Advances in Water Resources 34 (2011) 1165–1176

Contents lists available at ScienceDirect

Advances in Water Resources

journal homepage: www.elsevier.com/locate/advwatres

Discontinuous Galerkin methods for modeling Hurricane storm surge

Clint Dawson [a,*], Ethan J. Kubatko [b], Joannes J. Westerink [c], Corey Trahan [a], Christopher Mirabito [a], Craig Michoski [a], Nishant Panda [a]

[a] Institute for Computational Engineering and Sciences, 1 University Station, C0200, Austin, TX 78712, USA
[b] Department of Civil and Environmental Engineering and Geodetic Science, The Ohio State University, Columbus, OH 43210, USA
[c] Computational Hydraulics Laboratory, Department of Civil Engineering and Geological Sciences, 156 Fitzpatrick Hall, Notre Dame, IN 46556, USA

ARTICLE INFO

Article history:
Available online 27 November 2010

Keywords:
Discontinuous Galerkin methods
Hurricane storm surge
Shallow water equations

ABSTRACT

Storm surge due to hurricanes and tropical storms can result in significant loss of life, property damage, and long-term damage to coastal ecosystems and landscapes. Computer modeling of storm surge can be used for two primary purposes: forecasting of surge as storms approach land for emergency planning and evacuation of coastal populations, and hindcasting of storms for determining risk, development of mitigation strategies, coastal restoration and sustainability.

Storm surge is modeled using the shallow water equations, coupled with wind forcing and in some events, models of wave energy. In this paper, we will describe a depth-averaged (2D) model of circulation in spherical coordinates. Tides, riverine forcing, atmospheric pressure, bottom friction, the Coriolis effect and wind stress are all important for characterizing the inundation due to surge. The problem is inherently multi-scale, both in space and time. To model these problems accurately requires significant investments in acquiring high-fidelity input (bathymetry, bottom friction characteristics, land cover data, river flow rates, levees, raised roads and railways, etc.), accurate discretization of the computational domain using unstructured finite element meshes, and numerical methods capable of capturing highly advective flows, wetting and drying, and multi-scale features of the solution.

The discontinuous Galerkin (DG) method appears to allow for many of the features necessary to accurately capture storm surge physics. The DG method was developed for modeling shocks and advection-dominated flows on unstructured finite element meshes. It easily allows for adaptivity in both mesh (h) and polynomial order (p) for capturing multi-scale spatial events. Mass conservative wetting and drying algorithms can be formulated within the DG method.

In this paper, we will describe the application of the DG method to hurricane storm surge. We discuss the general formulation, and new features which have been added to the model to better capture surge in complex coastal environments. These features include modifications to the method to handle spherical coordinates and maintain still flows, improvements in the stability post-processing (i.e. slope-limiting), and the modeling of internal barriers for capturing overtopping of levees and other structures. We will focus on applications of the model to recent events in the Gulf of Mexico, including Hurricane Ike.

© 2010 Elsevier Ltd. All rights reserved.

1. Introduction

In this paper we describe recent advances in the application of discontinuous Galerkin (DG) methods to the modeling of shallow water flow in coastal environments. This work builds upon research of the authors and collaborators described in a number of recent papers [1,18–20,5]. Here we focus specifically on the application of DG methods to the modeling of coastal and inland flooding due to, e.g., storm surge from hurricanes or other tropical events.

Storm surge is primarily a competition between wind forcing and frictional resistance. As hurricanes approach the coast, water is driven inland often resulting in significant flooding, causing loss of life and damage to property and coastal ecosystems. Predicting and understanding the extent of surge is critical to emergency managers in the event of an impending storm, and to longer-term efforts to protect and sustain coastal environments. Computer models of storm surge are central to these efforts.

Hurricanes induce significant mixing through the water column in highly energetic storm events; therefore storm surge is modeled in this work by the depth-averaged shallow water equations. Boundary conditions on lateral boundaries of the 2D domain model tides, the interaction of water with dry land, and river inflows. In our model open ocean boundaries are extended out into the deeper ocean so as to prevent spurious boundary effects [3,4]. Body forces include wind stress, atmospheric pressure, bottom friction, Coriolis and tidal potential. In complex coastal regions such as the Gulf

* Corresponding author. Fax: +1 512 471 8694.
E-mail addresses: clint@ices.utexas.edu (C. Dawson), kubatko.3@osu.edu (E.J. Kubatko), jjw@nd.edu (J.J. Westerink).

0309-1708/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.advwatres.2010.11.004

Coast of the United States, channels, levees, raised roads and other internal barriers must be included in the description of the domain, as they either enhance or impede inland flow. The computational domain is discretized using triangular elements, to better represent complex coastal features, barrier islands, and internal barriers, and to allow for gradation of the mesh from the deeper ocean, the continental shelf, into estuaries, marshes, channels and over low-lying, potentially inundated coastal inlands.

Accurate modeling of storm surge requires numerical methods capable of capturing highly advective flows, wetting and drying, and multi-scale features of the solution. The DG method allows for many of the features necessary to accurately capture storm surge physics. The DG methodology we have adapted is based on the Runge–Kutta Local DG (RKLDG) methods of Cockburn and Shu [9] as modified by Cockburn and Dawson [8]; these methods were developed for modeling advection-dominated advection–diffusion equations on unstructured finite element meshes. In this method a "local" DG scheme approximates the diffusion terms, combined with a standard DG discretization for the advective and source terms. The RKLDG approach allows for adaptivity in both mesh (h) and polynomial order (p) for capturing multi-scale spatial events. Mass conservative wetting and drying algorithms can be formulated within the RKLDG method.

In this paper, we describe our initial efforts at applying the DG method to modeling storm surge. We first discuss the DG formulation in spherical coordinates, and then describe new features which have been added to the model to accurately capture surge in complex coastal environments. These features include improvements to the stability post-processing algorithms used in the code, and the modeling of internal "weir-type" barriers for capturing overtopping of levees and other structures. Finally, we present results of the model to a recent event in the Gulf of Mexico, Hurricane Ike. These results include comparisons to a well-known storm surge model, namely the Advanced Circulation (ADCIRC) model [21,22], which has been validated against several recent Gulf storms [6,11,12].

2. DG formulation in spherical coordinates

Tracking hurricanes and their effects through large sections of the ocean requires solving the shallow water equations on the sphere. We use a standard cylindrical projection to transform these equations into an equivalent set of equations in Cartesian coordinates.

Applying the hydrostatic and Boussinesq approximations, assuming the Earth's radius is large relative to the depth of the ocean, and averaging over the water depth H we arrive at the 2D governing equations in spherical coordinates (λ, ϕ):

- Continuity

$$\frac{\partial \zeta}{\partial t} + \frac{1}{R\cos\phi}\left(\frac{\partial (UH)}{\partial \lambda} + \frac{\partial (VH\cos\phi)}{\partial \phi}\right) = 0. \tag{1}$$

- Horizontal momentum

$$\frac{dU}{dt} = fV - \frac{1}{R\cos\phi}\frac{\partial[g(\zeta-\alpha\eta)+p_s/\rho_0]}{\partial\lambda} + \frac{\tau_{s\lambda}}{\rho_0 H} - \tau_{bf}U + m_\lambda, \tag{2}$$

$$\frac{dV}{dt} = -fU - \frac{1}{R}\frac{\partial[g(\zeta-\alpha\eta)+p_s/\rho_0]}{\partial\phi} + \frac{\tau_{s\phi}}{\rho_0 H} - \tau_{bf}V + m_\phi. \tag{3}$$

Here

- t = time
- λ, ϕ = degrees longitude, latitude
- ζ = free surface elevation relative to the geoid
- U, V = depth-averaged horizontal velocity components
- R = mean radius of the earth (6.3782064×10^6 m)
- $H = \zeta + h$ = water depth
- h = bathymetric depth relative to the geoid
- g = gravitational acceleration
- f = Coriolis coefficient
- p_s = atmospheric pressure at the free surface
- η = Newtonian equilibrium tide potential
- α = effective earth elasticity factor
- ρ_0 = reference density of water
- $\tau_{s\lambda}$, $\tau_{s\phi}$ = applied free-surface stress
- $\tau_{bf} = C_f[(U^2+V^2)^{1/2}/H]$ = bottom friction
- C_f = nonlinear bottom friction coefficient
- $\frac{d}{dt} = \frac{\partial}{\partial t} + \frac{U}{R\cos\phi}\frac{\partial}{\partial\lambda} + \frac{V}{R}\frac{\partial}{\partial\phi}$
- $m_\lambda, m_\phi = \frac{\nu_T}{H}\left[\frac{\partial}{\partial\lambda}\left(\frac{\partial UH}{\partial\lambda}\right) + \frac{\partial}{\partial\phi}\left(\frac{\partial UH}{\partial\phi}\right)\right],\ \frac{\nu_T}{H}\left[\frac{\partial}{\partial\lambda}\left(\frac{\partial VH}{\partial\lambda}\right) + \frac{\partial}{\partial\phi}\left(\frac{\partial VH}{\partial\phi}\right)\right],$
- ν_T = depth-averaged horizontal eddy viscosity

Using a standard, orthogonal cylindrical projection centered at a user-specified (λ_0, ϕ_0), we define

$$x = R(\lambda - \lambda_0)\cos\phi_0, \tag{4}$$

$$y = R\phi. \tag{5}$$

Using the chain rule we find

$$\frac{\partial}{\partial\lambda} = R\cos\phi_0\frac{\partial}{\partial x},$$

$$\frac{\partial}{\partial\phi} = R\frac{\partial}{\partial y}.$$

Replacing the derivatives in the spherical coordinate system with those in the Cartesian (x,y) system we find the transformed set of spherical equations

$$\frac{\partial\zeta}{\partial t} + S_p\frac{\partial(UH)}{\partial x} + \frac{\partial(VH)}{\partial y} = 0, \tag{6}$$

$$\frac{dU}{dt} = fV - S_p\frac{\partial[g(\zeta-\alpha\eta)+p_s/\rho_0]}{\partial x} + \frac{\tau_{s\lambda}}{\rho_0 H} - \tau_{bf}U + m_x, \tag{7}$$

$$\frac{dV}{dt} = -fU - \frac{\partial[g(\zeta-\alpha\eta)+p_s/\rho_0]}{\partial y} + \frac{\tau_{s\phi}}{\rho_0 H} - \tau_{bf}V + m_y, \tag{8}$$

where

$$S_p = \frac{\cos\phi_0}{\cos\phi}$$

is a spherical correction factor, the total derivative becomes

$$\frac{d}{dt} = \frac{\partial}{\partial t} + S_p\frac{\partial}{\partial x} + \frac{\partial}{\partial y}$$

and

$$m_x, m_y = \frac{\nu_T}{H}\left[S_p\frac{\partial}{\partial x}\left(\frac{\partial UH}{\partial x}\right) + \frac{\partial}{\partial y}\left(\frac{\partial UH}{\partial y}\right)\right],\ \frac{\nu_T}{H}\left[S_p\frac{\partial}{\partial x}\left(\frac{\partial VH}{\partial x}\right) + \frac{\partial}{\partial y}\left(\frac{\partial VH}{\partial y}\right)\right].$$

Finally we note that, multiplying (7) and (8) by H, using the continuity Eq. (6) and manipulating the gravity terms, the entire system can be written in divergence form:

$$\frac{\partial\mathbf{c}}{\partial t} + \nabla\cdot(\mathbf{A} - D\nabla\mathbf{c}) = \mathbf{h}(\mathbf{c}), \tag{9}$$

where

$$\mathbf{c} = \begin{pmatrix}\zeta\\ UH\\ VH\end{pmatrix},\quad \mathbf{A} = \begin{pmatrix} S_pUH & VH\\ S_p[U^2H + g(\zeta^2/2+\zeta h)] & UVH\\ S_pUVH & V^2H + g(\zeta^2/2+\zeta h)\end{pmatrix},$$

$$D = \begin{bmatrix}\mathbf{0} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \nu_T I & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \nu_T I\end{bmatrix}$$

and

$$\mathbf{h}(\mathbf{c}) = \begin{pmatrix} 0 \\ -\tau_{bf}UH + fVH + gS_p\zeta\frac{\partial h}{\partial x} + H\frac{\partial[g\alpha\eta - p_s/\rho_0]}{\partial x} + \frac{\tau_{s\lambda}}{\rho_0} \\ -\tau_{bf}VH - fUH + g\zeta\frac{\partial h}{\partial y} + H\frac{\partial[g\alpha\eta - p_s/\rho_0]}{\partial y} + \frac{\tau_{s\phi}}{\rho_0} \end{pmatrix}.$$

The matrix D has a block structure where $\mathbf{0}$ is the 2×2 zero matrix and I is the 2×2 identity matrix.

We note that the final set of Eq. (9) differs from the shallow water equations in standard Cartesian coordinates only in the factor S_p which appears in front of all of the x-derivative terms. This factor does depend on the latitude ϕ and hence is a function of y; therefore it is not constant, and we discuss the implications with respect to the DG discretization below. We now solve these equations on an (x,y) spatial domain Ω with appropriate boundary and initial conditions. For our purposes in this paper we only require two types of boundary conditions, open ocean and land:

- Open ocean boundary condition

$$\zeta = \zeta_{\text{tidal}}, \tag{10}$$
$$\nu_T\nabla(UH)\cdot\mathbf{n} = 0, \tag{11}$$
$$\nu_T\nabla(VH)\cdot\mathbf{n} = 0. \tag{12}$$

- Land boundary

$$\mathbf{q}\cdot\mathbf{n} = 0, \tag{13}$$

where ζ_{tidal} is a specified elevation typically determined by tides, and $\mathbf{n}$ is the unit outward normal to the boundary of Ω.

The RKLDG formulation of the system (9) has been described in several previous papers, most notably [1,20]. We briefly outline the formulation as follows.

Before describing the spatial discretization of (9) we define some notation. On any spatial domain R let $(\cdot,\cdot)_R$ denote the $L^2(R)$ inner product. To distinguish integration over domains $R \in \mathbb{R}^{d-1}$ (e.g., surfaces or lines), we will use the notation $\langle\cdot,\cdot\rangle_R$. Let $\{\mathcal{T}_h\}_{h>0}$ denote a family of finite element partitions of Ω such that no element Ω_e crosses the boundary of Ω, where h is the maximal element diameter. Let

$$W_{h,e} = \{\mathbf{v}: \text{ each component of } \mathbf{v} \text{ is a polynomial of degree } \leqslant k_e \text{ on } \Omega_e \in \mathcal{T}_h\}.$$

We do not specify here the number of components in $\mathbf{v}$ and below it may vary depending on the variable being approximated. Note that the degree k_e could vary from one element to the next. Let $\mathbf{n}_e$ denote the unit outward normal to $\partial\Omega_e$. Then, for $\mathbf{x} \in \partial\Omega_e$ we define

$$\mathbf{v}^{int}(\mathbf{x}) = \lim_{s\to 0^-}\mathbf{v}(\mathbf{x} + s\mathbf{n}_e)$$

and

$$\mathbf{v}^{ext}(\mathbf{x}) = \lim_{s\to 0^+}\mathbf{v}(\mathbf{x} + s\mathbf{n}_e).$$

That is, $\mathbf{v}^{int}$ is the value of $\mathbf{v}$ from the interior of Ω_e, and $\mathbf{v}^{ext}$ is the value of $\mathbf{v}$ from the exterior of Ω_e. We also define

$$\bar{\mathbf{v}} = (\mathbf{v}^{int} + \mathbf{v}^{ext})/2. \tag{14}$$

The LDG method is based on the following mixed form of (9). Define

$$\tilde{\mathbf{z}} = -\nabla\mathbf{c} \tag{15}$$

and

$$\mathbf{z} = D\tilde{\mathbf{z}}. \tag{16}$$

Then, substituting $\mathbf{z}$ into (9), multiplying by a sufficiently smooth test function $\mathbf{w}$, and integrating over an element Ω_e, we obtain

$$\left(\frac{\partial\mathbf{c}}{\partial t}, \mathbf{w}\right)_{\Omega_e} - (\mathbf{A} + \mathbf{z}, \nabla\cdot\mathbf{w})_{\Omega_e} + \langle(\mathbf{A} + \mathbf{z})\cdot\mathbf{n}_e, \mathbf{w}^{int}\rangle_{\partial\Omega_e} = (\mathbf{h}(\mathbf{c}), \mathbf{w})_{\Omega_e}. \tag{17}$$

Multiplying (15) by a suitable test function $\tilde{\mathbf{v}}$ and integrating we find

$$(\tilde{\mathbf{z}}, \tilde{\mathbf{v}})_{\Omega_e} - (\mathbf{c}, \nabla\cdot\tilde{\mathbf{v}})_{\Omega_e} + \langle\mathbf{c}, \tilde{\mathbf{v}}^{int}\cdot\mathbf{n}_e\rangle_{\partial\Omega_e} = 0 \tag{18}$$

and multiplying (16) by a test function $\mathbf{v}$ and integrating we obtain

$$(\mathbf{z}, \mathbf{v})_{\Omega_e} - (D\tilde{\mathbf{z}}, \mathbf{v})_{\Omega_e} = 0. \tag{19}$$

We approximate $\mathbf{c}$, $\tilde{\mathbf{z}}$ and $\mathbf{z}$ by functions $\mathbf{C}$, $\widetilde{\mathbf{Z}}$ and $\mathbf{Z}$ in $W_{h,e}$. We also approximate $\mathbf{A}\cdot\mathbf{n}_e$ on $\partial\Omega_e$ by a numerical flux $\widehat{\mathbf{A}}(\mathbf{C}^{int}, \mathbf{C}^{ext}; \mathbf{n}_e)$, which we discuss below. All other boundary terms are approximated by averaging. Thus, the DG method is given by

$$\begin{aligned}&\left(\frac{\partial\mathbf{C}}{\partial t}, \mathbf{w}\right)_{\Omega_e} - (\mathbf{A}(\mathbf{C}) + \mathbf{Z}, \nabla\cdot\mathbf{w})_{\Omega_e} \\ &\quad + \left\langle\widehat{\mathbf{A}}(\mathbf{C}^{int}, \mathbf{C}^{ext}; \mathbf{n}_e) + \overline{\mathbf{Z}}\cdot\mathbf{n}_e, \mathbf{w}^{int}\right\rangle_{\partial\Omega_e} \\ &\quad = (\mathbf{h}(\mathbf{C}), \mathbf{w})_{\Omega_e}, \quad \mathbf{w} \in W_{h,e},\end{aligned} \tag{20}$$

$$(\widetilde{\mathbf{Z}}, \tilde{\mathbf{v}})_{\Omega_e} - (\mathbf{C}, \nabla\cdot\tilde{\mathbf{v}})_{\Omega_e} + \langle\overline{\mathbf{C}}, \tilde{\mathbf{v}}^{int}\cdot\mathbf{n}_e\rangle_{\partial\Omega_e} = 0, \quad \tilde{\mathbf{v}} \in W_{h,e}, \tag{21}$$

$$(\mathbf{Z}, \mathbf{v})_{\Omega_e} - (D\widetilde{\mathbf{Z}}, \mathbf{v})_{\Omega_e} = 0, \quad \mathbf{v} \in W_{h,e}. \tag{22}$$

Remarks:

- The numerical flux $\widehat{\mathbf{A}}$ can be any consistent, locally Lipschitz, conservative entropy flux. In the numerical results in this paper, we used the Local Lax–Friedrichs (LLF) flux, given by

$$\widehat{\mathbf{A}}(C^-, C^+; \mathbf{n}) = \frac{1}{2}[\mathbf{A}(C^-)\mathbf{n} + \mathbf{A}(C^+)\mathbf{n} + |\Lambda|(C^- - C^+)], \tag{23}$$

  where Λ is an estimate of the largest eigenvalue of the Jacobian matrix of the flux $\mathbf{A}$ in the interval between C^- and C^+, and $\mathbf{n}$ is the normal to the edge which points from "minus" to "plus."
- The Eqs. (20)–(22) represent a system of ODE's. These ODE's are integrated in time using explicit Runge–Kutta methods [1].
- Our implementation of the DG method has been restricted to triangular elements. An orthogonal (Dubiner) basis is used as described in [20]. In the numerical results described below, we focus only on piecewise linear $k_e = 1$ approximations. We perform stability post-processing or slope limiting, described in Section 3. In this case the degree k_e may vary from 0 to 1 dynamically, depending on the action of the slope limiter.
- It is possible during a simulation that an element Ω_e can be wet ($H > 0$ throughout Ω_e), dry ($H = 0$), or partially wet ($H \geqslant 0$). The algorithm for handling wetting and drying is described in detail in [5].
- The boundary of an element may be part of an internal barrier; i.e., a levee, raised road, railway, etc. Internal barriers may start out as dry and be over-topped during the storm. Special formulas are used to handle the fluxes through these boundaries, as described in Section 4.

2.1. Preserving still flow

One question that arises in DG and other methods based on writing the shallow water equations in conservative form, is whether or not the method can preserve still flow. That is, if there are no forcing terms, and initially $U = V = 0$ and ζ is a constant, then ζ should remain constant for all time and U and V should remain zero. This can be especially tricky if the bathymetry h has discontinuities in it; however, we refer the reader to [24] for an elegant method for handling this case within the DG framework. In our

numerical results described below, the bathymetry is given at the vertices of each triangular element, and interpolated by a piecewise linear, continuous function. Therefore, ∇h is constant on each element.

First, note that if $\mathbf{C}$ is constant, then $\widetilde{\mathbf{Z}}$ and $\mathbf{Z}$ are zero by (21) and (22). In the presence of no forcing, the scheme is well-balanced if

$$-(\mathbf{A}(\mathbf{C}), \nabla \cdot \mathbf{w})_{\Omega_e} + \langle \widehat{\mathbf{A}}(\mathbf{C}^{int}, \mathbf{C}^{ext}; \mathbf{n}_e), \mathbf{w}^{int} \rangle_{\partial\Omega_e} = (\mathbf{h}(\mathbf{C}), \mathbf{w})_{\Omega_e}, \quad \mathbf{w} \in W_{h,e},$$

where $\mathbf{h}$ now only contains the gravity terms,

$$\mathbf{h} = \left(0, gS_p\zeta\frac{\partial h}{\partial x}, g\frac{\partial h}{\partial y}\right).$$

It can be shown that in standard Cartesian coordinates (when $S_p \equiv 1$) the scheme is well-balanced since h is continuous and piecewise linear. For spherical coordinates we must account for the fact that S_p is spatially dependent, in particular $S_p = S_p(y)$. Therefore, focusing on the x-momentum equation, noting that $U = V = 0$, we require

$$-\left(S_p(\zeta^2/2 + \zeta h), \frac{\partial w}{\partial x}\right)_{\Omega_e} + \langle (\zeta^2/2 + \zeta h) S_p n_x, w^{int} \rangle_{\partial\Omega_e} = \left(S_p\zeta\frac{\partial h}{\partial x}, w\right)_{\Omega_e}, \tag{24}$$

where ζ is now assumed constant everywhere. The factor S_p enters both the element and edge calculations. Recall that $S_p = \cos(\phi_0)/\cos(\phi) = \cos(\phi_0)/\cos(y/R)$. One could use this analytic form of S_p, but in order to maintain zero velocity we would have to integrate all terms involving S_p accurately over each element and edge. Instead we approximate S_p as follows.

On each edge we assume $S_p \approx \widehat{S}_p$, which is obtained by evaluating S_p at the midpoint of the edge. In addition, we compute an approximation to S_p on each element which is consistent with $\widehat{S}_p$. In the case where $k_e = 0$, S_p is approximated by a constant over each element, computed so that

$$\left(S_p, \frac{\partial \psi}{\partial x}\right)_{\Omega_e} = \langle \widehat{S}_p n_x, 1/2 + \psi \rangle_{\partial\Omega_e}, \tag{25}$$

where

$$\psi = \begin{cases} h, & \text{if } \frac{\partial h}{\partial x} \neq 0 \quad \text{on } \Omega_e, \\ x, & \text{otherwise}. \end{cases} \tag{26}$$

This is consistent with (24) where $w = 1$ and $\zeta = 1$. In the case that $k_e = 1$, we approximate $S_p \approx S_{p,e}$ which is in the linear Dubiner basis, by enforcing a type of Divergence Theorem:

$$\left(S_{p,e}(1/2 + \psi), \frac{\partial w}{\partial x}\right)_{\Omega_e} + \left(S_{p,e}\frac{\partial \psi}{\partial x}, w\right)_{\Omega_e} = \left\langle \widehat{S}_p(1/2 + \psi) n_x, w \right\rangle_{\partial\Omega_e} \tag{27}$$

for w linear on the element Ω_e.

3. Slope limiting for linear approximations

In the case where $k_e = 1$, we often need to apply a limiter or stability post-processing algorithm to prevent the solution from oscillating and eventually blowing-up. A number of limiters have been proposed in the literature, and an exhaustive study of all possible limiters is beyond the scope of this paper. We have implemented and tested three limiters specifically suited to triangular elements. The first limiter is based on a reconstruction scheme due to [14], and the second limiter is due to [10]. Both of these limiters are "edge-based," that is, on a given cell, one computes linear interpolants of the cell averages of the given cell and each of its neighboring elements which share an edge, taken two at a time. Thus, if a cell has three neighbors, one computes three linear interpolants, and compares these interpolants to the DG approximation. In the limiter in [14], the interpolant with the minimum gradient is chosen as the post-processed solution. The limiter in [10] is more complicated, it involves limiting in characteristic variables, but the basic idea is the same in that in involves neighbors which share an edge. In our experience, these limiters have performed well in some cases and not so well in others, as we will illustrate in the numerical results below.

Another limiter which we have implemented dates back to an idea first proposed in [2]. This limiter is "vertex-based." In this algorithm, we evaluate the linear DG solution at the vertex of each triangle. The basic idea then is to find the "closest" linear function to the DG solution which satisfies certain inequality constraints at the vertices. Let Ω_e be an element, and w a piecewise linear function defined on each element. At a vertex j of Ω_e, with coordinates (x_j, y_j), let w_j^-, w_j^+ denote the minimum and maximum cell average of w over all elements which share vertex j; i.e.,

$$w_j^- = \min_{\Omega_l \in K_j} w(x_l, y_l),$$

where K_j is the set of all elements sharing vertex j and (x_l, y_l) is the barycenter of Ω_l. A similar definition holds for w_j^+ with minimum replaced by maximum. We then check to see whether

$$w_j^- \leqslant w_e(x_j, y_j) \leqslant w_j^+, \tag{28}$$

where w_e represents w evaluated on Ω_e.

If (28) is violated, then we adjust the vertex values using a fairly simple heuristic. If the constraint is violated at a particular vertex, we adjust that vertex value just enough so that the inequality constraints are satisfied, however, we also want to preserve the cell average over Ω_e. The next step then is to distribute the excess (deficit) among the other vertices. The approach we use is to equidistribute the excess (deficit) among all of the other vertices subject to not causing any value to violate its inequality constraints. This can be done in a simple manner by making several passes over the vertices for equidistributing the values. We also note that it is always possible to satisfy (28) by setting the slope to zero in Ω_e, thus this is the default result if the equidistribution algorithm fails for some reason.

4. Modeling of internal barriers

Features within the domain such as levees, weirs, and raised roadways are treated as sub-grid scale internal barriers. Specifically, element edges are aligned with these features, which have

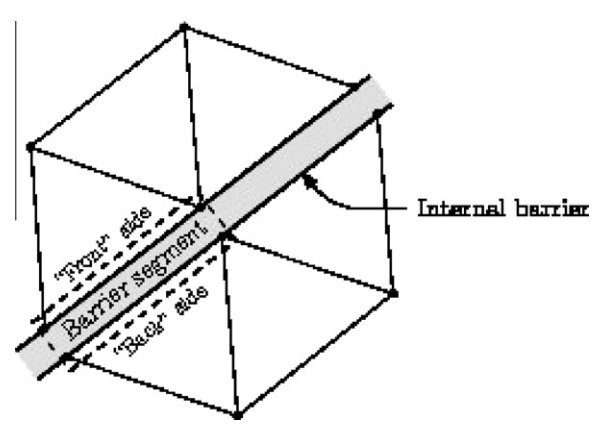

Fig. 1. An internal barrier and surrounding elements in a domain. Edge pairs on either side of the barrier are identified as barrier segments.

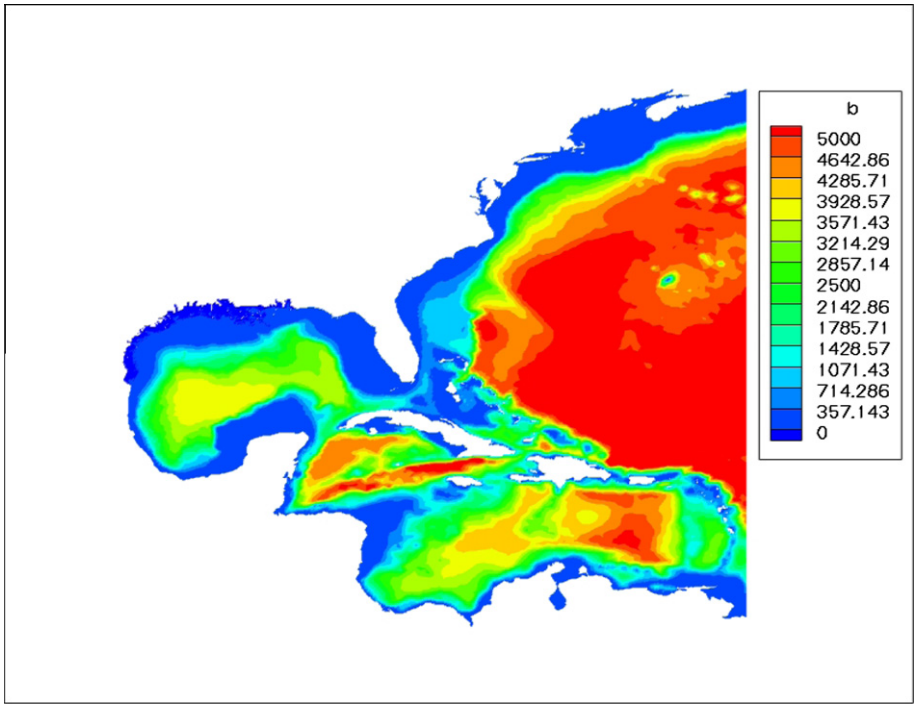

Fig. 2. Western North Atlantic domain and bathymetry in meters.

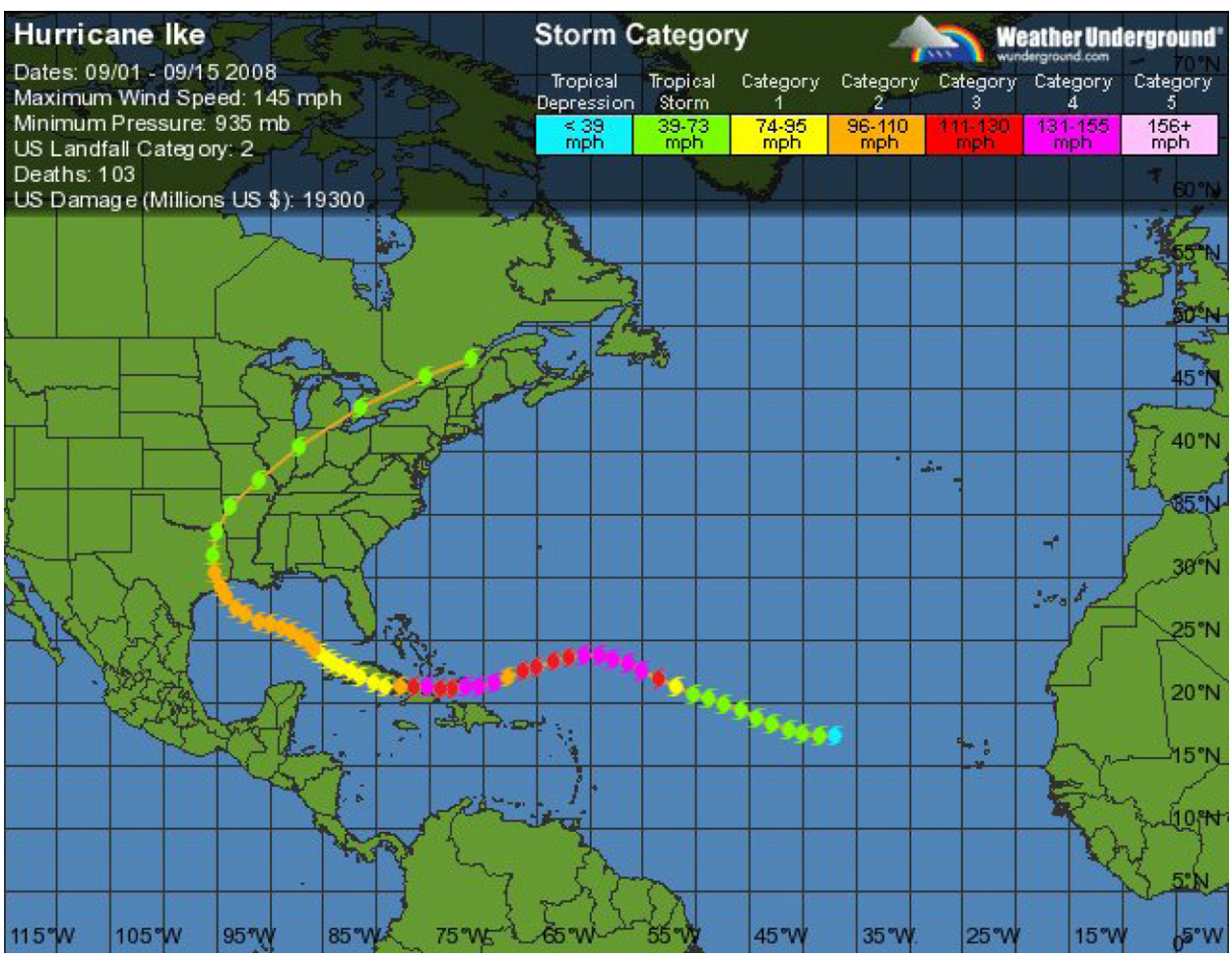

Fig. 3. Track of Hurricane Ike (http://www.wunderground.com).

some finite thickness, and their effects on the flow are incorporated into the model through the numerical flux terms computed in the evaluation of the boundary integral terms. To illustrate this procedure in more detail, consider the incorporation of the internal barrier shown in Fig. 1, which may represent, for example, a levee within the model domain. First, internal barrier *segments* are identified as pairs of edges that are "across" from one another, with one side arbitrarily labeled the "front" and the other labeled the "back". As outlined in Section 2, the numerical flux calculations make use of both interior (*int*) and exterior (*ext*) values of the variables $\mathbf{w}$. The exterior values of the variables on barrier segments are specified to enforce one of four main types of flow that may occur along the barrier based on the height of the free surface relative to the barrier height:

- *Case 1:* The free surface is below the height of the barrier on both the front and back sides of the segment. In this case, the internal barrier simply acts a solid wall, or no-normal flow, boundary; that is, the external variables on both sides are specified such that $\mathbf{q}\cdot\mathbf{n} = 0$ is enforced; see [1,20] for details.
- *Case 2:* The free surface is above the barrier segment on both sides and at the same height (within a specified tolerance). As in the case above, the external variables on both sides of the barrier segment are specified such that the condition $\mathbf{q}\cdot\mathbf{n} = 0$ is enforced.
- *Case 3:* The free surface on the front side of the barrier is greater than both the barrier height and the height of the free surface on the back side. In this case, overtopping of the barrier occurs with flow from the front to the back side of the barrier segment. Flow in this case will be one of two subcases based on flow conditions: subcritical or supercritical. Flow rates over the barrier $\mathbf{q}_{weir}$ are computed using standard weir formula; see, for example, [7]. The exterior values of $\mathbf{q}$ for both the front and back edges of the segment are then set accordingly so the numerical flux produces $\hat{\mathbf{q}} = \mathbf{q}_{weir}$.
- *Case 4:* The same as case 3 above but with overtopping of the barrier occurring from the back to the front side. Again, standard weir formula are used to compute the overtopping flow rate and exterior values of $\mathbf{w}$ are set such that $\hat{\mathbf{q}} = \mathbf{q}_{weir}$.

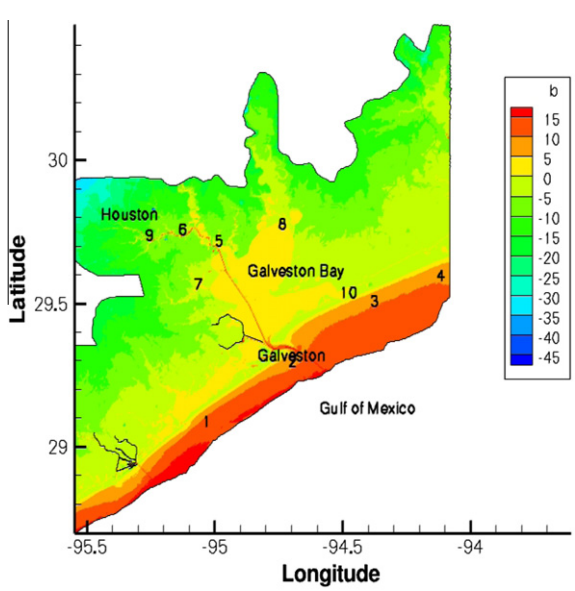

Fig. 4. Bathymetry (m) in Galveston Bay and the surrounding area. The Houston Ship Channel is the narrow line that runs through Galveston Bay from Galveston to Houston. Numbers indicate locations of hydrographs for Hurricane Ike simulations.

5. Numerical results

The main focus of this section is to give results of the DG method described above for Hurricane Ike, with comparisons to the ADCIRC (Advanced Circulation) model, which is a quasi-operational model that has been used to study many hurricanes in the Gulf of Mexico. In these studies, large domains which include the western North Atlantic Ocean are used, as seen in Fig. 2, discretized using graded triangular meshes which are highly refined in the area of interest, for example, Texas or Louisiana. The mesh diameters vary from kilometers in the deeper parts of the ocean, to hundreds of meters along the continental shelf and down to approximately fifty meters in the bays, estuaries, channels and inland regions impacted by the storm [6].

The track of Ike is shown in Fig. 3, the storm's primary impact occurred from Galveston Island eastward to Sabine Pass, which is on the Texas–Louisiana border. The effects of the storm on the areas around Galveston Bay are of particular interest, since this

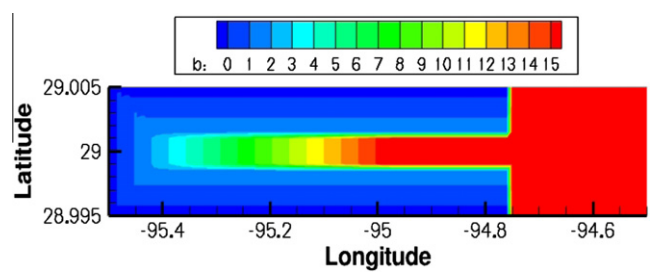

Fig. 5. Model channel domain. Bathymetry is in meters.

Table 1
Run parameters for model channel problem.

Length of simulation	1 day
Gravitational constant g	9.81 m/s^2
Bottom friction C_f	.0025
Coriolis f	.000070706 s^{-1}
Eddy viscosity ν_T	0
Slope limiter 1	Durlofsky–Engquist–Osher
Slope limiter 2	Cockburn and Shu
Slope limiter 3	Vertex
Δt	.5 s

Table 2
Hydrograph locations for model channel test case.

Number	Coordinates (lon/lat)
1	(−94.6,29)
2	(−94.8,29)
3	(−95.0,29)
4	(−95.2,29)
5	(−95.4,29)

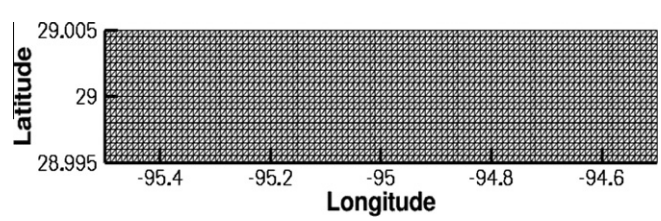

Fig. 6. Model channel grid.

area is heavily populated, is home to the Port of Houston and is vital to the US economy. Our numerical results focus primarily on computed maximum surge levels and hydrographs at various points around this region.

5.1. A comparison of slope limiters

Propagation of surge through narrow inland channels is a challenging but important problem. An example of such a regime is the

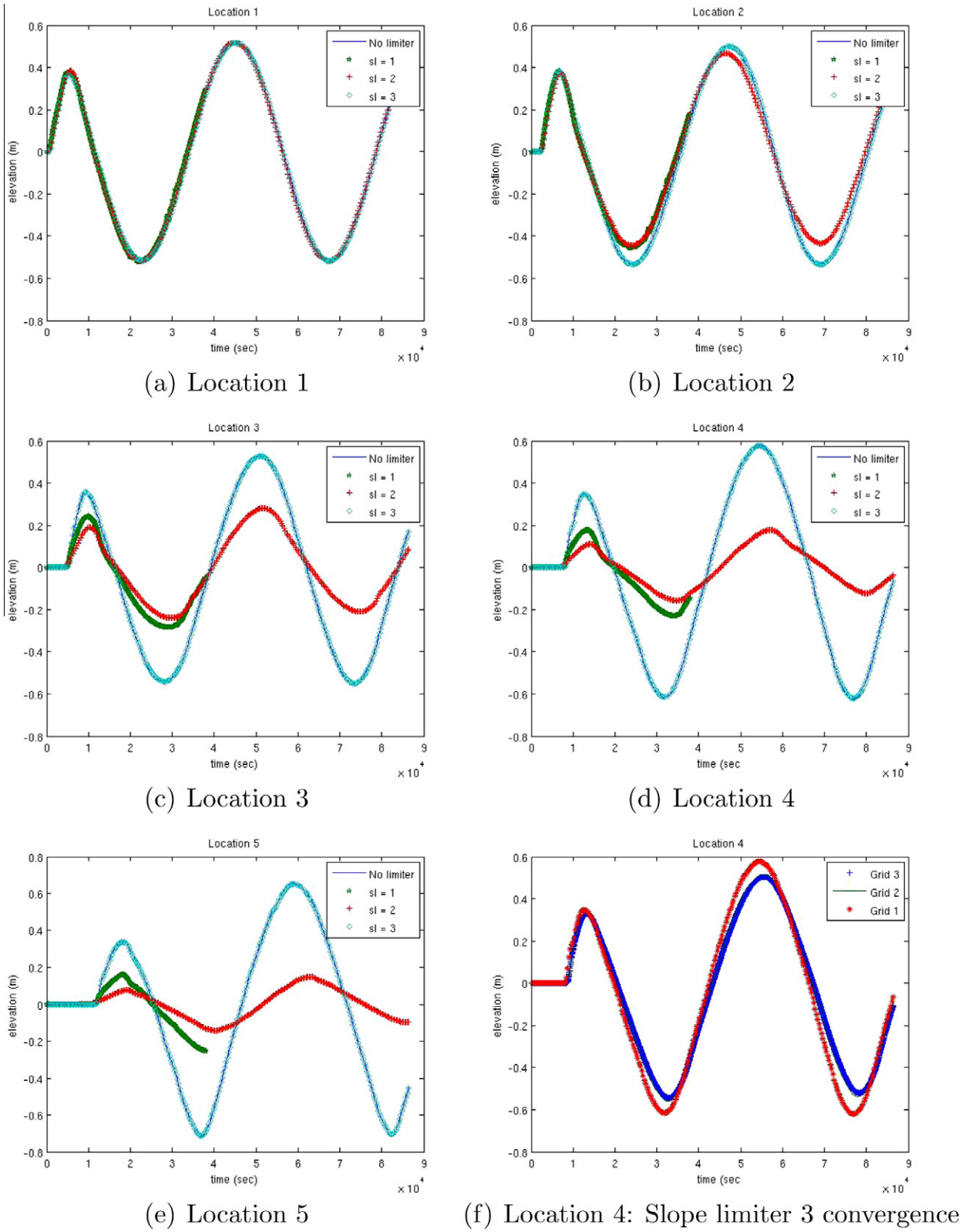

Fig. 7. Hydrographs of DG solutions with no limiting and slope limiters 1, 2 and 3 at measurement locations 1–5 (a–e); grid refinement study at location 4 for slope limiter 3 (f).

Houston Ship Channel, which connects the Port of Houston to the Gulf of Mexico through Galveston Bay. The channel is currently 160 m wide, 14 m deep, and 80 m long. The water depth in the surrounding bay averages around 2 m. The bathymetry in the bay is pictured in Fig. 4.

In order to test the ability of the DG method to propagate waves up a narrow channel, we developed a simple channel model with characteristics similar to the Houston Ship Channel, and simulated tidal flow through this channel. The model domain and bathymetry are shown in Fig. 5. The domain consists of an area of constant bathymetry between -94.8° and -94.5° longitude, connecting to a narrow channel of length approximately 80 km. The bathymetry is tapered to zero on the sides of the channel and towards the left end of the channel. The boundaries consist of an open ocean boundary on the right at $x = -95.5^\circ$, and all other boundaries are land boundaries. An $M2$ tide is specified with magnitude .5 m:

$$\zeta_{\text{tidal}}(t) = .5\cos(.0001405189025 * t),$$

where t is in seconds. The initial condition is a cold start; i.e., $\zeta = U = V = 0$. The tide is then ramped up over a period of .1 days. The friction law used is given by the Chezy formula

$$\tau_{bf} = C_f\sqrt{U^2 + V^2}/H,$$

with C_f given in Table 1. No atmospheric forcings are assumed. All other parameters are given in Table 1.

We computed DG solutions for orders $k_e = 1$ and for the three slope limiters mentioned in Section 3. Slope limiter 1 is the Durlofsky–Engquist–Osher reconstruction method, slope limiter 2 is the Cockburn–Shu limiter, and slope limiter 3 is the vertex-based limiter. We compare solutions by computing hydrographs at five locations in the domain, given in Table 2. The finite element mesh used in these comparisons is given in Fig. 6 and consists of 4000 elements and 2121 nodes (Grid 1).

In Fig. 7(a)–(e), we compare solutions with no slope limiting and slope limiters 1, 2 and 3. First, we remark that slope limiter 1 proved to be unstable, in fact, the solution blew up around $t = .44$ days. Reducing the time step did not resolve the problem, therefore we only show this solution up to $t = .44$ days. We note that the solutions with no limiting and using slope limiter 3 are virtually identical at all five measurement locations. The solutions at location 1 are virtually identical, as we move into the channel the solutions begin to differ. In particular, slope limiter 2 is much too diffusive, as we progress up the channel, by the time we reach measurement locations 4 and 5, the solution using this limiter has lost essentially all of the tidal signal.

Finally, as a check on the accuracy of the solution for slope limiter 3, we also solved the problem on two refined grids, one with 20,000 elements and 10,201 nodes (grid 2), and a second with 80,000 elements and 40,401 nodes (grid 3). For grid 2 the time step was $\Delta t = .125$ s and for grid 3, $\Delta t = .0625$ s. In Fig. 7(f), we compare solutions for all three grids at measurement location 4 using slope limiter 3. As seen in the figure, the solutions are comparable and appear to be converging, the grid 1 solution gives slightly higher peaks and troughs of the tidal wave. The solutions for grids 2 and 3 are virtually identical. We remark that we attempted to run the two finer grids with no slope limiting, and the DG solution blew up, even for very small time steps. Slope limiter 3 was able to compute stable solutions on all grids without being overly diffusive. Therefore, this limiter seems to provide the best combination of accuracy and stability of those we have tested to date.

5.2. Hurricane Ike

In this section, we present results from a hindcast study of Hurricane Ike. In this study, we simulate ten days of the storm, starting from September 5, 2008 at 12:00 UTC. The storm made landfall at Galveston, Texas approximately 8 days later on September 13. For these simulations we compare results between the DG code and the ADCIRC code. The ADCIRC code has been used extensively for studying Ike and many previous storms in the Gulf of Mexico, and excellent matches have been observed between ADCIRC model results, gauge data and high-water marks collected during these events. A complete description of the Hurricane Ike ADCIRC studies will be available in a forthcoming paper by some of the authors and several collaborators [16]. In short, ADCIRC has been validated for Ike and the purpose of this section is to see how the DG method compares to ADCIRC when using the exact same data from the storm.

The domain used for these studies is the Western North Atlantic model in Fig. 2. The finite element mesh consists of 6,615,381

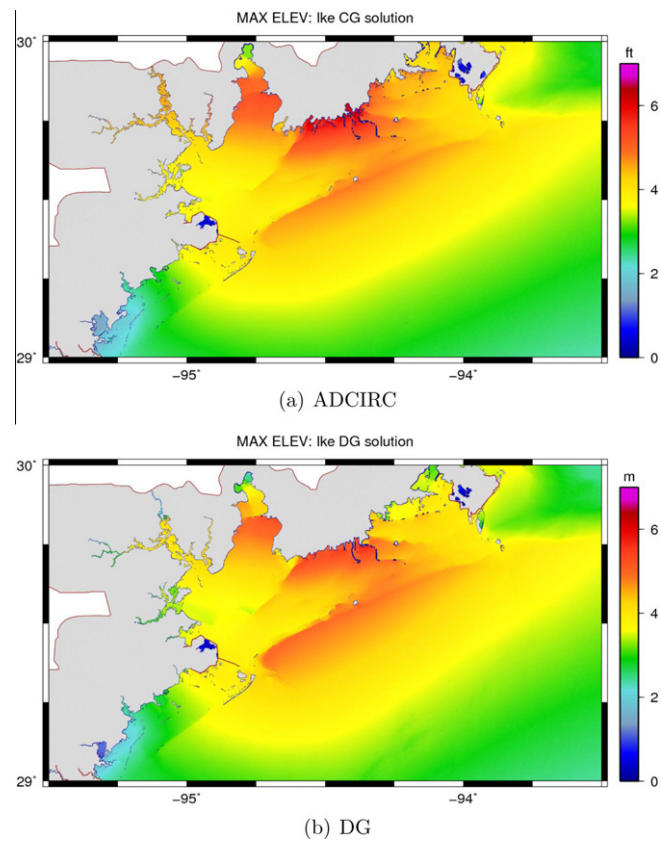

Fig. 8. Maximum water levels for ADCIRC (a) and DG (b) solutions during Hurricane Ike. Water elevation is in meters relative to the North American Vertical Datum of 1988 (NAVD88). The solution is plotted in the region between −93.5 and −95.5° longitude and 29 and 30° latitude.

Table 3
Hydrograph locations for Hurricane Ike simulation.

Number	Coord. (lon/lat)	Description
1	(−95.04,29.07)	Off-shore Galveston Island
2	(−94.71,29.28)	Off-shore Galveston Island
3	(−94.39,29.49)	Off-shore Bolivar Peninsula
4	(−94.13,29.58)	Off-shore Bolivar Peninsula
5	(−95.00,29.70)	Houston Ship Channel at SH 146 bridge
6	(−95.14,29.74)	Houston Ship Channel at Port of Houston
7	(−95.08,29.55)	Galveston Bay: Clear Creek entrance
8	(−94.75,29.76)	Galveston Bay: Upper Trinity Bay
9	(−95.27,29.72)	Manchester: Houston Ship Channel
10	(−94.51,29.52)	Bolivar Peninsula: Rollover Pass

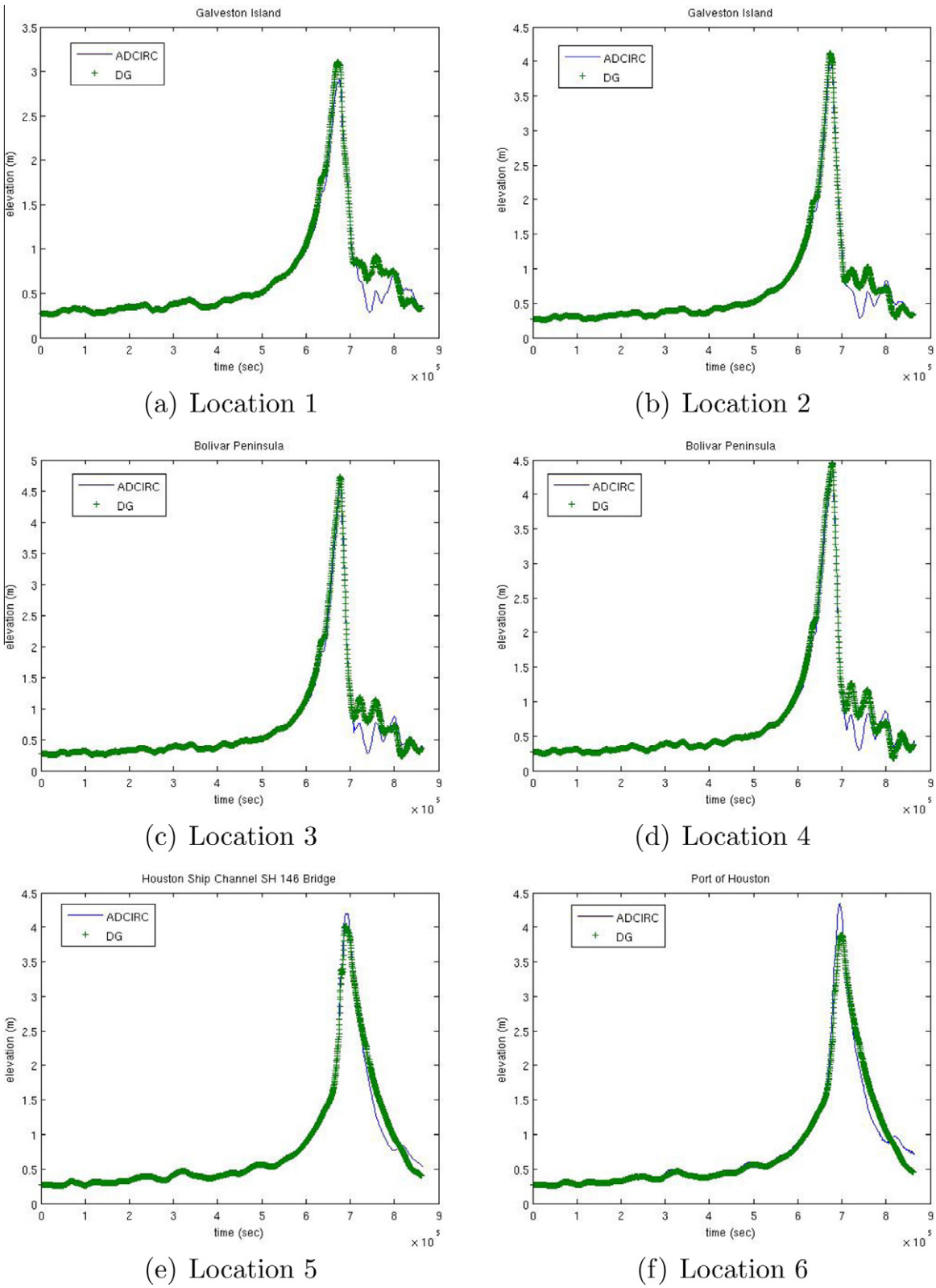

Fig. 9. Hydrographs of ADCIRC and DG solutions locations 1–6 for Hurricane Ike. Water levels are in meters relative to NAVD88.

elements and 3,322,439 nodes. A hybrid friction formula is used [6], with a spatially varying Mannings-n factor:

$$C_f = g\frac{n^2}{H^{1/3}}\left[1+\left(\frac{H_{\text{break}}}{H}\right)^{\theta_f}\right]^{\gamma_f/\theta_f}, \tag{29}$$

where n is Mannings-n, and the parameters $H_{\text{break}} = 2$ m, $\theta_f = 10$ and $\gamma_f = 4/3$. The wind field and atmospheric pressure used in this study are from a data-assimilated hindcast wind model developed by Ocean Weather, Inc. The wind surface stress is computed by a standard quadratic drag law.

Define

$$\frac{\hat{\tau}_{sx}}{\rho_0} = C_d \frac{\rho_{\text{air}}}{\rho_0} |\mathbf{W}| W_x, \tag{30}$$

$$\frac{\hat{\tau}_{sy}}{\rho_0} = C_d \frac{\rho_{\text{air}}}{\rho_0} |\mathbf{W}| W_y. \tag{31}$$

Here $\mathbf{W} = (W_x, W_y)$ is the wind speed sampled at a 10-m height over a 15 min time period and ρ_{air} is the air density. The wind speed is adjusted to account for local roughness directionally, for the level of local inundation, and for forested canopies as described in [6]. The drag coefficient is defined by Garratt's drag formula [15]:

$$C_d = (.75 + .06|\mathbf{W}|) * 10^{-3}. \tag{32}$$

We also remark that the wind surface stress is capped at .002, thus

$$|\tau_{sx}| = \min(.002, |\hat{\tau}_{sx}|), \quad |\tau_{sy}| = \min(.002, |\hat{\tau}_{sy}|). \tag{33}$$

We also use a spatially varying eddy viscosity parameter ν_T which is non-zero throughout the domain. Further details on the parameters used in the model can be found in [16]. The simulation was cold-started and no tidal forcing was used. Therefore the model is forced by wind, atmospheric pressure, bottom friction and Coriolis.

For these runs, we used a time step of $\Delta t = .5$ s in both the DG and ADCIRC codes. The DG code results were generated using piecewise linear approximations with slope limiter 3.

In Fig. 8, we compare the maximum water elevations computed over the duration of the storm for ADCIRC and the DG method. That is, we plot

$$\max_{0<t\leqslant T} \zeta(\cdot, t). \tag{34}$$

The figures focus only on the region of maximum inundation, which includes the Houston–Galveston metropolitan area and points eastward. The maximum storm surge occurred in an area east of Galveston Bay in Chambers County, Texas. As observed in these figures, the results agree qualitatively over most regions of the domain; however, significant disagreement in the results can be seen in the western and northwestern channels of Galveston Bay and in the upper reaches of the Houston Ship Channel. The DG solution shows higher surge along the Bolivar Peninsula, while the ADCIRC solution shows greater inland penetration to the north of Bolivar Peninsula and in the northeastern parts of Galveston Bay, possibly due to differences in the wetting and drying algorithms used in the two models.

Probing further, we compare hydrographs at several near-shore locations along the Texas coast, in Galveston Bay, and upward through the Houston Ship Channel. The descriptions of these locations are given in Table 3 and their physical locations can be seen in Fig. 4.

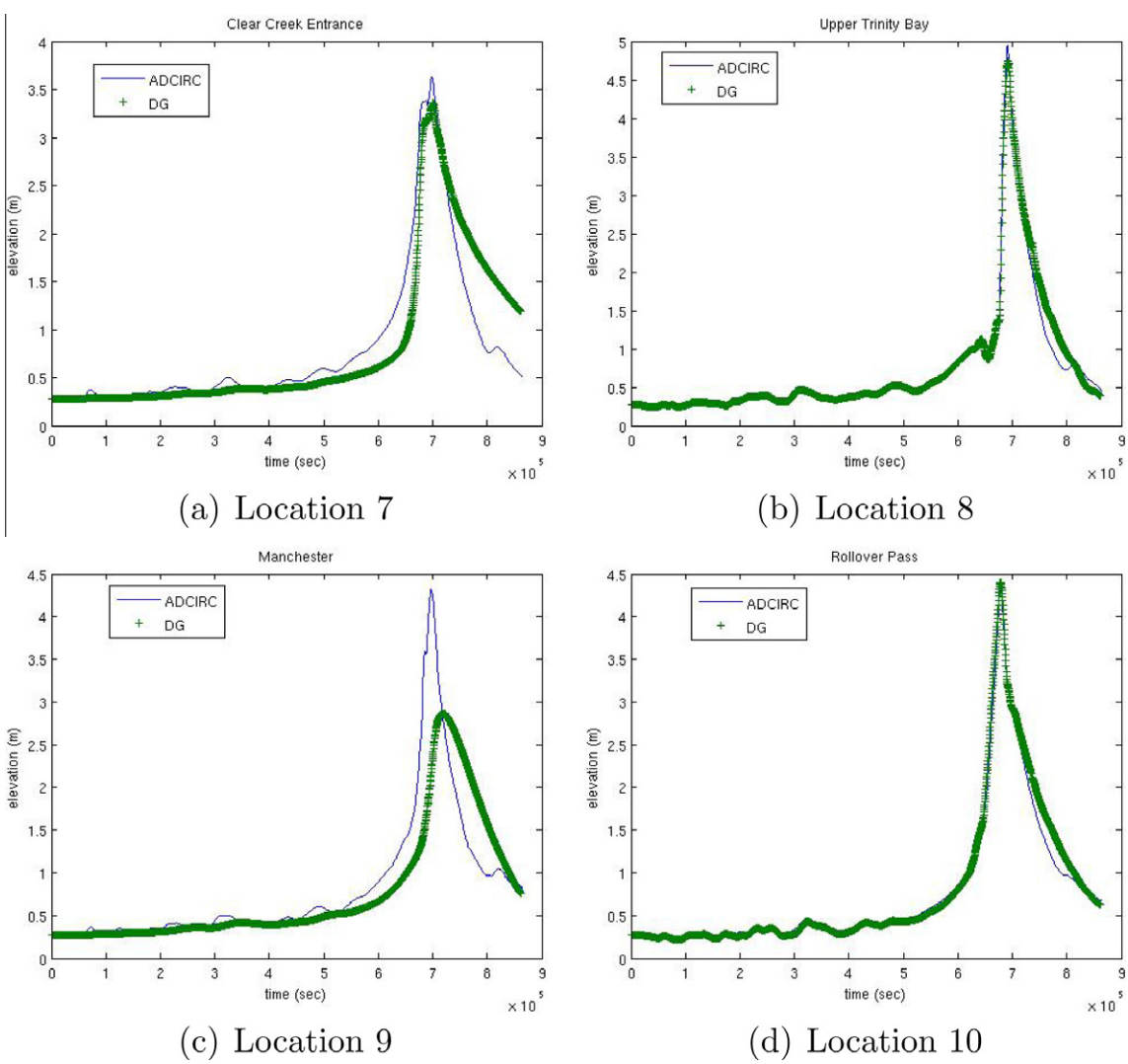

Fig. 10. Hydrographs of ADCIRC and DG solutions locations 7–10 for Hurricane Ike. Water levels are in meters relative to NAVD88.

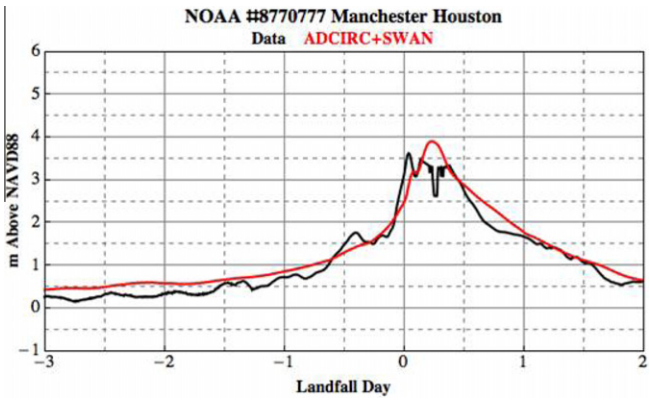

Fig. 11. Comparison of ADCIRC + SWAN computed water levels with NOAA gauge data at location 9 (Manchester, Houston, Texas).

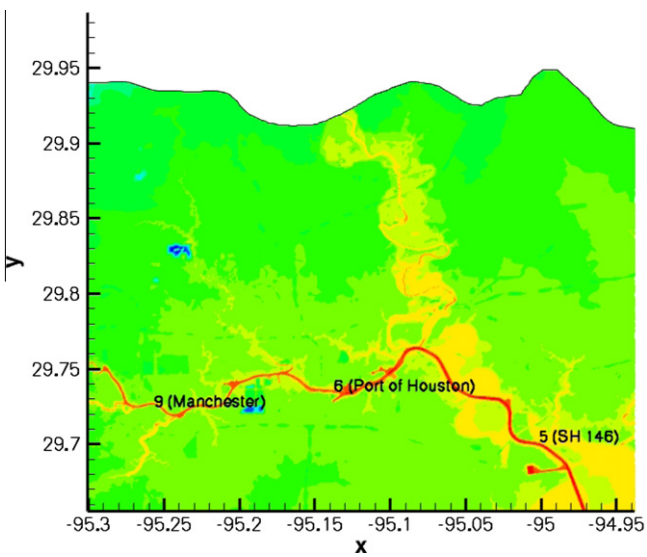

Fig. 12. Closer view of the upper reaches of the Houston Ship Channel, which shows measurement locations 5 (SH 146 bridge), 6 (Port of Houston) and 9 (Manchester). Colors represent bathymetric depth.

The comparisons are given in Figs. 9(a)–10(d). The two codes show agreement in areas off-shore near Galveston Island and the Bolivar peninsula, and in the lower and central parts of Galveston Bay. There is however a major discrepancy at location 9, which is near the community of Manchester at the farthest end of the Houston Ship Channel. The actual measured maximum surge level at location 9 matches well with the ADCIRC result, as seen in Fig. 11. We note that the numerical result in this figure was computed using a coupled wave-current model ADCIRC + SWAN (Simulating Waves Nearshore) [13]. Therefore, additional forcing due to wave radiation stresses are included; however the maximum surge is close to that observed in Fig. 10(c). If we look at the results at nearby locations 5 and 6 we begin to observe differences between the codes; in particular, the maximum surge levels in the DG solution are not as high as those produced by ADCIRC and diminish as one moves up the channel. One possible cause of this discrepancy can be explained by inspection of the ship channel and how it is discretized. In Fig. 12, we show the upper part of the Houston Ship Channel and the relative positions of measurement locations 5, 6 and 9. We observe that the channel narrows past the Port of Houston towards Manchester. In this region, there are several locations where a string of nodes has been placed to follow

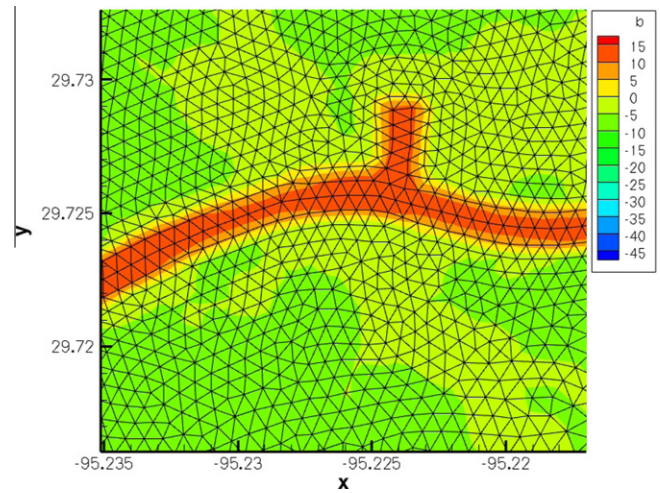

Fig. 13. Bathymetry with finite element mesh overlaid along a stretch of the Houston Ship Channel near Manchester. Bathymetry (b) is in meters. Note that a string of nodes follows the deepest part of the channel and in some places the channel is split across two elements.

the deepest part of the channel, with the channel itself straddling two elements, as seen in Fig. 13. This type of mesh is well-suited for a continuous Galerkin (nodal-based) discretization, as the degrees of freedom for elevation and velocity are defined at the nodes, and the flow up the channel essentially reduces to a quasi-1D solution along the nodes lying in the channel. This mesh is less suited for a DG method, which is not nodal but element and edge-based; a more suitable mesh would have at least one element across the width of the channel. While this is a plausible explanation for the discrepancy between the codes, it may not be the only cause, and deserves further examination.

6. Conclusions

The numerical simulation of Hurricane Ike given above indicates the complexities inherent in storm surge modeling: extreme wind forcings, highly varying bathymetry, complex coastal geometries with internal barriers, inlets and narrow channels, highly nonlinear bottom friction, and wetting and drying. In this paper, we have demonstrated that the DG method with piecewise linear approximations is generally competitive with the highly-tuned and quasi-operational ADCIRC model. The drawbacks of the ADCIRC model going forward are that it is restricted to linear polynomials on conforming meshes, has difficulty handling strong advection (in fact, advection terms have been turned off in ADCIRC for the Hurricane Ike simulations), and requires tuning the so-called "τ_0" parameter in the generalized wave continuity formulation [23,17]. The ADCIRC model is also not locally mass conservative, though this is not a critical issue for storm surge modeling at present.

In this paper, we have developed and implemented a new slope limiter which improves the performance of the DG method in inlets and narrow channels. Unfortunately, in our Hurricane Ike simulations, we still see what appears to be over-damping of the surge in the furthest reaches of the Houston Ship Channel, and further research is required to develop DG methods which are both accurate and stable for these complex situations. Approaches currently under investigation include the use of dynamic p and h adaptivity. In future work, we will examine more closely p adaptivity within the DG method and the stability of these methods for complex flow scenarios. We are also investigating the use of more general meshes, including the use of hybrid meshes based on mixing

quadrilateral and triangular elements. We note that the mesh used in our Hurricane Ike simulations was constructed for ADCIRC simulations and thus may be more suitable for a continuous, nodal-based finite element method than an element and edge-based DG method. The sensitivity of the DG method to various kind of meshes, especially in narrow channels, is also an area of current research.

We conclude with some comments on the computational performance of the DG method. One criticism of DG methods is their cost relative to continuous Galerkin methods, and this criticism is warranted in many cases. The DG storm surge calculations performed here were done on the Ranger supercomputer at the Texas Advanced Computing Center (TACC), as were the ADCIRC simulations. Presently, using the same finite element mesh, time step, and number of CPUs, the ADCIRC code is on average about 4 times faster than the DG code. Studies comparing the ADCIRC and DG codes for both efficiency and accuracy for some simple model problems can be found in [19]. These studies showed that the DG method was competitive with and in some cases superior to ADCIRC when measuring accuracy vs. parallel efficiency. Improving the overall performance of the DG method, through the use of adaptivity, more sophisticated time-stepping (including local time-stepping), the use of quadrature-free methods, and by using better software practices, is also the subject of future research.

Acknowledgments

The authors acknowledge support from the National Science Foundation, grants OCI-0749015 and OCI-0746232. Computational resources were provided by the National Science Foundation TeraGrid, project number TG-DMS080016N, and the Texas Advanced Computing Center at The University of Texas at Austin.

References

[1] Aizinger V, Dawson C. A discontinuous Galerkin method for two-dimensional flow and transport in shallow water. Adv Water Res 2002;25:67–84.
[2] Bell JB, Dawson C, Shubin G. An unsplit, higher-order Godunov method for scalar conservation laws in two dimensions. J Comput Phys 1988;74:1–24.
[3] Blain CA, Westerink JJ, Luettich RA. The influence of domain size on the response characteristics of a hurricane storm surge model. J Geophys Res 1994;99(C9):18467–79.
[4] Blain CA, Westerink JJ, Luettich RA. Grid convergence studies for the prediction of hurricane storm surges. Int J Numer Meth Fluids 1998;26:369–401.
[5] Bunya S, Kubatko EJ, Westerink JJ, Dawson C. A wetting and drying treatment for the Runge–Kutta discontinuous Galerkin solution to the shallow water equations. Comput Meth Appl Mech Eng 2009;198:1548–62.
[6] Bunya S, Dietrich JC, Westerink JJ, Ebersole BA, Smith JM, Atkinson JH, et al. A high-resolution coupled riverine ow, tide, wind, wind wave and storm surge model for Southern Louisiana and Mississippi: Part I – Model development and validation. Mon Weather Rev 2010;138:345–77. doi:10.1175/2009MWR2906.1.
[7] Chow VT. Open channel hydraulics. New York, NY: McGraw-Hill; 1959.
[8] Cockburn B, Dawson C. Some extensions of the local discontinuous Galerkin method for convection–diffusion equations in multidimensions. In: Whiteman JR, editor. The proceedings of the conference on the mathematics of finite elements and applications: MAFELAP, vol. X. Elsevier; 2000.
[9] Cockburn B, Shu C-W. The local discontinuous Galerkin finite element method for convection–diffusion systems. SIAM J Numer Anal 1998;35:2440–63.
[10] Cockburn B, Shu C-W. TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws II: general framework. Math Comp 1989;52:411–35.
[11] Dietrich JC, Bunya S, Westerink JJ, Ebersole BA, Smith JM, Atkinson JH, et al. A high-resolution coupled riverine ow, tide, wind, wind wave and storm surge model for Southern Louisiana and Mississippi: Part II – Synoptic description and analyses of Hurricanes Katrina and Rita. Mon Weather Rev 2010;138:378–404. doi:10.1175/2009MWR2907.1.
[12] Dietrich JC, Westerink JJ, Kennedy AB, Smith JM, Jensen R, Zijlema M, et al. Hurricane Gustav (2008) waves and storm surge: Hindcast, synoptic analysis, and validation in Southern Louisiana. Mon Weather Rev, submitted for publication.
[13] Dietrich JC, Zijlema M, Westerink JJ, Holthuijsen LH, Dawson C, Luettich RA, et al. Modeling hurricane wave and storm surge using integrally-coupled scalable computations. Coastal Eng 2011;58:45–65. doi:10.1016/j.coastaleng.2010.08.001.
[14] Durlofsky LJ, Engquist B, Osher S. Triangle based adaptive stencils for the solution of hyperbolic conservation laws. J Comp Phys 1992;98:64–73.
[15] Garratt JR. Review of drag coefficients over oceans and continents. Mon Weather Rev 1977;105:915–29.
[16] Kennedy AB, Gravois U, Zachry B, Westerink JJ, Hope M, Dietrich JC, et al. Hurricane Ike forerunner surge: origin and implications. Geophys Res Lett, submitted for publication.
[17] Kinnmark IPE, The shallow water wave equations: formulation, analysis, and application. Ph.D. thesis, Department of Civil Engineering, Princeton University, 1984.
[18] Kubatko E, Bunya S, Dawson C, Westerink JJ. Dynamic *p*-adaptive Runge–Kutta discontinuous Galerkin methods for the shallow water equations. Comput Meth Appl Mech Eng 2009;198:1766–74.
[19] Kubatko EJ, Bunya S, Dawson C, Westerink JJ, Mirabito C. A performance comparison of continuous and discontinuous finite element shallow water models. J Sci Comput 2009;40:315–39.
[20] Kubatko EJ, Westerink JJ, Dawson C. *hp* Discontinuous Galerkin methods for advection dominated problems in shallow water flow. Comput Meth Appl Mech Eng 2006;196:437–51.
[21] Luettich RA, Westerink JJ, Scheffner NW, ADCIRC: an advanced three-dimensional circulation model for shelves, coasts and estuaries. Report 1: Theory and methodology of ADCIRC-2DDI and ADCIRC-3DL, Dredging Research Program Technical Report DRP-92-6, US Army Engineers Waterways Experiment Station, Vicksburg, MS, 1992.
[22] Luettich RA, Westerink JJ. ADCIRC: a parallel advanced circulation model for oceanic, coastal and estuarine waters: Users manual. Available from http://www.adcirc.org.
[23] Lynch DR, Gray WR. A wave equation model for finite element computations. Comput Fluids 1979;7:207–28.
[24] Rhebergen S, Bokhove O, van der Vegt JJW. Discontinuous Galerkin finite element methods for hyperbolic nonconservative partial differential equations. J Comp Phys 2008;227:1887–922.