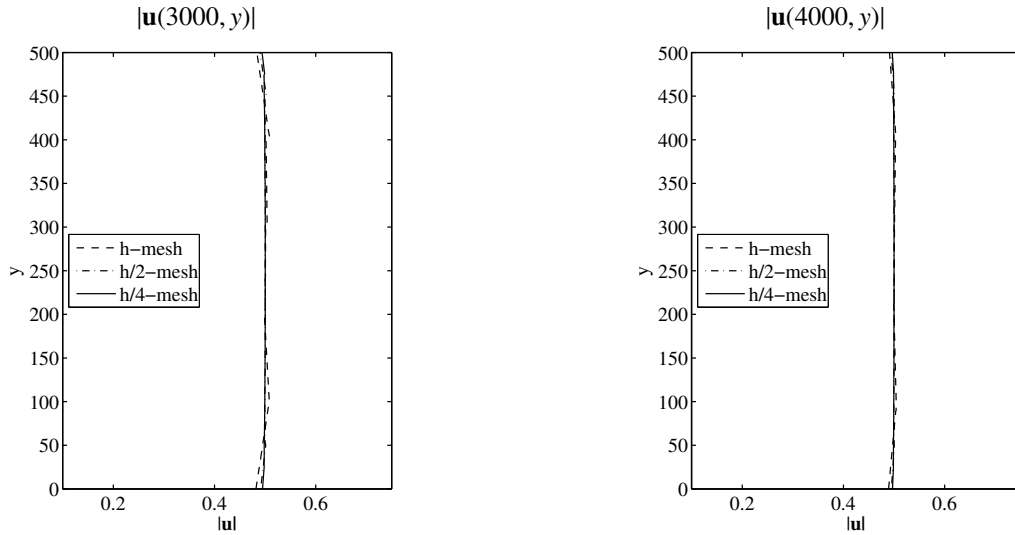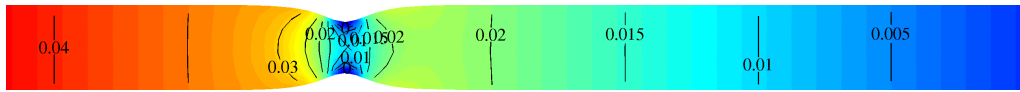
(a) LLF DG
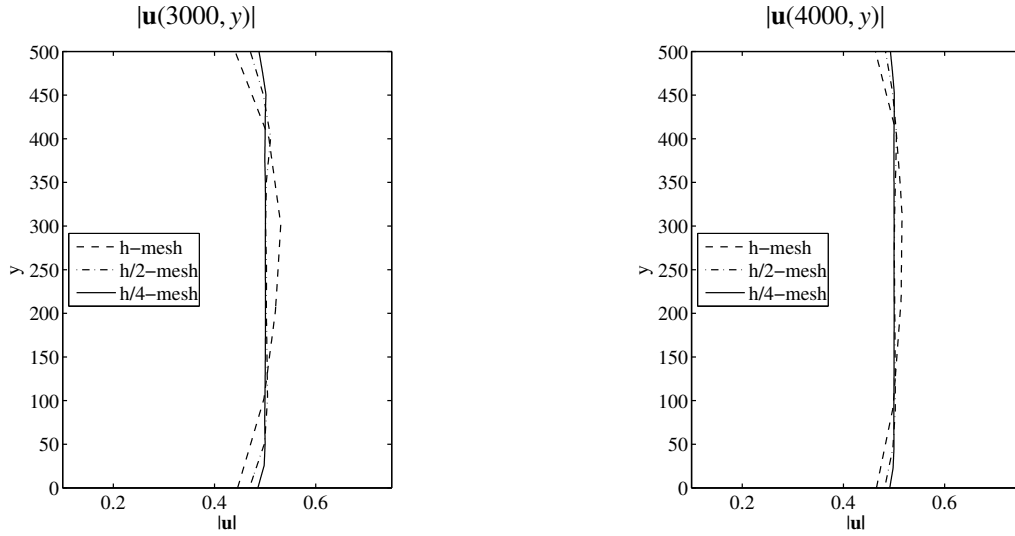
(b) Roe-flux DG

Figure 16: 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. Top: LLF DG solution; bottom: Roe-flux DG solution.

are clearly visible in the velocity plot. These boundary layers are smeared in comparison with those existing in the case with $\nu_T = 0$ (see Figure 22(b)). It can be observed from Figure 22 that, when comparing the surface elevations, the level of surface elevation on the upstream side becomes higher as the value of eddy viscosity increases (in this cases, a steady state solution is a balance of the pressure gradient, convection, the bottom friction, and the diffusion term). Figure 23 shows the velocity magnitudes obtained from the identical DG scheme, except that the CBC approach is used to treat the no-normal flow boundary conditions. Note that the DG solutions are free of an artificial boundary layer when the effect of curved walls is taken into account in the calculations. As one might expect, this has an implication on the surface elevation on the upstream side. Figure 24 plots the surface elevations along the channel
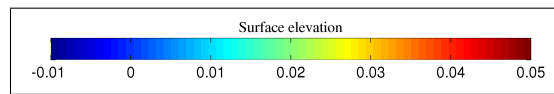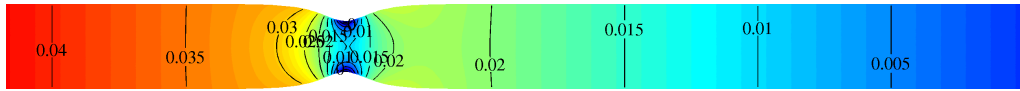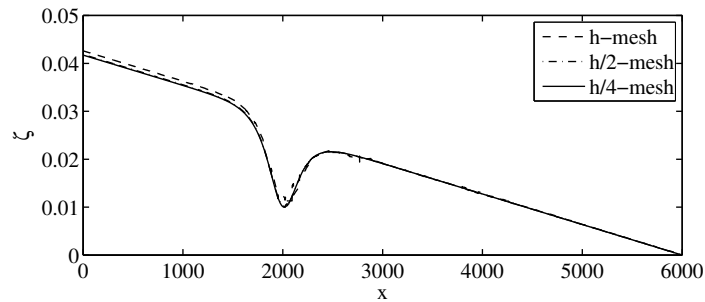
(a) *h*-mesh

(b) *h*/4-mesh

Figure 17: Surface elevation ζ at *t* = 2 days obtained from using the Roe-flux 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.
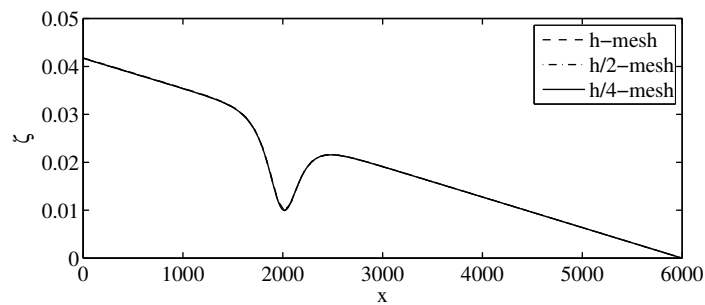
(a) LLF DG

(b) Roe-flux DG

Figure 18: Solution at *t* = 2 days from using the CBC implementation of the no-normal flow boundary condition. Surface elevation along the section *y* = 250, ζ(*x*, *y* = 250) computed using DG schemes with *p* = 1 on meshes of different resolutions. (a) LLF DG solution; (b) Roe-flux DG solution.

centerline ζ(*x*, *y* = 250) from employing the CBC approach and the conventional no-normal flow implementation. The surface elevation on the upstream side obtained from the CBC approach are evidently less sensitive to the values of eddy viscosity in comparison to that obtained from the conventional no-normal flow implementation. This is attributed
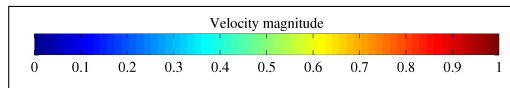
(a) $p = 2$

(b) $p = 3$

Velocity magnitude

| 0 | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 | 1 |

Figure 19: 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) $p = 3$.

(a) $p = 2$

(b) $p = 3$

Velocity magnitude

| 0 | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 | 1 |
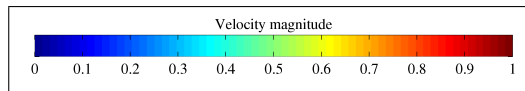
Figure 20: Velocity magnitude $|\boldsymbol{u}|$ at $t = 2$ days obtained from using the Roe's numerical flux and the CBC approach for no-normal flow boundary conditions. Results are solved on h-mesh with (a) $p = 2$, and (b) $p = 3$.

(a) $v_T = 5$

(b) $v_T = 10$

Velocity magnitude

| 0 | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 | 1 |
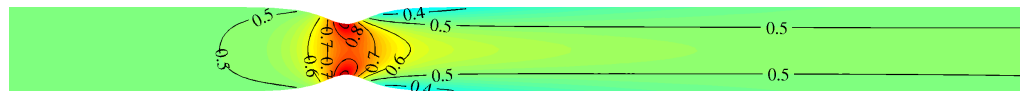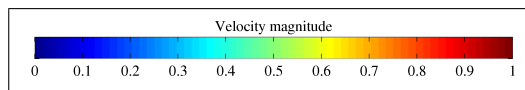
Figure 21: Non-zero eddy viscosity flow. Velocity magnitude $\|\boldsymbol{u}\|$ at $t = 2$ days obtained from DG scheme with $p = 1$, h-mesh, the convective flux of the LLF type, and the conventional implementation of the no-normal flow boundary condition. Solution of flow with: (a) $v_T = 5$ and (b) $v_T = 10$.

19

Table 1: L_2-errors and numerical rate of convergence in DG solutions at $t = 2$ days using the CBC no-normal flow approach. Data for: (a) surface elevation; (b) x-directed momentum component uH.

(a) $\|\zeta_h - \zeta_{\text{ref}}\|_2$ and numerical rates of convergence

Flux	Mesh	$p = 1$		$p = 2$		$p = 3$	
		L_2-error	rate	L_2-error	rate	L_2-error	rate
Roe	h	1.467e-01	-	5.144e-02	-	4.198e-02	-
	$h/2$	3.156e-02	2.21	9.797e-03	2.39	8.827e-03	2.25
	$h/4$	7.846e-03	2.01	1.824e-03	2.42	1.749e-03	2.33
LLF	h	1.098e+00	-	7.499e-02	-	4.271e-02	-
	$h/2$	2.564e-01	2.10	1.395e-02	2.43	8.881e-03	2.27
	$h/4$	5.967e-02	2.10	2.383e-03	2.55	1.754e-03	2.34

(b) $\|(uH)_h - (uH)_{\text{ref}}\|_2$ and numerical rates of convergence

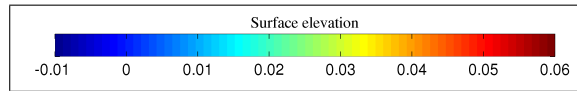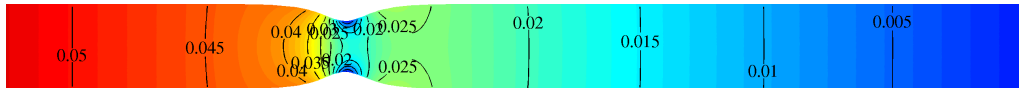| Flux	Mesh	$p = 1$		$p = 2$		$p = 3$	
		L_2-error	rate	L_2-error	rate	L_2-error	rate
Roe	h	4.767e+01	-	8.063e+00	-	4.220e+00	-
	$h/2$	9.427e+00	2.34	1.329e+00	2.60	8.740e-01	2.27
	$h/4$	1.821e+00	2.37	2.067e-01	2.67	1.727e-01	2.34
LLF	h	2.037e+02	-	1.503e+01	-	4.684e+00	-
	$h/2$	5.011e+01	2.02	2.470e+00	2.61	9.234e-01	2.34
	$h/4$	9.464e+00	2.40	3.326e-01	2.89	1.791e-01	2.37

(a) $v_T = 5$
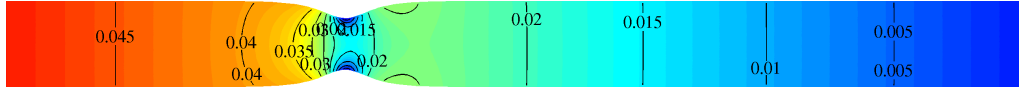
(b) $v_T = 10$

Figure 22: Non-zero eddy viscosity flow. Surface elevation ζ at $t = 2$ days obtained from DG scheme with $p = 1$, $h/4$-mesh, the convective flux of the LLF type, and the conventional implementation of the no-normal flow boundary condition. Solution of flow with: (a) $v_T = 5$ and (b) $v_T = 10$.

mainly to the fact that the solutions from the CBC approach are not polluted by the artificial boundary layers (which, in effect, narrows the channel width and consequently leads to higher velocities in the channel-centerline area). In other words, the presence of the artificial boundary layers in the DG solution with the conventional no-normal flow implementation results in the over-prediction of the surface elevation on the upstream. We observe that, for the conventional no-normal flow implementation, the level of surface elevation on the upstream side is more sensitive

20

Table 2: L_∞-errors and numerical rate of convergence in DG solutions at $t = 2$ days using the CBC no-normal flow approach. Data for: (a) surface elevation; (b) x-directed momentum component uH.

(a) $\|\zeta_h - \zeta_{\text{ref}}\|_\infty$ and numerical rates of convergence

Flux	Mesh	$p = 1$		$p = 2$		$p = 3$	
		L_∞-error	rate	L_∞-error	rate	L_∞-error	rate
Roe	h	4.317e-03	-	1.320e-03		9.675e-04	-
	$h/2$	1.269e-03	1.77	2.791e-04	2.24	2.346e-04	2.04
	$h/4$	3.056e-04	2.05	4.997e-05	2.48	4.745e-05	2.30
LLF	h	7.748e-03	-	1.242e-03	-	8.423e-04	-
	$h/2$	2.963e-03	1.38	2.993e-04	2.05	2.186e-04	1.95
	$h/4$	1.028e-03	1.53	8.264e-05	1.85	4.611e-05	2.24

(b) $\|(uH)_h - (uH)_{\text{ref}}\|_\infty$ and numerical rates of convergence

Flux	Mesh	$p = 1$		$p = 2$		$p = 3$	
		L_∞-error	rate	L_∞-error	rate	L_∞-error	rate
Roe	h	6.803e-01	-	1.739e-01	-	7.572e-02	-
	$h/2$	2.172e-01	1.65	5.391e-02	1.69	2.161e-02	1.81
	$h/4$	5.149e-02	2.08	9.985e-03	2.43	4.656e-03	2.21
LLF	h	1.226e+00	-	1.702e-01	-	8.018e-02	-
	$h/2$	5.118e-01	1.26	4.425e-02	1.94	2.401e-02	1.74
	$h/4$	1.825e-01	1.49	9.499e-03	2.21	4.986e-03	2.27

(a) $\nu_T = 5$

(b) $\nu_T = 10$

Figure 23: Non-zero eddy viscosity flow. Velocity magnitude $\|\mathbf{u}\|$ at $t = 2$ days obtained from DG scheme with $p = 1$, $h/4$-mesh, the convective flux of the LLF type, and the CBC implementation of the no-normal flow boundary condition. Solution of flow with: (a) $\nu_T = 5$ and (b) $\nu_T = 10$.
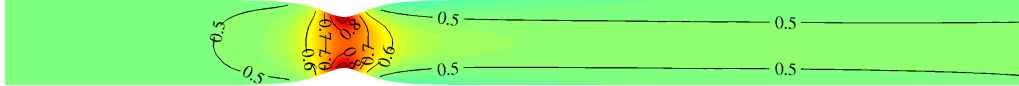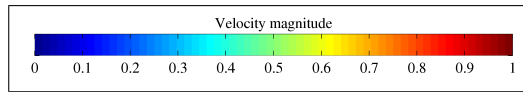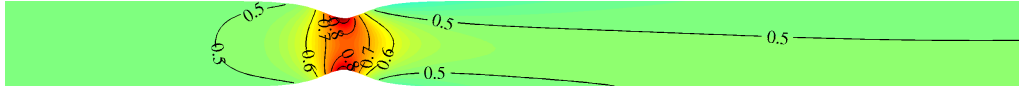
to the value of viscosity on the coarser-mesh calculations. This is reflected in Table 3 which lists the values of $\zeta_m \equiv \zeta(x = 0, y = 250)$, the surface elevation at the middle point of the channel entrance, obtained from the linear-element DG solutions on various computational meshes. In addition, it is evident from this table that the DG solution with the conventional no-normal flow implementation yields ζ_m that is sensitive to both the value of eddy viscosity and the resolution of the computational mesh. Data listed in Table 3 shows that the value of the eddy viscosity has less effect on ζ_m of the DG solution with the CBC approach. Furthermore, with the CBC approach, ζ_m from the

21

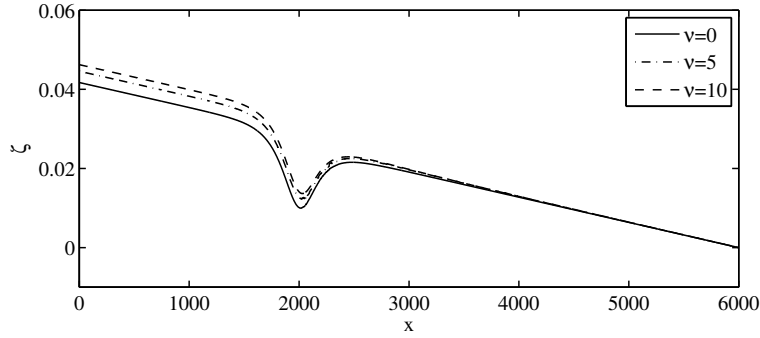(b) The conventional implementation
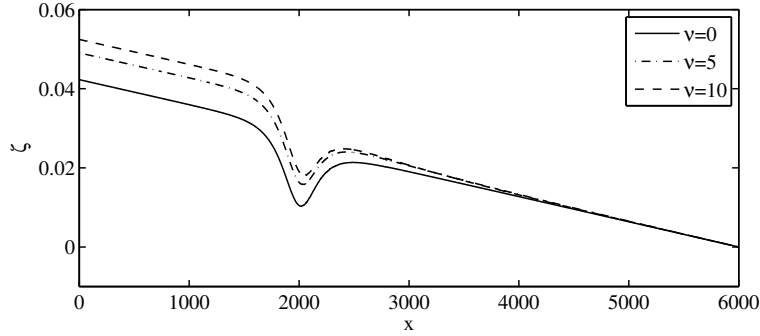
(b) The CBC approach

Figure 24: DG solutions to flow with $v_T = 0$, 5, and 10. Surface elevation of along $y = 250$ ($\zeta(x, y = 250)$) at $t = 2$ days from the schemes with $p = 1$, $h/8$-mesh, and the LLF flux for the numerical convective flux. Treatments of the no-normal flow boundary conditions: (a) the conventional approach; (b) the CBC approach.

Table 3: Values of surface elevation ζ at the point $(x, y) = (0, 250)$. Linear-element DG solutions to flow problem with various values of the eddy viscosity v_T.

| Comp. Mesh | $\zeta_m \equiv \zeta(x = 0, y = 250)$ | | | | | |
| | Conventional approach | | | CBC approach | | |
	$v_T = 0$	$v_T = 5$	$v_T = 10$	$v_T = 0$	$v_T = 5$	$v_T = 10$
h-mesh	0.0494	0.0594	0.0638	0.0426	0.0458	0.0481
$h/2$-mesh	0.0451	0.0532	0.0574	0.0419	0.0441	0.0470
$h/4$-mesh	0.0423	0.0491	0.0524	0.0417	0.0446	0.0464

coarse-mesh calculation differs slightly from that of the finer mesh calculations. In fact, it can be seen that ζ_m from the CBC approach on the coarsest mesh are comparable to the conventional no-normal flow approach on the $h/4$-mesh, a uniformly-two-level finer mesh. Figure 25 plots the velocity magnitude of the flow problem with $v_T = 5$ obtained from the DG solution with $p = 2$ on the h-mesh. Evidently, the boundary layers existing in the quadratic-element DG solution with the conventional no-normal flow approach appear to be significantly stronger that the linear-element DG solution (see 23(a) for the linear-element solution). Note that, in this case, the solution from the quadratic-element DG scheme with the LLF flux for $\widehat{\mathbf{F}} \cdot \mathbf{n}$ reaches steady state (see the quadratic-element, LLF-flux DG solution for the zero-viscosity flow in Figure 12(a)). It can be seen from Figure 25, the quadratic-element DG solution with the CBC approach has no artificial boundary layer. Clearly, this demonstrates the importance of a proper treatment of the
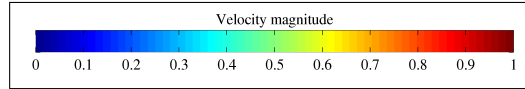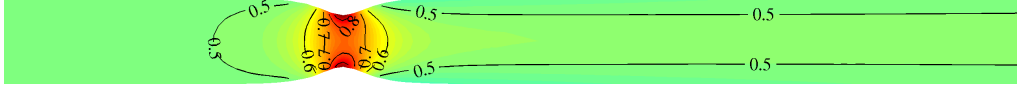
22

(a) The conventional approach
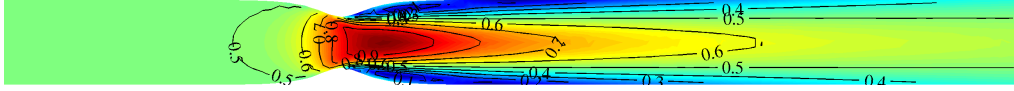
(b) The CBC approach

Figure 25: Velocity magnitude of the flow problem with $\nu_T = 5$ computed using the DG scheme with $p = 2$, h-mesh, the LLF flux as the numerical convective flux $\widehat{\mathbf{F}} \cdot \mathbf{n}$. The no-normal flow boundary condition is treated by: (a) the conventional approach; (b) the CBC approach.

no-normal flow conditions on curved walls in achieving realistic coarse grid solution in channels.

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. We consider two approaches, both of which use a straight-sided-element mesh with numerical calculations, for treating such a condition: the conventional approach and the CBC approach. The conventional approach, a commonly used technique, imposes the no-normal flow condition on the boundary of the straight-sided-element mesh, *i.e.* on the linear approximation of the solid curved wall. The CBC approach uses the exterior states that lead to a value of the numerical flux that approximates the physical no-normal flow boundary condition (rather than the no-normal flow boundary condition on the approximate boundary). We use, as a test problem, a subcritical flow problem that involves constant discharge of flow, through the converging-diverging channel, to an open ocean.

For the test problem without the momentum diffusion term from turbulence, the conventional no-normal flow implementation yields an approximate solution that visibly contains boundary layers downstream of the channel throat. These boundary layers are of one-grid-size thickness, which indicates that they are indeed artificial. The presence of boundary layers effectively leads to higher velocities away from the channel walls and thus an overprediction, especially on a coarse mesh, of the surface elevation on the upstream side. Errors introduced in the conventional no-normal flow approach appear to not only dominate but also manifest themselves especially in the LLF DG solution under p-refinement, as the solution exhibits poor quality as p increases. There is no boundary layer in the DG solution obtained from using the CBC no-normal flow implementation. In addition, the surface elevation obtained with this approach on the coarse mesh is qualitatively comparable to the solution computed on the fine mesh using the conventional no-normal flow implementation. Errors measured in the L_2- and L_∞-norm indicate that the DG scheme with the CBC approach yields convergent approximate solutions. The numerical solutions converge approximately at the optimal rate, *i.e.* of $O(h^2)$, for the linear-element DG method. The high-order schemes ($p > 1$), however, produce approximate solutions that converge at a sub-optimal rate. Although errors introduced by not matching the computational boundary to the physical boundary in the CBC approach are less than the conventional no-normal flow approach, it makes logical sense that they may increasingly become the major source of errors as p increases and

23

lead to sub-optimal convergence. Note that, to some degree, these convergence results suggest that the CBC approach offers more benefit for DG solutions with relatively low order interpolant ($p = 1$ or 2) and may be of less benefit for a very high order p.

Numerical experiments from applying DG to the flow problem with a non-zero eddy viscosity show that there are smeared boundary layers in the velocity field of the DG solution with the conventional no-normal flow approach. For the test set up, these artificial boundary layers result in the surface elevation on the upstream of the channel being susceptible to not only the mesh resolution but also the value of eddy viscosity, especially in the coarse-grid solution. A fine-grid solution is therefore required in this approach to obtain more quantitatively realistic results. The CBC approach yields an approximate solution that has no boundary layers. The upstream surface elevation using the CBC approach, as expected, is not as sensitive to the value of eddy viscosity and that obtained from the coarse-mesh calculation is in good agreement with the results on a fine-mesh calculation.

Numerical results provide clear evidence that, as observed in gas dynamics, a proper treatment of no-normal flow boundary conditions on the curved wall is crucial for achieving an 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 use an approach that accounts for errors introduced by the geometric approximation for the treatment of the curved walls. The use of curvilinear elements, which is an approach that accurately represents the geometry, is currently underway and will be reported in our future work.

7. Acknowledgements

Appendix

A. Numerical Fluxes

Below, $\mathbf{n}$ denotes a unit outward normal vector to an element edge. The value of the approximate solution on the element edge is denoted by $\boldsymbol{q}^{\text{in}}$ when approaching the edge from the interior of an element and $\boldsymbol{q}^{\text{ex}}$ when approaching the edge from the exterior of element (*i.e.* from neighboring element sharing the edge).

A.1. Local Lax-Friedrichs flux (LLF)

The LLF flux is given by

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

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

$$C = \max_{\mathbf{s} \in [\boldsymbol{q}^{\text{in}}, \boldsymbol{q}^{\text{ex}}]} \left| \lambda \left(n_x \left. \frac{\partial \mathbf{f}_1}{\partial q} \right|_{\mathbf{s}} + n_y \left. \frac{\partial \mathbf{f}_2}{\partial q} \right|_{\mathbf{s}} \right) \right| = \max_{\mathbf{s} \in [(\mathbf{u}, H)^{\text{in}}, (\mathbf{u}, H)^{\text{ex}}]} \left[\left(|\mathbf{n} \cdot \boldsymbol{u}| + \sqrt{gH} \right) \right] \tag{20}$$

where $\lambda(\cdot)$ represent the eigenvalue of the matrix.

A.2. Roe's numerical flux

The Roe numerical flux is an approximate Riemann solver. Such a flux is determined from solving (exactly) the so-called Roe linearized problem of the nonlinear normal Riemann problem [25] and, like the exact Riemann flux, is an upwind flux. Since the linear problem is considered, it eliminates the iterative procedure necessitated in determining the exact Riemann flux, thus offering a computational benefit. For a given data (q^{in}, q^{ex}; $\mathbf{n}$), the Roe numerical flux can be written as follows (see *e.g.* [25, 26] for a thorough treatment of Roe's approximate solver),

$$\widehat{\mathbf{F}} \cdot \mathbf{n} = \frac{1}{2} \left[\mathbf{F}(\mathbf{q}_h^{\text{in}}) + \mathbf{F}(\mathbf{q}_h^{\text{ex}}) \right] \cdot \mathbf{n} - \frac{1}{2} \widehat{\mathbf{R}} |\widehat{\lambda}| \widehat{\mathbf{R}}^{-1} \left(\mathbf{q}_h^{\text{in}} - \mathbf{q}_h^{\text{ex}} \right) \tag{21}$$

where

$$\widehat{\mathbf{R}} = \begin{pmatrix} 1 & 0 & 1 \\ \widehat{u} - \widehat{c} n_x & -n_y & \widehat{u} + \widehat{c} n_x \\ \widehat{v} - \widehat{c} n_y & n_x & \widehat{v} + \widehat{n}_y \end{pmatrix}, \tag{22}$$

$$|\widehat{\lambda}| = \begin{pmatrix} |\widehat{\mathbf{u}} \cdot \mathbf{n} - \widehat{c}| & 0 & 0 \\ 0 & |\widehat{\mathbf{u}} \cdot \mathbf{n}| & 0 \\ 0 & 0 & |\widehat{\mathbf{u}} \cdot \mathbf{n} + \widehat{c}| \end{pmatrix}, \tag{23}$$

In the above equations, n_x, n_y denote the x- and y-component of $\mathbf{n}$, respectively, $\widehat{c} \equiv \sqrt{g\widehat{H}}$ and the states $\widehat{H}, \widehat{u}, \widehat{v}$ denote the so-called Roe averages,

$$\widehat{H} = \frac{1}{2}(H^{\text{in}} + H^{\text{ex}}), \quad \widehat{u} = \frac{u^{\text{in}} \sqrt{H^{\text{in}}} + u^{\text{ex}} \sqrt{H^{\text{ex}}}}{\sqrt{H^{\text{in}}} + \sqrt{H^{\text{ex}}}}, \quad \text{and } \widehat{v} = \frac{v^{\text{in}} \sqrt{H^{\text{in}}} + v^{\text{ex}} \sqrt{H^{\text{ex}}}}{\sqrt{H^{\text{in}}} + \sqrt{H^{\text{ex}}}}. \tag{24}$$

References

[1] D. R. Lynch and W. G. Gray. A wave equation model for finite element tidal computations. *Computer & Fluids*, 7:207–228, 1979.

[2] R. A. Luettich, Jr, J. J. Westerink, and N. W. Scheffner. ADCIRC: An advanced three-dimensional circulation model for shelves, coasets, 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 and G. F. Carey. A symmetric formulation and SUPG scheme for shallow water equations. *Advances in Water Resoures*, 19:123–131, 1996.

[4] J. C. Galland, N. Goutal, and J. M. Hervouet. TELEMAC: a new numerical model for solving shallow water equations. *Advances in Water Resoures*, 14:138–148, 1991.

[5] B. Cockburn and C.-W. Shu. Runge-Kutta discontinuous Galerkin methods for convection-dominated problems. *Journal of Scientific Computing*, 16:173–261, 2001.

[6] B. Cockburn. Discontinuous Galerkin methods. *Zeitschrift fur Angewandte Mathematik und Mechanik*, 83:731–754, 2003.

[7] J. S. Hesthaven and T. Warburton. *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Application*. Springer Science+Business Media, LLC, 2008.

[8] V. Aizinger and C. Dawson. A discontinuous Galerkin method for two-dimensional flow and transport in shallow water. *Advances in Water Resoures*, 25:67–84, 2002.

[9] E. J. Kubatko, J. J. Westerink, and C. Dawson. hp Discontinuous Galerkin methods for advection dominated problems in shallow water flow. *Computer Methods in Applied Mechanics and Engineering*, 196:437–451, 2006.

[10] E. J. Kubatko, S. Bunya, C. Dawson, J. J. Westerink, and C. Mirabito. A performance comparison of continuous and discontinuous finite element shallow water models. *Journal of Scientific Computing*, 40:315–339, 2009.

[11] F. X. Giraldo and T. Warburton. A high-order triangular discontinuous Galerkin oceanic shallow water model. *International Journal for Numerical Methods in Fluids*, 56(7):899–925, July 2008.

[12] S. Bunya, E. J. Kubatko, J. J. Westerink, and C. Dawson. A wetting and drying treatment for the Runge-Kutta discontinuous Galerkin solution to the shallow water equations. *Computer Methods in Applied Mechanics and Engineering*, 198:1548–1562, 2009.

[13] C. Dawson, E. J. Kubatko, J. J. Westerink, C. Trahan, C. Mirabito, C. Michoski, and N. Panda. Discontinuous Galerkin methods for modeling hurricane storm surge. *Advances in Water Resources*, 34:1165–1176, 2010. DOI 10.1016/j.advwatres.2010.11.004.

[14] T. Kärnä, B. de Brye, O. Gourgue, J. Lambrechts, R. Comblen, V. Legat, and E. Deleersnijder. A fully implicit wetting-drying method for DG-FEM shallow water models, with an application to the scheldt estuary. *Computer Methods in Applied Mechanics and Engineering*, 200:509–524, 2011.

[15] C. Michoski, C. Mirabito, C. Dawson, D. Wirasaet, E. J. Kubatko, and J. J. Westerink. Dynamic *p*-enrichment schemes for multicomponent reactive flows. *Advances in water resources*, 34:1666–1680, 2011.

[16] C. Dawson, J. J. Westerink, J. C. Feyen, and D. Pothina. Continuous, discontinuous, and coupled discontinuous-continuous Galerkin finite elements methods for the shallow water equations. *International Journal for Numerical Methods in Fluids*, 52:63–88, 2006.

[17] D. Wirasaet, S. Tanaka, E. J. Kubatko, J. J. Westerink, and C. Dawson. A performance comparison of nodal discontinuous Galerkin methods on triangles and quadrilaterals. *International Journal for Numerical Methods in Fluids*, 64:1326–1362, 2010.

[18] D. Wirasaet, E. J. Kubatko, C. E. Michoski, S. Tanaka, J. J. Westerink, and C. Dawson. Discontinuous Galerkin methods with nodal and hybrid modal/nodal triangular, quadrilateral, and polygonal elements for nonliner shallow water flow. *Computer Methods in Applied Mechanics and Engineering*, 2012, in review.

[19] E. J. Kubatko, C. Dawson, and D. Wirasaet. Requirements for achieving high-order accuracy for discontinuous Galerkin solutions of the shallow water equations, 2011. *In preparation*.

[20] F. Bassi and S. Rebay. High-order accurate discontinuous finite element solution of the 2D euler equations. *Journal of Computational Physics*, 138:251–285, 1997.

[21] L. Krivodonova and M. Berger. High-order accurate implementation of solid wall boundary conditions in curved geometries. *Journal of Computational Physics*, 211:492–512, 2006.

[22] C. Mirabito, C. Dawson, E. J. Kubatko, J. J. Westerink, and S. Bunya. Implementation of a discontinuous Galerkin morphological model on two-dimensional unstructured meshes. *Computer Methods in Applied Mechanics and Engineering*, 200:189–207, 2011.

[23] P. A. Tassi, C. A. Vionnet S. Rhebergen, and O. Bokhove. A discontinuous Galerkin finite element model for river bed evolution under shallow flows. *Computer Methods in Applied Mechanics and Engineering*, 197:2930–2947, 2008.

[24] C. Michoski, C. Mirabito, C. Dawson, D. Wirasaet, E. J. Kubatko, and J. J. Westerink. Adaptive hierarchic transformations for dynamically *p*-enriched slope-limiting over discontinuous Galerkin systems of generalized equations. *Journal of Computational Physics*, 230(22):8028–8056, 2011.

[25] R. J. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, Cambridge, 2002.

[26] E. F. Toro. Riemann problems and the WAF method for solving the two-dimensoinal shallow water equations. *Philosophical Transactions: Physical Sciences and Engineering*, 338:43–68, 1992.

[27] Q. Zhang and C.-W. Shu. Error estimates to smooth solutions of Runge-Kutta discontinuous Galerkin methods for scalar conservative laws. *SIAM Journal on Numerical Analysis*, 42:641–666, 2004.

[28] F. Bassi and S. Rebay. Numerical evaluation of two discontinuous Galerkin methods for the compressible Navier-Stokes equations. *International Journal for Numerical Methods in Fluids*, 40:197–207, 2002.

[29] M. Dubiner. Spectral methods on triangle and other domains. *Journal of Scientific Computing*, 6:345–390, 1991.

[30] D. A. Dunavant. High degree efficient symmetrical Gaussian quadrature rules for the triangles. *International Journal for Numerical Methods in Engineering*, 21:1129–1148, 1985.

[31] S. Gottlieb, C.-W. Shu, and E. Tadmor. Strong stability-preserving high-order time discretization methods. *SIAM Review*, 43:89–112, 2001.

[32] E. J. Kubatko, J. J. Westerink, and C. Dawson. Semi-discrete discontinuous Galerkin methods and stage-exceeding-order, strong-stability-preserving Runge-Kutta time discretizations. *Journal Computational Physics*, 228:832–848, 2007.