



Time adaptivity in the diffusive wave approximation to the shallow water equations

Nathan Collier^{a,*}, Hany Radwan^a, Lisandro Dalcin^b, Victor M. Calo^a

^a Applied Mathematics and Computational Science and Earth and Environmental Sciences and Engineering, King Abdullah University of Science and Technology, Thuwal, Saudi Arabia

^b Centro Internacional de Métodos Computacionales en Ingeniería, Santa Fe, Argentina

ARTICLE INFO

Article history:

Received 15 March 2011

Received in revised form 10 June 2011

Accepted 13 July 2011

Available online 22 August 2011

Keywords:

Time adaptivity
Shallow water flow
Overland flow

ABSTRACT

We discuss the use of time adaptivity applied to the one dimensional diffusive wave approximation to the shallow water equations. A simple and computationally economical error estimator is discussed which enables time-step size adaptivity. This robust adaptive time discretization corrects the initial time step size to achieve a user specified bound on the discretization error and allows time step size variations of several orders of magnitude. In particular, the one dimensional results presented in this work feature a change of four orders of magnitudes for the time step over the entire simulation.

© 2011 Elsevier B.V. All rights reserved.

1. Introduction

The diffusive wave approximation of the shallow water (DSW) equations is used to model overland flows such as floods, dam breaks, and flows through vegetated areas. The shallow water equations (SWEs) are obtained from the full Navier–Stokes equations by introducing the following simplifying assumption: the vertical momentum scales are small relative to those of the horizontal momentum, that is, due to depth restrictions the velocity structures in the horizontal direction are much larger than the ones in the vertical one. This assumption reduces the vertical momentum equation to a hydrostatic pressure relation, which is integrated in the vertical direction and results in a two dimensional system of equations known as the shallow water equations. Further details and discussion of the scaling assumptions made can be found in [1].

The DSW equation is a further simplification of the shallow water equations (or St. Venant equation in one spatial dimension). The velocity term in the continuity equation is replaced with an empirical relationship, such as Manning's equation [2], from open channel flow. The horizontal momentum equations are then used to further reduce the system under the simplification that the slope of the water surface does not vary much from the slope of the bathymetry. The DSW equation is applicable to overland flow which is fully turbulent and driven mainly by gravitational forces. The DSW equation reduces to a scalar equation which resembles

nonlinear diffusion. While the nonlinearities present challenges, the DSW equation is a simpler framework with which economically simulate shallow flows. The properties of solutions to the DSW equations have been studied [3–5]. In the more recent paper [3], convergence of the numerical scheme was studied numerically and error estimates are derived.

In this paper, we address computational aspects of solving the DSW equation, specifically we focus on time integration and adaptivity. We use the generalized- α method for time integration, which is a second-order accurate method with controlled dissipation for high frequencies [6,7]. We describe a simple and robust error estimator which can be used to guide time adaptivity. Time adaptivity in the solution of DSW is useful for two reasons. First, the adaptivity aids in determining the initial time step. After selecting an initial step size, the algorithm will automatically reduce it to acceptable tolerances, relative to the quality of the error estimator. Second, a significant amount of computation may be avoided by allowing the time step to change, while keeping the error under a user-prescribed tolerance. Our numerical results show that in 1D tests, the time step can change several orders of magnitude over the simulation time. In particular, the simulations presented herein, after the initial time-step size adjustment, the time-step size varies by four orders of magnitude.

2. DSW equations

The DSW strong form is obtained by special assumptions which simplify the shallow water equations, leading to the following

* Corresponding author.

E-mail address: nathaniel.collier@gmail.com (N. Collier).

initial/boundary value problem on the domain Ω for times $t \in [0, T]$ [3],

$$\begin{cases} \dot{u} - \nabla \cdot (\kappa(u, \nabla u) \nabla u) = f & \text{on } \Omega \times (0, T) \\ u = u_0 & \text{on } \Omega \times \{t = 0\} \\ (\kappa(u, \nabla u) \nabla u) \cdot n = B_N & \text{on } \Gamma_N \times (0, T) \\ u = B_D & \text{on } \Gamma_D \times (0, T), \end{cases} \quad (1)$$

where u is the water height, \dot{u} is its time derivative, f is a forcing function such as rainfall acting as a source or infiltration acting as a sink, u_0 is the initial condition, B_N and B_D are the Neumann and Dirichlet conditions, respectively, and the diffusion coefficient κ is given by

$$\kappa(u, \nabla u) = \frac{(u - z)^{\alpha_M}}{C_f |\nabla u|^{1-\gamma_M}}.$$

The bathymetry is represented by the function z , and α_M and γ_M are constants which specify the empirical method used to obtain the DSW equation. Following [3] we use parameters corresponding to Manning’s formula, $\alpha_M = 5/3$ and $\gamma_M = 1/2$. The function C_f represents Manning’s coefficient which is also known as a friction coefficient. Typical values are experimentally measured and available in the literature, but for the sake of simplicity we assume $C_f = 1$.

The equation is doubly degenerate in that the diffusion disappears in cases where $u = z$ or when ∇u becomes large such as in regions where the solution represents a wave front. This creates difficulties in developing a numerical solution technique which can handle these difficulties. These properties are discussed in detail in [3].

The weak form for the DSW equation is to find $u \in \mathcal{V}$ such that for every $w \in \mathcal{W}$,

$$B(w, u) = \left(w, \frac{\partial u}{\partial t} \right) + (\nabla w, \kappa(u, \nabla u) \nabla u)_{\Omega} + (w, f)_{\Omega} = 0, \quad (2)$$

where $(\cdot, \cdot)_{\Omega}$ refers to the L^2 inner product and the trial, \mathcal{V} , and weighting, \mathcal{W} , spaces are appropriately chosen for Eq. (2) to be well defined [3]. A discrete approximation to the solution is obtained constructing a Galerkin approximation appropriately choosing proper subspaces of \mathcal{V}_h and \mathcal{W}_h of \mathcal{V} and \mathcal{W} , respectively [9,10]. The discrete function space chosen in the numerical example described in Section 4 is obtained using 1024 linear elements, while the L^2 inner product over the the domain is approximated using Gaussian quadrature, where four Gauss points per element are used.

3. Time discretization

We advance in time using the generalized- α method [6,7]. The method was originally developed in [6] for structural dynamics, which is of second order in time. Subsequently it was extended for first order problems in time in [7]. This time-stepping methodology has been successfully applied to several nonlinear problems such as, turbulent simulations [7,11], and more recently first time adaptive technique was proposed for the Cahn–Hilliard equations [12] and subsequently used in [13] to model bubble formation and evolution.

The generalized- α method for first order in time problems is stated as follows: given (u_n, \dot{u}_n) , find $(u_{n+1}, \dot{u}_{n+1}, u_{n+\alpha_f}, \dot{u}_{n+\alpha_m})$, such that

$$R(u_{n+\alpha_f}, \dot{u}_{n+\alpha_m}) = 0, \quad (3)$$

$$u_{n+\alpha_f} = u_n + \alpha_f(u_{n+1} - u_n), \quad (4)$$

$$\dot{u}_{n+\alpha_m} = \dot{u}_n + \alpha_m(\dot{u}_{n+1} - \dot{u}_n), \quad (5)$$

$$u_{n+1} = u_n + \Delta t((1 - \gamma)\dot{u}_n + \gamma\dot{u}_{n+1}), \quad (6)$$

where the A^{th} component of the residual vector $R_A = B(N_A, u^h)$, where u^h represents the finite element solution, $\Delta t = t_{n+1} - t_n$ and α_f, α_m , and γ are real-valued parameters. It has been shown in [7] that for a linear model problem, unconditional stability is attained if $\alpha_m \geq \alpha_f \geq (1/2)$, and second order accuracy can be achieved with $\gamma = (1/2) + \alpha_m - \alpha_f$. The method can be stated as a one-parameter method, where α_m, α_f and γ can all be expressed in terms of a parameter known as spectral radius, ρ_{∞} . We select $\alpha_m = 5/6$ and $\alpha_f = \gamma = 2/3$ which correspond to $\rho_{\infty} = 1/2$. The generalized- α algorithm is detailed in Algorithm 1. The algorithm is a predictor/multi-corrector method where the corrector steps are indicated by a superscript index inside of parenthesis.

Algorithm 1. Generalized- α method

```

1: Compute predictors [7,12]  $u_{n+1}^{(0)} = u_n$  and  $\dot{u}_{n+1}^{(0)} = \frac{\gamma-1}{\gamma}\dot{u}_n$ 
2:  $i = 1$ 
3: while  $i <$  maximum iterations do
4:    $u_{n+\alpha_f}^{(i)} = u_n + \alpha_f (u_{n+1}^{(i-1)} - u_n)$ 
5:    $\dot{u}_{n+\alpha_m}^{(i)} = \dot{u}_n + \alpha_m (\dot{u}_{n+1}^{(i-1)} - \dot{u}_n)$ 
6:    $R_{n+1}^{(i)} = R(u_{n+\alpha_f}^{(i)}, \dot{u}_{n+\alpha_m}^{(i)})$ 
7:    $K_{n+1}^{(i)} = \alpha_m \frac{\partial R(u_{n+\alpha_f}^{(i)}, \dot{u}_{n+\alpha_m}^{(i)})}{\partial u_{n+\alpha_m}} + \alpha_f \gamma \Delta t \frac{\partial R(u_{n+\alpha_f}^{(i)}, \dot{u}_{n+\alpha_m}^{(i)})}{\partial u_{n+\alpha_f}}$ 
8:   Solve  $K_{n+1}^{(i)} \Delta \dot{u}_{n+1}^{(i)} = -R_{n+1}^{(i)}$ 
9:    $\dot{u}_{n+1}^{(i)} = \dot{u}_{n+1}^{(i-1)} + \Delta \dot{u}_{n+1}^{(i)}$ 
10:   $u_{n+1}^{(i)} = u_{n+1}^{(i-1)} + \gamma \Delta t_n \Delta \dot{u}_{n+1}^{(i)}$ 
11:  if  $\|R_{n+1}^{(i)}\| \leq \epsilon_{\alpha} \|R_{n+1}^{(0)}\|$  then
12:    stop
13:  end if
14:   $i = i + 1$ 
15: end while

```

3.1. Time adaptivity

Time step adaptivity is achieved by a simple error predictor for first order methods. Given a u_n, \dot{u}_n , and Δt_n the solution at t_{n+1} may be computed given the generalized- α algorithm described in Algorithm 1. For $\rho_{\infty} = 1/2$, we can then use \bar{u}_{n+1} to make a first order approximation of u_{n+1}

$$\bar{u}_{n+1} = u_n + \Delta t_n \dot{u}_{n+1}. \quad (7)$$

which is obtained by manipulating a first-order Taylor expansion from u_{n+1} to u_n . With this inexpensive approximation we can estimate the error as

$$E_{n+1} = \frac{\|\bar{u}_{n+1} - u_{n+1}\|}{\|u_{n+1}\|} \quad (8)$$

and adapt the time-step size using the typical equation [12,14–16],

$$\Delta t_{n+1} = \rho \left(\frac{\epsilon}{E_{n+1}} \right)^{1/2} \Delta t_n. \quad (9)$$

Following [12], the factor of safety ρ was chosen to be 0.9 and the tolerance ϵ to be 10^{-3} . The time step is rejected if $E_{n+1} > \epsilon$ and recomputed once Δt is modified. We added an additional constraint in the implementation of this algorithm which restricts the growth of the time step at a given time,

$$\frac{1}{10} \leq \frac{\Delta t_{n+1}}{\Delta t_n} \leq 5. \quad (10)$$

We found this last constraint to be useful in situations where the step size grows and then suddenly must be reduced.

4. Numerical results

The spatial, linear finite elements, and temporal, generalized- α discretizations were applied to the solution of a 1D problem, which models a fictitious dam break, arrested by a series of two dikes. The bathymetry is shown in Fig. 1(a), indicated by grey shading. The source of water is modeled by a Neumann boundary condition on

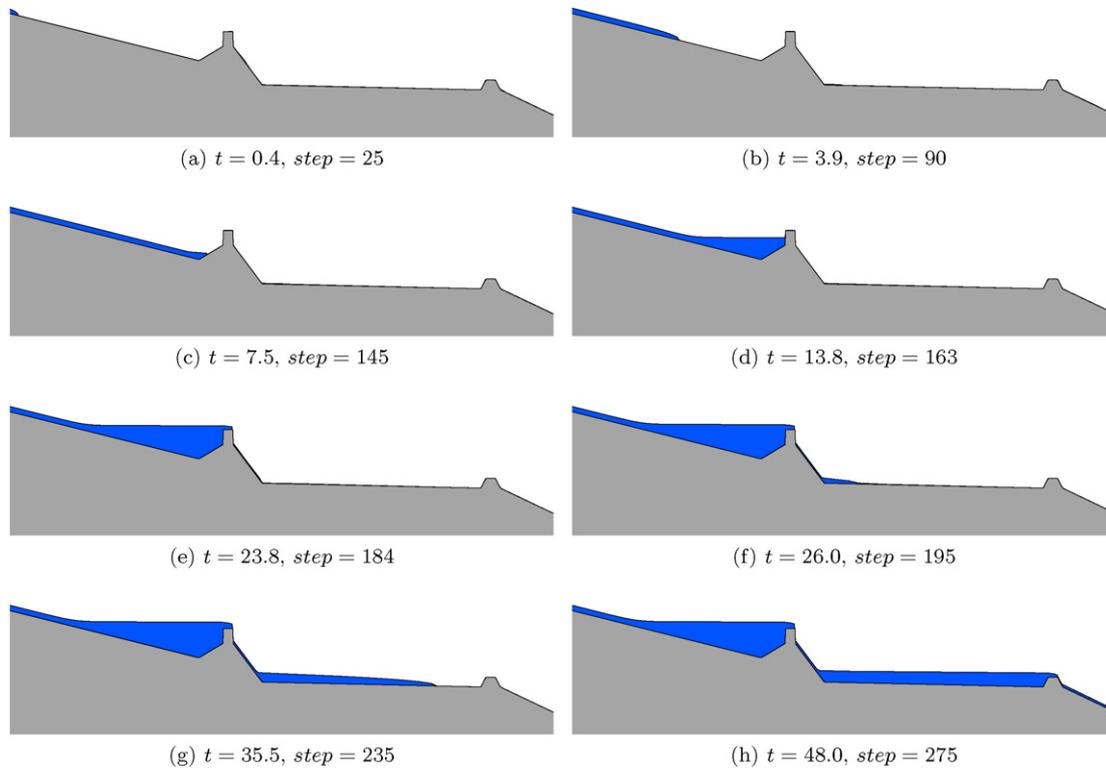


Fig. 1. Time sequence of solutions to DSW with bathymetry shown in grey and the water flow shown in blue. This sequence models the arrest of a dam break by a series of dikes. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of the article.)

the left side and a zero Dirichlet condition on the right. This example illustrates how the solution of the DSW can benefit from adaptive time stepping.

Fig. 1(a)–(h) is a graphical representation of the solution obtained by our numerical simulation. As time progresses, the downhill flow (Fig. 1(a)–(c)) is arrested by a large dike (Fig. 1(d)) where it pools until the water level overtakes the dike (Fig. 1(e)). A second, smaller dike is reached after a long plain where the flow is arrested again (Fig. 1(f) and (g)) until finally the second dike is also overtaken (Fig. 1(h)).

Fig. 2 emphasizes the usefulness of time adaptivity in this problem. From step number 2 to 50, the time step grows two orders of magnitude where it then remains roughly constant until step 145 (Fig. 1(c)). At this point, the flow is being slowed down by the first dike and begins to pool. The water height is not changing drastically during this phase of the simulation and so the time step grows further until the water begins to flow over the first dike, step number 170. Large and sudden changes in the water height require a smaller step size and so the step is reduced again. As the simulation continues, two other size changes are seen:

1. as the flow hits the bottom of the first dike on the right side (around step number 200)
2. as the flow passes over the second dike (around step number 260).

We observe that time adaptivity is a great advantage when flow must pass sharp obstacles such as these dikes.

While each subfigure is titled with the time at which the solution is displayed, these times do not correspond to real flooding events. The frictional coefficient, C_f , in the DSW equations is taken here to be unity. This choice allows us to focus on the time adaptive scheme where flow velocity is controlled by topographic gradients. Notice that in spite of have a constant C_f the diffusivity is not constant,

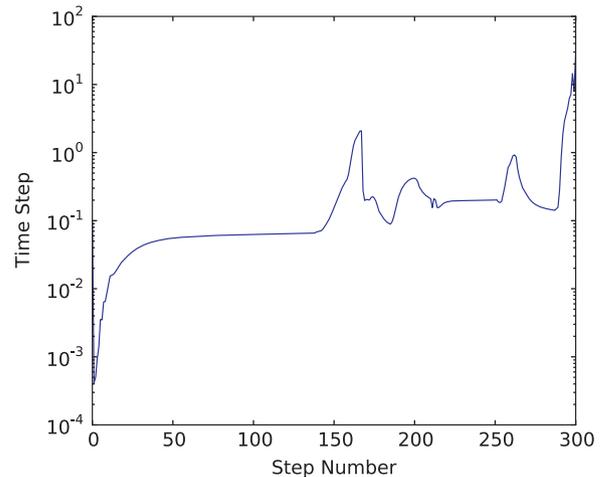


Fig. 2. The evolution of the time step growth over the simulation.

due its non-linear dependence on both the water height, u , and its gradient, ∇u .

Time step adaptivity amounts to great savings in computational costs. The time adaptive simulation described in the above paragraph required 275 time steps to reach completion. The smallest time step size was on the order 10^{-3} . A simulation running at this constant time step size would need 48,000 steps to run to steady state.

In our numerical tests, the adaptivity also did not overly affect the solution. We ran a simulation using a constant time step size of 10^{-4} and 1024 elements. We saved the solution at 35.5 s, designating it \bar{U}_{1024} . Then for a series of meshes, $N = \{128, 256, 512, 1024\}$, we ran the simulation using time adaptivity with $\epsilon = 10^{-3}$.

Table 1
Relative errors between time step adaptive simulations ($\epsilon = 10^{-3}$) and \bar{U}_{1024} .

N	128	256	512	1024
E_{L_2}	5.0%	4.1%	3.7%	0.1%

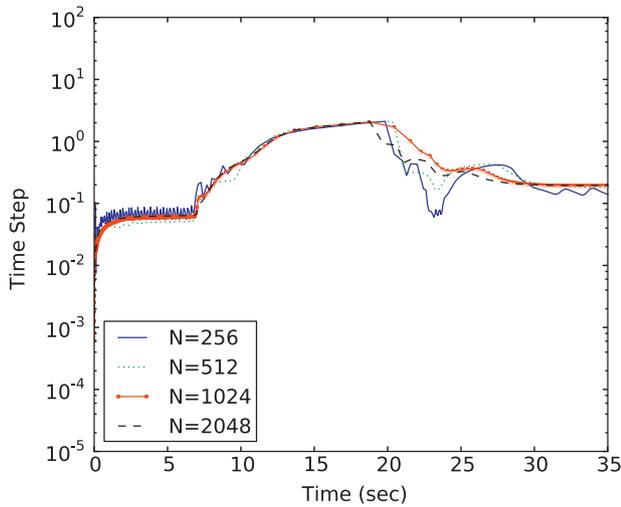


Fig. 3. The effect of the spatial discretization on time adaptivity.

We compute a relative error in the L_2 norm, E_{L_2} , between the approximate solutions obtained, U , and \bar{U}_{1024} .

$$E_{L_2}(U) = \frac{\|\bar{U}_{1024} - U\|_{L_2}}{\|\bar{U}_{1024}\|_{L_2}} \quad (11)$$

Table 1 shows that while some error is incurred, it is on the order of a few percent error and can be reduced by modifying the tolerance, ϵ , discussed later.

4.1. Influence of spatial error on time adaptivity

The error in the spatial discretization has a minor effect on the time adaptivity. We ran the fictitious problem, varying the number of elements $N = \{256, 512, 1024, 2048\}$ used in the finite element discretization until a simulated time of 35.5 s. Note that this time corresponds to the solution shown in Fig. 1g. The time adaptivity error tolerance was held constant for each mesh, $\epsilon = 10^{-3}$. Fig. 3 displays the time step adaptivity over the simulation time for each mesh. We note that for coarse meshes $N = \{256, 512\}$ the time step adaptivity oscillates. As the mesh is refined and the spatial error decreases, the adaptivity curve is smoothed, although the minimum and maximum time step size remains constant for all meshes.

4.2. Influence of the tolerance parameter ϵ on the solution

The time step adaptivity depends on a parameter ϵ , which is an estimated error tolerance above which a time step is rejected. This parameter also controls the evolution of the time step size. In this section we study how the choice of ϵ affects the solution. We again solve the fictitious problem on a series of meshes, $N = \{128, 256, 516\}$, and vary ϵ from 10^{-1} to 10^{-5} by decades and compare the solution at a time of 35.5 s.

Fig. 4 shows that for a large ϵ , the error can be quite large (170%). This is because the time step size is related to the amount of temporal error allowed in each time step, which causes the solution to be poor. Decreasing ϵ causes the time step size to be smaller, allowing less temporal error to remain in the solution. At some point, the time step size becomes over-refined

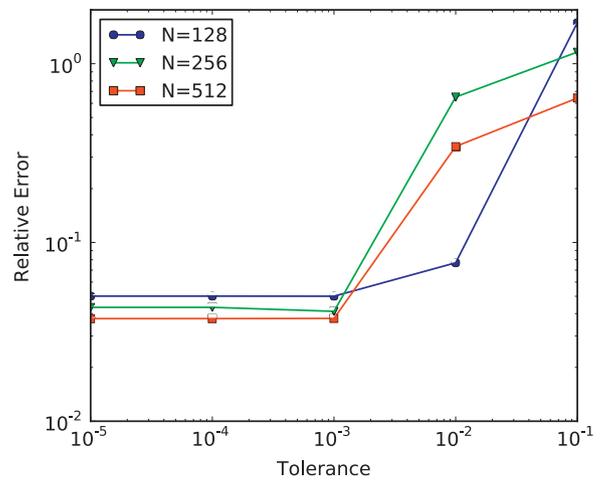


Fig. 4. The effect of ϵ on time adaptivity.

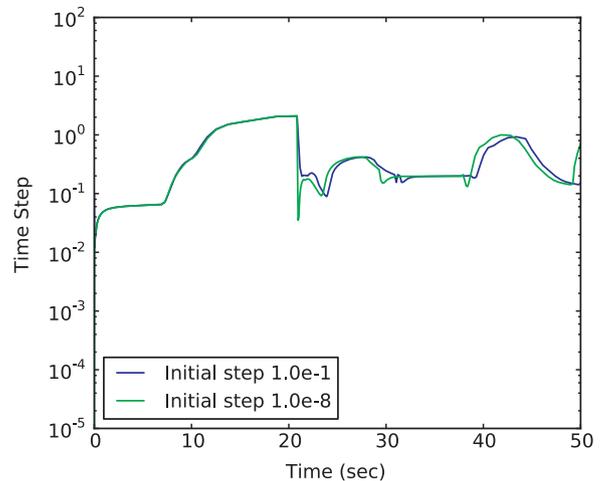


Fig. 5. The time step size adaptivity is not sensitive to the choice of the initial condition.

and the spatial error dominates the overall error in the solution. For this series of problems, this point is $\epsilon = 10^{-3}$. This means that while the choosing of this parameter can have large effects on the solution, there is a sufficiently small threshold that can be chosen after which no change is seen in the approximation.

4.3. Influence of initial time step size on time adaptivity

Time adaptivity is also useful for choosing the initial time step. In the original simulation, shown in Fig. 5, the initial time step chosen was $\Delta t = 0.1$ which initially is 2 orders of magnitude too large. The adaptive step process assists in choosing this initial step in that values which are too large are cut to acceptable sizes. In this case the adaptation reduces the time step to $\Delta t = 3 \times 10^{-3}$. Alternatively an initial time step which is too small, will quickly grow. Fig. 5 shows a comparison between the evolution of two initial step sizes, $\Delta t = 10^{-1}$ and $\Delta t = 10^{-8}$. In both situations, the time-step size evolves over the solution time with good agreement.

5. Conclusions

We briefly described the diffusive wave approximation to the shallow water equations and presented a numerical strategy for their approximation using a Galerkin finite element procedure for spatial discretization and the generalized- α method for temporal discretization. We discussed a new error estimator which can be used to economically enable time adaptivity. This adaptivity proves useful even in simple 1D problems, while being robust. This work is an initial study on solution strategies for the DSW equation for the modeling of 2D overland flows. While nothing in the DSW equation limits its application to 1D, extending a numerical technique into high dimensions is not completely trivial. In future work, we will both extend the numerical technique to 2D as well as incorporate methods for more accurately determining and utilizing accurate frictional coefficients.

References

- [1] C.B. Vreugdenhil, Numerical Methods for Shallow-water Flow, Kluwer Academic Publishers, 1994.
- [2] V.T. Chow, Open-channel Hydraulics, The Blackburn Press, 2009.
- [3] M. Santillana, C. Dawson, A numerical approach to study the properties of solutions of the diffusive wave approximation of the shallow water equations, Journal of Computational Geosciences 14 (1) (2009) 31–53.
- [4] R. Alonso, M. Santillana, C. Dawson, On the diffusive wave approximation of the shallow water equations, Journal of Applied Mathematics 19 (5) (2008) 575–606.
- [5] R.J. Alonso, M. Santillana, C. Dawson, On the diffusive wave approximation of the shallow water equations, Journal of Applied Mathematics 19 (5) (2008) 575–606.
- [6] J. Chung, G.M. Hulbert, A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized- α method, Journal of Applied Mechanics 60 (1993).
- [7] K.E. Jansen, C.H. Whiting, G.M. Hulbert, A generalized- α method for integrating the filtered Navier–Stokes equations with a stabilized finite element method, Computer Methods in Applied Mechanics and Engineering 190 (3–4) (2000) 305–319.
- [8] T.J.R. Hughes, The Finite Element Method, Dover, Mineola, New York, 2000.
- [9] L. Demkowicz, Computing with hp -adaptive finite elements, Chapman & Hall, CRC, 2007.
- [10] Y. Bazilevs, V.M. Calo, J. Cottrell, T.J.R. Hughes, A. Reali, G. Scovazzi, Variational multiscale residual-based turbulence modeling for large eddy simulation of incompressible flows, Computer Methods in Applied Mechanics and Engineering 197 (1–4) (2007) 173–201.
- [11] H. Gomez, V.M. Calo, Y. Bazilevs, T.J.R. Hughes, Isogeometric analysis of the Cahn–Hilliard phase-field model, Computer Methods in Applied Mechanics and Engineering 197 (49–50) (2008) 4333–4352.
- [12] H. Gomez, T.J.R. Hughes, X. Nogueira, V.M. Calo, Isogeometric analysis of the isothermal Navier–Stokes–Korteweg equations, Computer Methods in Applied Mechanics and Engineering 199 (2010) 1828.
- [13] P.J. van der Houwen, B.P. Sommeijer, W. Couzy, Embedded diagonally implicit Runge–Kutta algorithms on parallel computers, Mathematics of Computation 58 (1992).
- [14] K. Gustafsson, Control-theoretic techniques for stepsize selection in implicit Runge–Kutta methods, ACM Transactions on Mathematical Software 20 (1994) 496–517.
- [15] G. Benderskaya, M. Clemens, H.D. Gersem, T. Weiland, Embedded Runge–Kutta methods for field-circuit coupled problems with switching elements, IEEE Transactions on Magnetics 41 (2005) 1612–1615.



Nathan Collier has his PhD in Civil Engineering from the University of South Florida, obtained in 2009. He is currently a postdoctoral researcher in Applied Mathematics and Computational Science and Earth Sciences and Engineering at King Abdullah University of Science and Technology.

His research work revolves around computation, principally as it applies to the finite element method. He is the author of PetscNURBS, a high performance, parallel framework written using PETSc to solve partial differential equations using NURBS.



Hany Radwan is a teaching assistant in the Irrigation and Hydraulic Department at Cairo university in Egypt. He prepared his Master degree from Cairo university in 2007 under the theme of solving the analytical equations for fully/partially penetrating groundwater wells, using Matlab. He prepares his PhD thesis in Cairo university under the theme of developing a rotational scheduling for the integrated improved irrigation system in old lands in Egypt. He participated in the design of dewatering system in several monumental areas in Egypt using Visual Modflow Model. At the end of 2010, he has come to KAUST as a programmer position. His research interests are groundwater, well hydraulics, watershed, and irrigation systems.



Lisandro Dalcin obtained his PhD in Engineering from Universidad Nacional del Litoral (Santa Fe, Argentina) in 2008. His first degree is in electromechanical Engineering from Universidad Tecnológica Nacional (Concepcion del Uruguay, Argentina). In 2010, he joined the National Council for Scientific and Technological Research of Argentina (Consejo Nacional de Investigaciones Científicas y Tecnológicas) as an Assistant Researcher.

He is the author of mpi4py and petsc4py/slepc4py a set of bindings for the MPI standard and PETSc/SLEPc libraries targeting parallel distributed computing with the high-level scripting language Python. He is a semi-regular contributor to the Cython project and also contributes to

the development of the PETSc library. He won the R&D 100 Award in 2009 as part of the PETSc team.



Victor M. Calo is an Assistant Professor, Applied Mathematics and Computational Science and Assistant Professor, Earth Sciences and Engineering at KAUST. He assumed his duties in September 2008. Dr. Calo holds a professional engineer degree in Civil Engineering from the University of Buenos Aires. He received a master in Geomechanics and a doctorate in Civil and Environmental Engineering from Stanford University.

He is a member of the International Association for Computational Mechanics, the U.S. Association for Computational Mechanics, the Argentinean Association for Computational Mechanics, the Society for Industrial and Applied Mathematics, the American Society of Civil Engineers, the American Society of Mechanical Engineers, and the Argentinean Society for Geotechnical Engineering. He is a member of the Computational Mechanics committee of the Engineering Mechanics Institute of the American Society of Civil Engineers. He was an invited editor for three special issues of Computer Methods in Applied Mechanics and Engineering that appeared in 2009 and 2010. Also, He is a member of the editorial board of the Journal of Serbian Society for Computational Mechanics.