Adaptive TVD-RK Discontinuous Galerkin Algorithms for Shallow Water Equations

Thida Pongsanguansin, Khamron Mekchay and Montri Maleewong

Abstract— The adaptive Discontinuous Galerkin (DG) method for solving the one-dimensional conservation equation is presented. In this paper we consider the advection equation, the Burgers' equation, and the shallow water equations. To improve the accuracy of this method, two types of adaptive technique are employed. These are the adaptive polynomial(p-daptive) and the adaptive mesh(h-adaptive). Troubled cells needed to be refined are detected by two types of indicators, which are error and gradient indicators. The present schemes have been improved the accuracy of the solution during time integration process. For smooth solution the accuracy can be improve by the adaptive polynomial criteria, while the accuracy of moving shock, rarefaction and high gradient solution can be improved by the adaptive mesh scheme. High gradient area in the computational domain can be detected efficiently by both presented indicators

Keywords—Adaptive Discontinuous Galerkin method, Advection equation, Burgers' equation, Shallow water equations

I. Introduction

Many real flow problems such as transport flows or shallow water flows, for examples, can be expressed in the form of conservation laws. These systems are usually represented as hyperbolic partial differential equations. There are numerous numerical methods, for instance, the finite difference method (FDM) [1], the finite element method (FEM) [18], the finite volume method (FVM) [11, 16], or recently, the discontinuous Galerkin method (DG) [14, 15] for solving these systems. By the FDM, A. Harten and H. Tal-Ezer [1] presented a family of two-level five-point implicit schemes to solve the one-dimensional systems of hyperbolic conservation laws, which generalized the Crank-Nicholson scheme to the fourth order accuracy in both space and time. For the finite element method, Z. Xu et. al [18] applied the h-adaptive streamline diffusion finite element method with a small mesh-dependent artificial viscosity to solve nonlinear hyperbolic conservation equations. By applying the finite volume method, G. Manzini [11] developed the cell-centered upwind differences to solve the one-dimensional linear conservation laws with stiff reaction source terms. More details of the FVM can be found in Leveque, [16].

Conservation laws is an important class of homogeneous hyperbolic equation. The simplest case is when we have constant coefficients in one dimension, namely a scalar problem, in this case the equation is called the advection equation, the advection equation is an example of equation in conservative form that has various kinds of behaviors. For example, a weak solution can be in the form of a shock wave, namely, a solution that has a sharp gradient. Other scalar problem is the Burgers' equation, this equation has nonlinear flux function and a weak solution can be in the form of a shock for a large time although initial condition is smooth. For the case of system of equations, the shallow water equations are also in conservation forms which can be used to model many problems in real world such as dam break, tsunami, flood, etc. These equations can be derived from the conservation of mass and the conservation of momentum.

The Runge-Kutta Discontinuous Galerkin (RKDG) method has several advantages. Not only it can be used to handle complex geometries, and also adaptivity strategies are easily applied since grid refinement can be done without taking into account the continuity condition that is typically required by most conforming finite element methods. Moreover, the degree of approximating polynomials can be adjusted locally, which allows an efficient polynomial adaptivity in each cell volume with totally independent of its neighbors. Since we want to improve the accuracy of numerical solutions, especially in the locally sharp gradient

This work was supported in part by Science Achivement Scholarship Thailand(SAST) and Center of Advanced Studies in Industrial Technology, Faculty of Engineering, Kasetsart University.

Thida Pongsanguansin is with Department of Mathematics and Computer Science, Faculty of Science, Chulalongkorn University, Thailand, 10330 (t.pongsanguansin@gmail.com).

Khamron Mekchay is with Department of Mathematics and Computer Science, Faculty of Science, Chulalongkorn University, Thailand, 10330 (K.Mekchay@gmail.com).

Montri Maleewong is with Department of Mathematics, Faculty of Science, Kasetsart University, Thailand, 10900 (Montri.M@ku.ac.th).

area in the computational domain. There are two strategies that can overcome these issues, the adaptive polynomial degree (p-adaptive), [10], and the adaptive mesh (h-adaptive), [12].

In this work, we have applied these two concepts and constructed algorithms for the adaptive strategies employing two types of indicators, gradient and error indicators, for detecting troubled cells in the computational domain. After detecting the troubled cells, we trace back to the solution in the previous time to increase the polynomial degree of those cells (p-adaptive) or divided those cells into two smaller sub-cell (h-adaptive) and then evolve numerical solutions to the current time step. This algorithm gives the new numerical solutions that have more accuracy than the old one in the current time step.

The advection equation, the Burgers' equation and the shallow water equations, which are conservation equation, are presented in Section 2, the numerical methodology for the one-dimensional conservation laws, including the algorithms for the adaptive polynomial and adaptive mesh strategies, is presented in Section 3. The numerical experiments and results are illustrated in Section 4, and finally, the conclusions is given in Section 5.

II. Conservation equations

The one-dimensional hyperbolic equation of conservation laws represented in the time dependent problem takes the form

$$\frac{\partial}{\partial t}U(x,t)+\frac{\partial}{\partial x}F(U(x,t))=0, \tag{1}$$

where U is an m-dimensional vector of conserved quantities (or state variables), such as mass, momentum, or energy, in a fluid dynamic problem, $F(U)$ is an m-dimensional vector called the flux function.

In this work, we consider three forms of (1) which are the advection equation (m=1), the Burgurs' equation (m=1) and the shallow water equations (m=2).

A. *The advection equation*

The one-dimensional advection equation is

$$\frac{\partial u}{\partial t}+c\frac{\partial u}{\partial x}=0, \tag{2}$$

where $x\in(0,L)$, and $t\in(0,T)$. Here $U=u(x,t)$, $F(U)=cu$, and c is constant.

B. *The Burgers' equation*

The inviscid Burgers' equation is expressed in the form

$$\frac{\partial u}{\partial t}+\frac{\partial u^2}{\partial x}=0, \tag{3}$$

where $U=u(x,t)$ and $F(U)=u^2$.

C. *The shallow water equations*

The one-dimensional shallow water equations are in the form

$$\frac{\partial U}{\partial t}+\frac{\partial F(U)}{\partial x}=S(U), \tag{4}$$

where

$$U=\begin{pmatrix} h\\ q\end{pmatrix},\ \text{and}\ F(U)=\begin{pmatrix} q\\ \frac{q^2}{h}+\frac{gh^2}{2}\end{pmatrix} \tag{5}$$

are the vector of conserved variables and the flux vector in the x direction, respectively. For the conserved variables, h is the water depth, u is the average flow velocity in the x-direction, $q=uh$ is the discharge, and g is the acceleration due to gravity.

The right hand side of the system (4) represents source term, given by

$$S(U)=\begin{pmatrix} 0\\ gh(S_0-S_f)\end{pmatrix}, \tag{6}$$

which contains the effect of the bed slope S_0, and the bed friction S_f. The term S_f can be estimated by an empirical formula as

$$S_f=\frac{n^2q|q|}{h^{10/3}}, \tag{7}$$

where n is the Manning roughness coefficient.

In this work, we neglect the effect of source term, so $S(U)=0$. We then investigate the accuracy of the adaptive DG method for solving the shallow water equations without source term treatment.

III. Methodology

In this section, we presented the Discontinuous Galerkin method and adaptive algorithms for solving the one-dimensional conservation equation.

A. *Discontinuous Galerkin method for the one-dimensional conservation equation*

We first partition the domain $(0,L)$ into K subintervals, and denoted the j-th cell by $I_j=\left[x_{j-1/2},x_{j+1/2}\right]$, $j=1,\dots K$, with grid size $\Delta_j=x_{j+\frac{1}{2}}-x_{j-\frac{1}{2}}$, and cell center $x_j=\left(x_{j+\frac{1}{2}}+x_{j-\frac{1}{2}}\right)/2$, $x_{j+\frac{1}{2}}$ and $x_{j-\frac{1}{2}}$ are the left and right boundaries of the considering cell respectively.

For the advection and Burgers' equations, we approximate the solution u by u_h in the finite dimensional space V_h^N. For in the shallow water equations, we approximate the solution U by $U_h=\begin{pmatrix} h_h\\ q_h\end{pmatrix}$ in the finite dimensional space, $V_h^N\times V_h^N$, where V_h^N defined by

$$V_h^N = \left\{ v \in L^1 (0,1) : v|_{I_j} \in P^N (I_j), j = 1, \ldots K \right\}, \tag{8}$$

where $P^N (I_j)$ denotes the space of polynomials of degree at most N on I_j.

Multiplying (1) by a test function $v_h (x) \in P^N (I_j)$ and using the integration by parts over I_j, we obtain a weak form,

$$\int_{I_j} (\partial_t U_h) v_h dx - \int_{I_j} F (U_h) \partial_x v_h dx$$
$$+ [F (U_h) v_h]_{j+\frac{1}{2}} - [F (U_h) v_h]_{j-\frac{1}{2}} = 0. \tag{9}$$

The flux function F can be approximated using numerical flux $\hat{F}$ that depends on two values of U_h at the interfaces $x_{j\pm\frac{1}{2}}$ by

$$\hat{F}_{j\pm\frac{1}{2}} = \hat{F} \left(U_h|_{j\pm\frac{1}{2}}^-, U_h|_{j\pm\frac{1}{2}}^+ \right) \tag{10}$$

where $U_h|_{j\pm\frac{1}{2}}^-$ and $U_h|_{j\pm\frac{1}{2}}^+$ are the approximate solutions from the left and right of the boundaries $j \pm \frac{1}{2}$ of the j-th cell, respectively.

By choosing the Legendre polynomials as the local basis functions, the approximate solution U_h can be written as

$$U_h (x,t) = \sum_{m=0}^{N} U_j^m (t) \varphi_m (x) , \tag{11}$$

where $U_j^m (t)$ is the coefficient function of t, and $\varphi_m (x)$ is the Legendre polynomial defined by

$$\varphi_m (x) = P_m \left(\frac{2 (x - x_j)}{\Delta_j} \right) . \tag{12}$$

As in the standard Galerkin method, we choose the test functions $v_h (x)$ to be the same as the basis functions, i.e., $v_h (x) = \{\varphi_l (x)\}_{l=0}^N$. Some important properties of the Legendre's polynomial are

$$\int\limits_{-1}^{1} P_m (\xi) P_l (\xi) d\xi = \frac{2}{2l+1} \delta_{ml} , \tag{13}$$

where

$$\xi = \frac{2 (x - x_j)}{\Delta_j} , \quad \delta_{ml} = \begin{cases} 1, & m = l \\ 0, & m \neq l \end{cases} , \tag{14}$$

and

$$P_l (-1) = (-1)^l \ , \ P_l (1) = 1 \, . \tag{15}$$

The weak form (9) is then simplified to

$$\frac{dU_j^l (t)}{dt} = \frac{2l+1}{\Delta_j} \int_{I_j} F (U_h) \partial_x \varphi_l (x) dx +$$
$$\frac{2l+1}{\Delta_j} \left\{ (-1)^l \hat{F}_{j-\frac{1}{2}} - \hat{F}_{j+\frac{1}{2}} \right\} , \tag{16}$$

where $j = 1, \ldots, K$.

B. Numerical flux

The system of ODEs in equation (16) can be integrated in time by any standard integrators. In our scheme, we apply the Runge-Kutta method. Detail derivation will be given in the next subsection. The accuracy of approximate solutions is depend directly on the method for approximating fluxes at interfaces, $\hat{F}_{j\pm\frac{1}{2}}$, called numerical fluxes. we apply two approximations which are LLF and HLL. Details are provided as follows.

1. *The local Lax-Friedrichs flux (LLF flux)*
   We employ the local Lax-Friedrichs flux for the advection equation and the Burgers' equation, this flux is represented by

$$\hat{F}^{LLF} (U_L, U_R) = \frac{1}{2} [F (U_L) + F (U_R) - C (U_R - U_L)] \ , \tag{17}$$

   where $F_L = F (U_L)$, $F_R = F (U_R)$, and $C = \max \left| F^{'} (s) \right|$, and $\min (U_L, U_R) \leq s \leq \max (U_L, U_R)$.

2. The Harten-Lax-van Leer flux (HLL flux)
   For the shallow water equations, Toro [8, 9] presented a suitable HLL(Harten-Lax-van Leer)-type flux [7, 8, 9, 15] based on the work by Harten et. al.[1],

$$\hat{F}^{HLL} (U_L, U_R) = \begin{cases} F_L & if \; 0 \leq S_L; \\ F^* & if \; S_L \leq 0 \leq S_R; \\ F_R & if \; 0 \geq S_R, \end{cases} \tag{18}$$

   where F^* is given by

$$F^* = \frac{S_R F_L - S_L F_R + S_L S_R (U_R - U_L)}{S_R - S_L} . \tag{19}$$

   The wave speeds are chosen under the assumption of two-rarefaction waves in the star region,

$$S_L = \min \left(u_L - \sqrt{g h_L}, u^* - \sqrt{g h^*} \right) , \tag{20}$$

$$S_R = \min \left(u_R + \sqrt{g h_R}, u^* + \sqrt{g h^*} \right) , \tag{21}$$

with

$$\sqrt{gh^*} = \frac{1}{2}\left(\sqrt{gh_L} + \sqrt{gh_R}\right) - \frac{1}{4}\left(u_R - u_L\right), \tag{22}$$

$$u^* = \frac{1}{2}\left(u_L + u_R\right) + \sqrt{gh_L} - \sqrt{gh_R}. \tag{23}$$

The expressions for the wave speeds are obtained by assuming the wet bed condition, i.e. positive flow depth h on both sides of the computational domain. For the dry bed case, the wave speeds are approximated by

$$S_L = u_L - \sqrt{gh_L}, \tag{24}$$

and

$$S_R = u_L + 2\sqrt{gh_L}. \tag{25}$$

C. *Total Variation Diminishing Runge Kutta (TVD-RK)*

After discretizing (4) in spatial space by the DG method, we obtain a system of ODEs corresponding to (16). The system can be rewritten in the form of

$$\frac{dU_h(t)}{dt} = L_h\left(U_h, t\right), \ \forall t \in (0, T), \tag{26}$$

with initial condition

$$U_h(x, 0) = U_{0h}, \tag{27}$$

where $L_h(U_h, t)$ represents the right hand side of equation (16).

Time discretization process is performed by the high-order TVD Runge Kutta method which was introduced previously by Chi-Wang Shu [6]. Noted that when the polynomial degree N is used, a TVD version of Runge Kutta method (TVD-RK) at least of order $N+1$ must be employed.

Let $\{t^n\}_{n=0}^{M}$ be a partition of $[0,T]$ in M intervals, and $\Delta t^n = t^{n+1} - t^n$, for $n = 0, \ldots, M-1$. The time marching algorithm can be summarized as follows.

1. Set $U_h^0 = U_{0h}$,
2. For $n = 0, .., M-1$ compute U_h^{n+1} from U_h^n as follows:
3. set $d^{(0)} = U_h^n$,
4. for $i = 1, ..., k+1$ compute the intermediate functions:

$$d^{(i)} = \left\{\sum_{s=0}^{i-1}\left(\alpha_{is} d^{(s)} + \beta_{is}\triangle t^n L_h\left(d^{(s)}, t\right)\right)\right\},$$

5. set $U_h^{n+1} = d^{(k+1)}$,

where α_{is} and β_{is} are parameters depend on the order of TVD-RK.

For example, the TVD-RK of orders 2 and 3 are given by

TVD-RK order 2

$$d^{(1)} = U_h^n + L_h\left(U_h^n, t^n\right), \tag{28}$$

$$U_h^{n+1} = d^{(2)} = \frac{1}{2}\left(U_h^n + d^{(1)} + \triangle t L_h\left(d^{(1)}, t^n + \triangle t\right)\right), \tag{29}$$

TVD-RK order 3

$$d^{(1)} = U_h^n + \Delta t L_h\left(U_h^n, t^n\right), \tag{30}$$

$$d^{(2)} = \frac{1}{4}\left(3U_h^n + d^{(1)} + \Delta t L_h\left(d^{(1)}, t^n + \Delta t\right)\right), \tag{31}$$

$$U_h^{n+1} = \frac{1}{3}\left(U_h^n + 2d^{(2)} + 2\Delta t L_h\left(d^{(2)}, t^n + \frac{1}{2}\Delta t\right)\right). \tag{32}$$

By these setting, the TVD-RK has some useful stability properties. More details can be seen in [2, 14].

D. *The MUSCL slope limiter*

When high-order polynomials are used for approximating the solution, numerical method may produce some unphysical oscillations [2, 7, 14, 15, 16]. A slope limiter concept can be applied on every computational cell to avoid these undesired oscillations. For instance, in the case of piecewise linear appoximation, the slope limiter of $U_h|_{I_j}$ denoted by $\Lambda\Pi_h^1\left(U_h|_{I_j}\right)$ is expressed as

$$\Lambda\Pi_h^1\left(U_h|_{I_j}\right) = \overline{U}_j + (x - x_j) \quad m\left(U_{x,j}, \frac{\overline{U}_{j+1} - \overline{U}_j}{\Delta_j}, \frac{\overline{U}_j - \overline{U}_{j-1}}{\Delta_j}\right), \tag{33}$$

for $j = 1, \ldots, K$.

In (33), $\bar{U}_j$ is the mean value in the j-th cell, $U_{x,j}$ is the slope of solution in the j-th cell, and m is the *minmod* function defined by

$$m\left(a_1,\ldots,a_n\right)\begin{cases} sign\left(a_1\right)\min_{1\le i\le n}\left|a_i\right|, \\ \text{ if } sign\left(a_1\right)=\ldots=sign\left(a_n\right)\ , \\ \\ 0, \text{otherwise.}\end{cases} \tag{34}$$

This is the well-known slope limiter of the MUSCL schemes introduced by vanLeer[4, 5].

In the case that the aprroximate solution is a polynomial degree $N \geq 2$, that is,

$$U_h|_{I_j}\left(x,t\right)=\sum_{l=0}^{N}U_j^l\left(t\right)\varphi_l\left(x\right)\ .$$

The P^1-part of u_h denoted by U_h^1 defined as, see [2],

$$U_h^1|_{I_j}\left(x,t\right)=\sum_{l=0}^{1}U_j^l\left(t\right)\varphi_l\left(x\right)\ .$$

The slope limiter procedure in this case denoted by $\Lambda\Pi_h^N$, is summarized as follows:

(1) Compute $\tilde{U}^-_{j+\frac{1}{2}}$ and $\tilde{U}^+_{j-\frac{1}{2}}$ from

$$\tilde{U}^-_{j+\frac{1}{2}}=\bar{U}_j+m\left(U^-_{j+\frac{1}{2}}-\bar{U}_j,\bar{U}_j-\bar{U}_{j-1},\bar{U}_{j+1}-\bar{U}_j\right)\ , \tag{35}$$

$$\tilde{U}^+_{j-\frac{1}{2}}=\bar{U}_j-m\left(\bar{U}_j-U^+_{j-\frac{1}{2}},\bar{U}_j-\bar{U}_{j-1},\bar{U}_{j+1}-\bar{U}_j\right)\ . \tag{36}$$

(2) If $\tilde{U}^-_{j+\frac{1}{2}} = U^-_{j+\frac{1}{2}}$ and $\tilde{U}^+_{j-\frac{1}{2}} = U^+_{j-\frac{1}{2}}$ set $\Lambda\Pi_h^N\left(U_h|_{I_j}\right)=U_h|_{I_j}$.

(3) Otherwise, take $U_h|_{I_j}$ equal to $\Lambda\Pi_h^1\left(U_h^1|_{I_j}\right)$.

When the slope limiter has been applied at each intermediate computation of the Runge-Kutta method, the intermediate function in this case becomes

$$d^{(i)}=\Lambda\Pi_h\left\{\sum_{s=0}^{i-1}\left(\alpha_{is}d^{(s)}+\beta_{is}\triangle t^n L_h\left(d^{(s)},t\right)\right)\right\}.$$

The parameters α_{is} and β_{is} depend on the choice of TVD-RK order.

E. Adaptive Discontinuous Galerkin method

In general, the accuracy of the approximate solution can be improved by increasing the degrees of polynomials of local basis functions in each cell or refining grid cells with smaller mesh size in the computational domain. But for computational efficiency, it should be adapted only on troubled cells where the solutions have large error, sharp gradients, or discontinuities.

In this section, we present two types of adaptive algorithms which are adaptive polynomial and adaptive mesh. The adaptive polynomial is successfully applied for solving the case of smooth flow by increasing the order of polynomial basis while the adaptive mesh should be used to handle the case of high gradients or discontinuities. To detect which cell needed to be applied the adaptive criteria, we need some indicators to detect those troubled cells. In this work, we employ two types of indicators which are error indicator (assuming we know the exact solution) and gradient indicator. The error indicator is introduced for checking the performance of the gradient indicator. Since most of problems have no exact solution, so the gradient indicator can be applied in practical case.

E.1 Indicators

The values of the indicators for the j-th cell at time t^n, denoted by ε_j^n, are defined as follows:

(1) Error indicator

$$\varepsilon_j^n=\sqrt{\frac{\sum_{n_g}\left(q^n_{exact}-q^n_{approx}\right)^2}{n_g}}, \tag{37}$$

where q^n_{exact} is the exact solution, q^n_{approx} is an approximate solution, and n_g is the number of nodes on the j-th cell at time t^n.

(2) Gradient indicator

$$\varepsilon^n_{j,1}=\frac{\left|q^n_{approx}|_{j+\frac{1}{2}}-q^n_{0approx}|_j\right|}{\Delta_j/2}, \tag{38}$$

$$\varepsilon^n_{j,2}=\frac{\left|q^n_{approx}|_{j-\frac{1}{2}}-q^n_{0approx}|_j\right|}{\Delta_j/2}, \tag{39}$$

where $q^n_{approx}|_{j-\frac{1}{2}}$ and $q^n_{approx}|_{j+\frac{1}{2}}$ are approximate solutions for the j-th cell at the left and right boundaries respectively, and $q^n_{0approx}|_j$ is approximate solution at the center of the j-th cell.

E.2 Adaptive polynomial for RKDG method (p-adaptive)

Algorithm for the adaptive polynomial

Given:
A control number for increasing the degree of polynomial, θ_1.
A control number for decreasing the degree of polynomial, θ_2.
The initial degree of polynomial, $mindeg$.
The maximum degree of polynomial, $maxdeg$.
The final time, T.
A partition of time domain $[0,T]$, $\{t^n\}_{n=0}^N$.
Initial condition, u^0 at time step t^0.
The number of cells, K.

The algorithm is summarized below:

Step 1.

1. Given a uniform partition of the domain.
2. Compute approximate solution u^1 at time step t^1.
3. For each cell $j = 1, 2, ..., K$,
   - set the initial degree of polynomial $deg_j^0 = mindeg$.

Step 2. Define the degree of polynomial in the j-th cell at time t^n by deg_j^n,

1. Compute the indicator ε_j^n.
2. Set $max\varepsilon = \max_{1 \le j \le K} \left\{\varepsilon_j^n\right\}$. For each cell $j = 1, 2, ..., K$,
   - error indicator case: If $\varepsilon_j^n > \theta_1 max\varepsilon$, mark this j-th cell as a troubled cell.
   - gradient indicator case: If $\varepsilon_{j,1}^n > \theta_1 max\varepsilon$ or $\varepsilon_{j,2}^n > \theta_1 max\varepsilon$, mark this j-th cell as a trouble cell
3. Increase the degree of polynomial of the mark cell.
   - If $deg_j^{n-1} = maxdeg$, do nothing.
   - If $deg_j^{n-1} < maxdeg$, set the new degree in that troubled cell as $deg_j^{n-1}|_{new} = deg_j^{n-1} + 1$.
4. For the current troubled cell at time t^n, $n \ge 1$
   - error indicator case: If $\varepsilon_j^n < \theta_2 max\varepsilon$, release this troubled cell as a usual cell.
   - gradient indicator case: If $\varepsilon_{j,1}^n < \theta_2 max\varepsilon$ and $\varepsilon_{j,2}^n < \theta_2 max\varepsilon$ release this troubled cell as a usual cell.
5. Decrease the degree of polynomial of usual cells.
   - If the $deg_j^{n-1} = mindeg$, do nothing.
   - If $deg_j^{n-1} > mindeg$, set the new degree as $deg_j^{n-1}|_{new} = deg_j^{n-1} - 1$.

In this step, we have a new degree of polynomial for each cell at time t^{n-1}.

Step 3. Using L^2-projection, project the temporal coefficients $\left\{u_j^l\left(t^{n-1}\right)\right\}_{l=0}^{deg_j^{n-1}}$ to the new set of $\left\{u_j^l\left(t^{n-1}\right)\right\}_{l=0}^{deg_j^{n-1}}|_{new}$.

Step 4. Evolve numerical solutions of all cells from t^{n-1} to t^n.

Step 5. If $n < N$, repeat the whole steps by going back to step 2.

E.3 Adaptive mesh for RKDG method (h-adaptive)

Since for shock or discontinuous solution, the high-order of polynomial approximation cannot improve the accuracy of numerical solution, the adaptive mesh criteria can be employed to increase the accuracy of solution in this case.

Algorithm for adaptive mesh

Given:
A control number used to divide troubled cell into two smaller cells, θ_1.
A control number for merging untroubled cell into one cell, θ_2.
The degree of polynomial, N.
The maximum level of mesh partition, $maxlev$.
The final time, T.
A partition of time domain $[0, T]$, $\{t^n\}_{n=0}^N$.
Initial condition, u^0 at time step t^0.
The initial number of cell K.

The algorithm is summarized below:

Step 1.

1. Given a uniform partition of the domain.
2. Compute approximate solution u^1 at time step t^1.
3. For each cell $j = 1, 2, ..., K$,
   - set the initial mesh level of every cell as $lev_j^0 = 0$..

Step 2. Define the mesh partition at time t^n and the mesh level for the j-th cell by $\{l^n\}$ and lev_j^n, respectively.

1. Compute the indicator ε_j^n.
2. Set $max\varepsilon = \max_{1 \le j \le K} \left\{\varepsilon_j^n\right\}$. For each cell $j = 1, 2, ..., K$,
   - error indicator case: If $\varepsilon_j^n > \theta_1 max\varepsilon$, mark this j-the cell as a troubled cell.
   - gradient indicator case: If $\varepsilon_{j,1}^n > \theta_1 max\varepsilon$ or $\varepsilon_{j,2}^n > \theta_1 max\varepsilon$, mark this j-th cell as a trouble cell
3. Divide mark cells into two sub-cells.
   - If the $lev_j^{n-1} = maxlev$, do nothing.
   - If $lev_j^{n-1} < maxlev$, set new level as $lev_j^{n-1}|_{new} = lev_j^{n-1} + 1$, and divide it into two sub-cells.
4. For the current troubled cell at time t^n, $n \ge 1$

- error indicator case: If $\varepsilon_j^n < \theta_2 max\varepsilon$, release this troubled cell as a usual cell.
- gradient indicator case: If $\varepsilon_{j,1}^n < \theta_2 max\varepsilon$ and $\varepsilon_{j,2}^n < \theta_2 max\varepsilon$ release this troubled cell as a usual cell.

5. Merge two usual cells into one cell.

- If $lev_j^{n-1} = 0$, do nothing.
- If $lev_j^{n-1} > 0$, set the new level to be $lev_j^{n-1}|_{new} = lev_j^{n-1} - 1$ and merge those two sub-cells into one cell (merge cells that come from the same primary cell).

  This step provides the new mesh level for each cell at time t^{n-1}.

Step 3. Using L^2-projection, project the temporal coefficients $\left\{u_j^l\left(t^{n-1}\right)\right\}_{l=0}^{N}$ from the mesh partition$\left\{l^{n-1}\right\}$ to the new mesh partition $\left\{l^{n-1}\right\}|_{new}$.

Step 4. Evolve numerical solutions for every cell from t^{n-1} to t^n.

Step 5. If $n < N$, repeat the whole steps by going back to step 2.

IV. Numerical results

The numerical results obtained by the adaptive Discontinuous Galerkin method with two indicators are presented in this section.

A. *The advection equation*

A.1 *Adaptive polynomial RKDG method*

We consider the pure advection equation with initially smooth condition

$$u_t + 3u_x = 0, \qquad (\text{-10,10}) \times (0,1)\,, \tag{40}$$

$$u\,(x,0) = e^{-(x-3)^2}. \tag{41}$$

We will show some results by varying the values of θ_1 and θ_2, and considering different initial degree of polynomial, maximum degree of polynomial, and the number of cells.

We denote P^N as the numerical solutions obtained from the RKDG method without adaptive criteria with polynomial degree N, and P^l/P^h refers to the numerical results obtained from the adaptive polynomial RKDG with initial degree l and the maximum degree h.

We show the accuracy of numerical solutions by the adaptive polynomial RKDG method using error indicator and gradient indicator in Figure 1, where the initial degree is 1 and the maximum degree is 3. Here, θ_1 and θ_2 are 0.025 and 0.01 respectively. It is found that, the numerical solution (circle) is in good agreement with the exact solution (solid line) including around the peak area when using error indicator, but the profile is in good agreement with the exact solution except around the peak area when using gradient indicator. However, if we want to improve the solution accuracy in this area, we have to set the values of θ_1 and θ_2 to be sufficiently small in order to detect more troubled cells near the peak because gradient is quite small in this area. The degrees of polynomial for each cell in the time domain of Figure 1 is shown in Figure 2. For each time step, the indicators can detect the troubled cells which are moving with the same speed as of the solution profile. The troubled cells are in the high gradient areas. The color bar shows the values of degrees of polynomial. The polynomial degree increases from one to three for the troubled cells. Also, the troubled cells are released to usual cells when the solution moves to the right such that polynomial degree decreases from three to one.

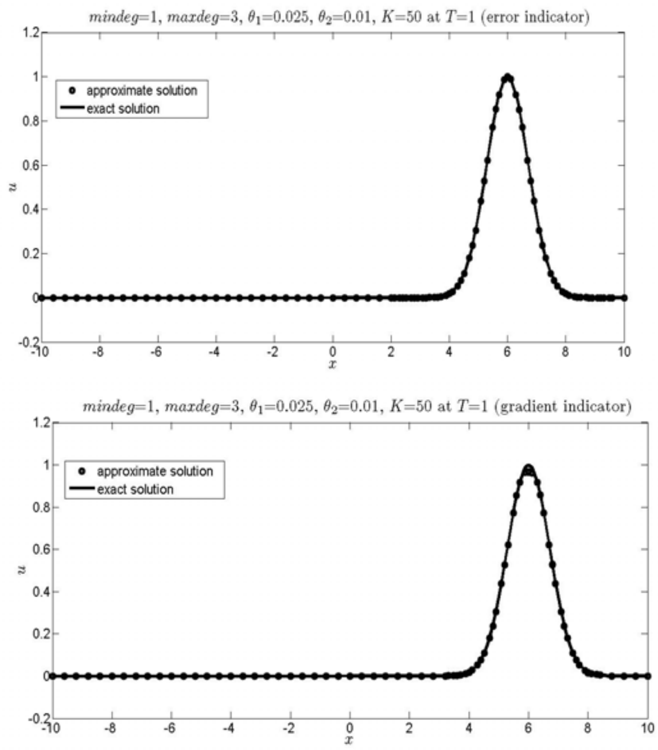

Figure 1: The comparison of the exact solution and the numerical solution at the final time $T = 1$ using error indicator(top) and gradient indicator(bottom).

The RMS errors obtained by the adaptive polynomial RKDG method when using error indicator and gradient indicator are shown in Tables I and II respectively.

It can be seen from Table I and II that, when the values of θ_1 and θ_2 are fixed and without adaptive polynomial cri-

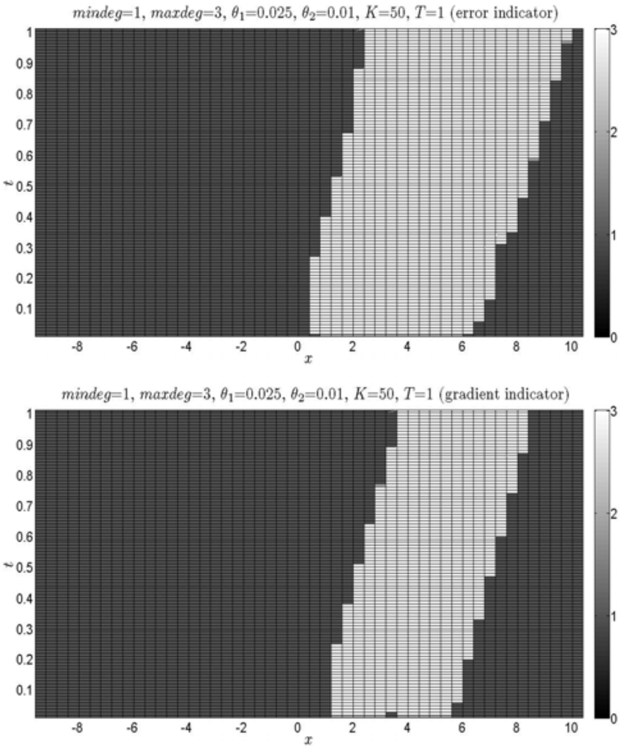

Figure 2: The $x-t$ plot shows adaptive polynomial method from degree 1 to degree 3 when using error indicator(top) and gradient indicator(bottom).

Table I: The RMS errors using error indicator with $K = 100$ cells and for some values of θ_1 and θ_2.

(θ_1, θ_2)	$(0.1, 0.05)$	$(0.025, 0.01)$	$(0.01, 0)$
P^1	1.6468e-003	1.6468e-003	1.6468e-003
P^2	7.6041e-005	7.6041e-005	7.6041e-005
P^3	2.5861e-006	2.5861e-006	2.5861e-006
P^1/P^2	1.3456e-004	8.9834e-005	8.3948e-005
P^1/P^3	2.8599e-004	2.8096e-005	9.8543e-006
P^2/P^3	1.6238e-005	2.8713e-006	2.7624e-006

Table II: The RMS errors when using gradient indicator with $K = 100$ cells for some values of θ_1 and θ_2.

(θ_1, θ_2)	$(0.1, 0.05)$	$(0.025, 0.01)$	$(0.01, 0)$
P^1	1.6468e-003	1.6468e-003	1.6468e-003
P^2	7.6041e-005	7.6041e-005	7.6041e-005
P^3	2.5861e-006	2.5861e-006	2.5861e-006
P^1/P^2	3.1666e-004	1.1455e-004	9.0263e-005
P^1/P^3	3.5018e-004	8.1399e-005	3.4526e-005
P^2/P^3	1.0345e-005	4.2764e-006	3.1197e-006

teria, the results obtained by the higher degree polynomial are more accurate than the results calculated by the lower degrees of polynomial. These results correspond with the theoretical results in [2]. The RMS error of P^2 is comparable with P^1/P^2. They are in the same order, however the P^2 method gives more slightly accurate results. Likewise, the results from the P^3 and the P^2/P^3 methods are comparable. The P^3 method provides slightly better results. The RMS errors of P^1/P^3 are less than P^1 but are in the same order of P^3 when using small values of θ_1 and θ_2. These results show that the computational cells have been adapted from degree one to degree three until the numerical solution has the RMS error in the same order as the maximum degree of polynomial applied. Numerial results by other adaptive degrees of polynomial can be concluded similarly.

A.2 Adaptive mesh RKDG method

For the initial condition with discontinuity, the accuracy of numerical solution is not improved by increasing the order of polynomial. But the accuracy can be improved by increasing the number of cells. Hence, adaptive mesh refinement is suitable in this case.

We consider the pure advection equation with initially discontinuous profile

$$u_t + 3u_x = 0, \qquad \text{(-10,10)} \times (0,1)\,, \tag{42}$$

$$u\,(x,0) = \begin{cases} 1, & \text{if } x \geqslant 2 \\ 2, & \text{if } x < 2\,. \end{cases} \tag{43}$$

Some numerical results by error indicator when fixing the degree of polynomial are shown in Figure 3. We have varied four different values of $maxlev$ from 1 to 4 in the adaptive mesh algorithm. Thus, $maxlev = 4$ corresponds to the smallest divided cell from the primary cell, and $maxlev = 0$ meaning initial mesh partition size.

In Figure 3, We have set $(\theta_1, \theta_2) = (0.025, 0.01)$. It is shown that the sharp-front can be captured accurately when setting $maxlev = 4$ because more cells have been detected as troubled cells and these cell are divided into smaller subcells. The smallest mesh spacing occurs at $maxlev = 4$. Similar arguments can be made when the gradient indicator is applied.

When we use polynomial degree 1 as a basis function, the results of various values of θ_1 and θ_2 when using error and gradient indicators are shown in Tables III and IV, respectively. The notation P^N_{maxlev} refers to the results of using polynomial degree N as a basis with $maxlev$ in the adaptive mesh method.

It can be seen from Tables III and IV that the most accurate result can be obtained by using $maxlev$=4, and the RMS errors are not directly effected by the values of θ_1 and θ_2 because the troubled cells appear only around the

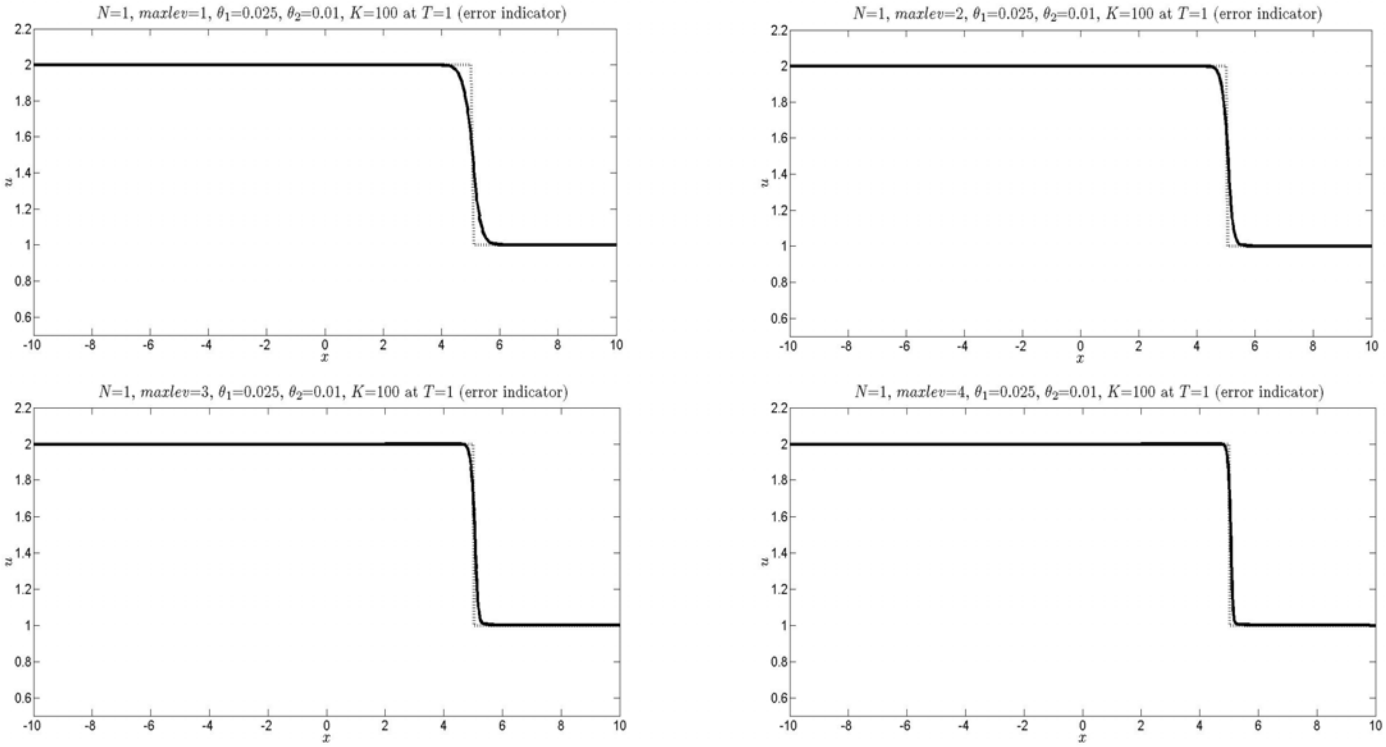

Figure 3: The comparisons between the exact solution (dot line) and the numerical solution (solid line) at time $T = 1$, using error indicator for four cases of $maxlev$: $maxlev = 1$, $maxlev = 2$, $maxlev = 3$, and $maxlev = 4$.

Table III: The RMS error when using error indicator with K=100, N=1 for some values of θ_1 and θ_2.

(θ_1, θ_2)	$(0.1, 0.05)$	$(0.025, 0.01)$	$(0.01, 0)$
P^1	0.0647	0.0647	0.0647
P^1_1	0.0524	0.0524	0.0524
P^1_2	0.0435	0.0441	0.0442
P^1_3	0.0381	0.0393	0.0395
P^1_4	0.0362	0.0378	0.0381

Table IV: The RMS error when using gradient indicator with K=100, N=1 for some values of θ_1 and θ_2.

(θ_1, θ_2)	$(0.1, 0.05)$	$(0.025, 0.01)$	$(0.01, 0)$
P^1	0.0647	0.0647	0.0647
P^1_1	0.0523	0.0524	0.0524
P^1_2	0.0426	0.0435	0.0439
P^1_3	0.0375	0.0368	0.0385
P^1_4	0.0658	0.0300	0.0353

Table V: The RMS error when using error indicator with K=100, N=2 for some values of θ_1 and θ_2.

(θ_1, θ_2)	$(0.1, 0.05)$	$(0.025, 0.01)$	$(0.01, 0)$
P^2	0.0492	0.0492	0.0492
P^2_1	0.0399	0.0384	0.0404
P^2_2	0.0294	0.0294	0.0310
P^2_3	0.0231	0.0227	0.0233
P^2_4	0.0191	0.0204	0.0204

shock area and the number of troubled cells are not different for each pair of θ_1 and θ_2. However, for the gradient indicator case, and $(\theta_1, \theta_2) = (0.1, 0.05)$, these values are relatively large so that the trouble cells cannot be detected correctly like the case of $(\theta_1, \theta_2) = (0.025, 0.01)$. The $(x - t)$ plot showing the adaptive mesh area is presented in Figure 4. The troubled cell zone for the case of $(\theta_1, \theta_2) = (0.1, 0.05)$ is smaller when comparing with the case of $(\theta_1, \theta_2) = (0.025, 0.01)$. The results when increasing the polynomial degree to $N = 2$ are shown in Tables V and VI. The RMS error decreases as the $maxlev$ increases such that $maxlev$ = 4 provides the smallest RMS error. This case is the pure advection case with initially discontinuous profile, these observations reveal that increasing the order of polynomial can improve slightly numerical accuracy but it is not significant.

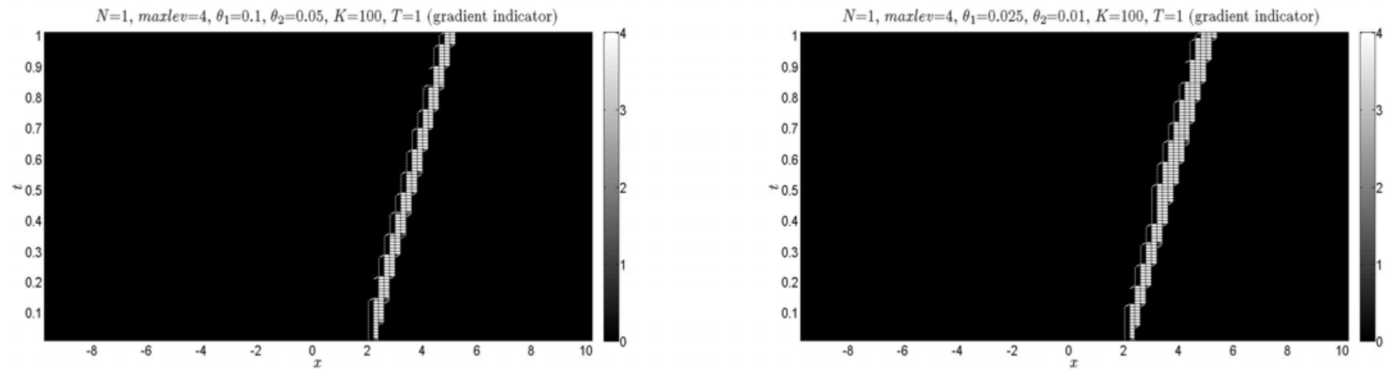

Figure 4: The adaptive area for $N = 1$, $K = 100$, $(\theta_1, \theta_2) = (0.1, 0.05)$ (left) and $(\theta_1, \theta_2) = (0.025, 0.01)$ (right) with gradient indicator.

Table VI: The RMS error when using gradient indicator with $K = 100$ cells, polynomial degree $N = 2$ for some values of θ_1 and θ_2.

(θ_1, θ_2)	$(0.1, 0.05)$	$(0.025, 0.01)$	$(0.01, 0)$
P^2	0.0492	0.0492	0.0492
P^2_1	0.0415	0.0419	0.0411
P^2_2	0.0351	0.0319	0.0307
P^2_3	0.0388	0.0258	0.0231
P^2_4	0.0549	0.0245	0.0199

B. The Burgers' equation

We consider the Burgers' equation with initial condition

$$u(x, 0) = \sin(x), \qquad 0 < x < \pi. \qquad (44)$$

B.1 Adaptive polynomial RKDG method

In this section, we fix the number of cell K where as the polynomial degree is adapted automatically following the results from indicators. The comparisons of numerical solutions with the exact solution by two types of indicators are shown in Figure 5. The polynomial degree increase from 1 ($mindeg$) to 3 ($maxdeg$). The numerical results with increasing order of polynomial basis are in good agreement with the exact solutions almost the whole domain except at the area that sharp gradient is forming. It shows that increasing just the polynomial degree is not enough to improve the accuracy of numerical solution in case of high gradient. Instead, the adaptive mesh method can be appiled in this case. Some results will be shown in the next section.

The behaviors of detecting troubled cells by the error and gradient indicators are shown in Figure 6. They are relatively different in the way of increasing or decreasing the polynomial degrees.

Table VII: The RMS errors when using error indicator with $K = 50$ and $K = 100$ cells for $\theta_1 = 0.85$ and $\theta_2 = 0.5$.

	$K = 50$	$K = 100$
P^1	0.0579	0.0325
P^2	0.0495	0.0276
P^3	0.0437	0.0244
P^1/P^2	0.0493	0.0276
P^1/P^3	0.0436	0.0243
P^2/P^3	0.0437	0.0244

The solution accuracy measured in terms of the RMS error is shown in Tables VII and VIII. In the case of without adaptive polynomial, increment of polynomial degree results to smaller RMS error when K is fixed (0.057-0.043). But for the same value of N, for example N=3, increasing cell from $K = 50$ to 100 provides significantly improvement. Error decreases from 0.0437 to 0.0244 for both error and gradient indicator used. It impiles that error in the sharp gradient area can be decreased by increasing the number of cell, not polynomial degree. When we apply the adaptive polynomial criteria by P^1/P^2, P^1/P^3 and P^2/P^3 for K fixed, the RMS errors are almost the same value. It shows again that the adaptive polynomial gives only small improvement and in order of degree $N = 3$ used. It can be concluded that the adaptive mesh method should be applied instead of using the adaptive polynomial method in the case of shock deformation or high gradient appearing in the computational domain. The resluts by the adaptive mesh method for the Burgers' equation will be shown in the next section.

B.2 Adaptive mesh RKDG method

Some numerical results by error and gradient indicators when fixing the degree of polynomial are shown in this section. We have varied four different values of $maxlev$ from 1 to 3 in the adaptive mesh algorithm. The comparison between the exact solution and the approximate solu-

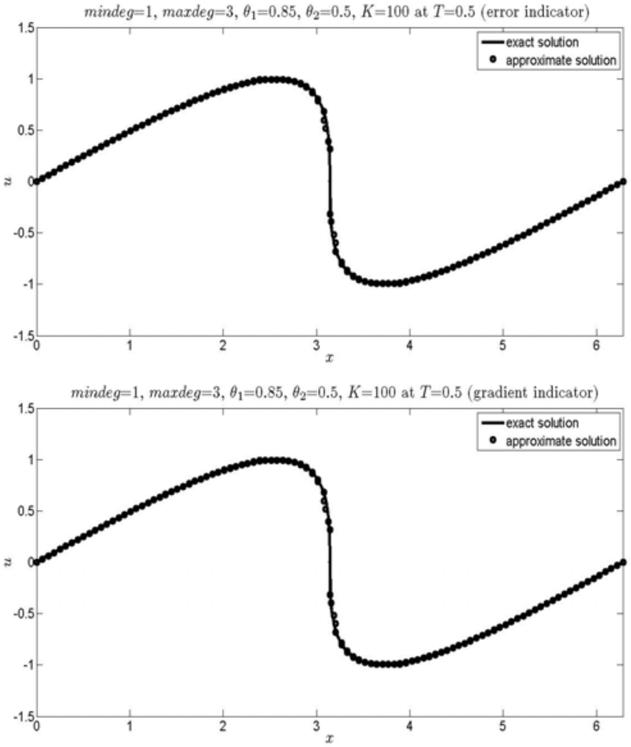

Figure 5: The comparisons between the exact solution and the approximate solution at $T = 0.5$, for $mindeg = 1$, $maxdeg = 3$ $K = 100$, using error indicator (top) and gradient indicator (bottom).

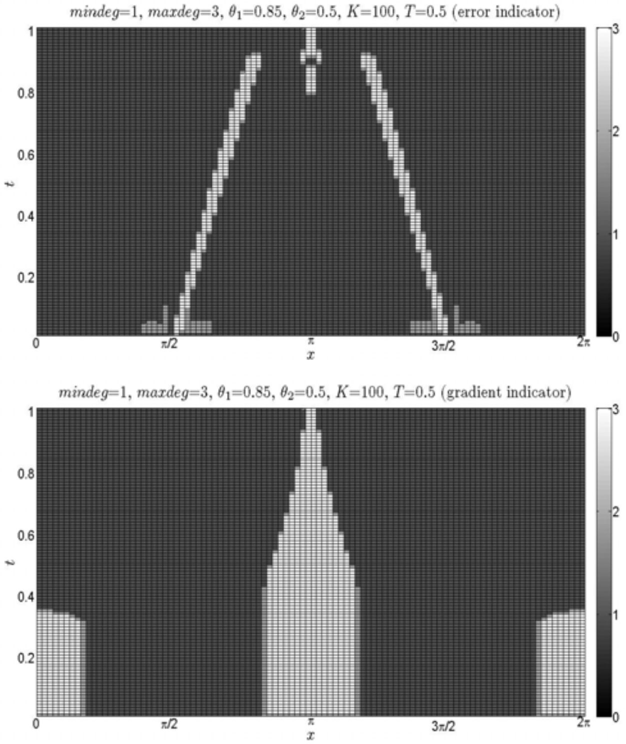

Figure 6: The adaptive area for $mindeg = 1$, $maxdeg = 3$ $K = 100$, $(\theta_1, \theta_2) = (0.85, 0.5)$ using error indicator (top) and gradient indicator (bottom).

Table VIII: The RMS errors when using gradient indicator with $K = 50$ and $K = 100$ cells for $\theta_1 = 0.85$ and $\theta_2 = 0.5$.

	$K = 50$	$K = 100$
P^1	0.0579	0.0325
P^2	0.0495	0.0276
P^3	0.0437	0.0244
P^1/P^2	0.0495	0.0277
P^1/P^3	0.0438	0.0244
P^2/P^3	0.0437	0.0244

tion when we use $N = 2$, $K = 100$, $(\theta_1, \theta_2) = (0.85, 0.5)$, and final time T = 0.5 is shown in Figure 7. It shows that the numerical results from the error and gradient indicator are in good agreement with the exact solution. They provide similar results of numerical accuracy. But they are different in detecting adaptive areas.

Figure 8 shows the adaptive mesh area in each time steps from t = 0 to t = 1. The error indicator detects troubled cell zone dynamically. It increases slightly the adaptive level. The shock area appears at the middle of the computational domain as time increases. At the early step, the error indicator does not detect this zone. It detects later when this zone has relatively large error. In contrast to the gradient indicator, the adaptive areas are static for all time step calculated. The adaptive areas appear on the left, the middle, and the right of the computational domain due to large gradients. It shows different patterns in detecting adaptive areas by these two indicators.

When we use polynomial degree N = 1 or 2 as a basis function, and the number of cell K = 50 or 100 cells, the numerical results for various $maxlev$ comparing with the RKDG method without adaptive mesh criteria are shown in Tables IX and X, for error and gradient indicators, respectively. Numerical accuracy in terms of the RMS error for various cases of $maxlev$ in comparing with the RKDG method without adaptive mesh criteria are shown in Tables IX and X. Superscipt and subscript of P refer to the polynomial degree and $maxlev$ respectively. It can be seen that for fixing the degree of polynomial and K, the RMS error decreases as $maxlev$ increases. Case of larger K provides smaller RMS errors. Also, using higher polynomial degree can improve accuracy in the case of Burger's equation because the solution remain has many smooth areas such that increasing order of approximation can improve solution accuracy.

Table IX: The RMS error using $N = 1$ and $N = 2$ with $K = 50$ and 100, and $(\theta_1, \theta_2) = (0.85, 0.5)$, error indicator is applied.

$maxlev$	$K = 50$	$K = 100$
P^1	0.0579	0.0325
P^1_1	0.0340	0.0171
P^1_2	0.0222	0.0116
P^1_3	0.0189	0.0103
$maxlev$	$K = 50$	$K = 100$
P^2	0.0360	0.0202
P^2_1	0.0249	0.0111
P^2_2	0.0143	0.0065
P^2_3	0.0136	0.0062

Table X: The RMS error using $N = 1$ and $N = 2$ with $K = 50$ and 100, and $(\theta_1, \theta_2) = (0.85, 0.5)$, gradient indicator is applied.

$maxlev$	$K = 50$	$K = 100$
P^1	0.0579	0.0325
P^1_1	0.0348	0.0188
P^1_2	0.0235	0.0109
P^1_3	0.0184	0.0068
$maxlev$	K=50	K=100
P^2	0.0360	0.0202
P^2_1	0.0249	0.0111
P^2_2	0.0138	0.0063
P^2_3	0.0089	0.0036

C. *Adaptive mesh RKDG method for the shallow water equations*

In this section, we apply the RKDG method for solving the one-dimensional shallow water equations in case of no source terms. We consider in both wet and dry beds. The initial condition is assumed to be discontinuous with two different water levels on the left and right at the point of x = 0. This problem is called the one-dimensional dam break problem. Due to its initially discontinuous condition, the solution developes a moving sharp gradient (call shock wave) in the computational domain. The adaptive polynomial strategy cannot improve numerical accuracy. Thus, we apply only the RKDG method with the adaptive mesh criteria to solve the shallow water equations in this section.

Wet bed case

The initial water levels are assumed as

$$h\left(x,t\right) = \begin{cases} 1, & \text{if } x < 0, \\ 0.6, & \text{if } x > 0. \end{cases} \tag{45}$$

The initial velocity is assumed to be zero meaning that fluid is initially at rest.

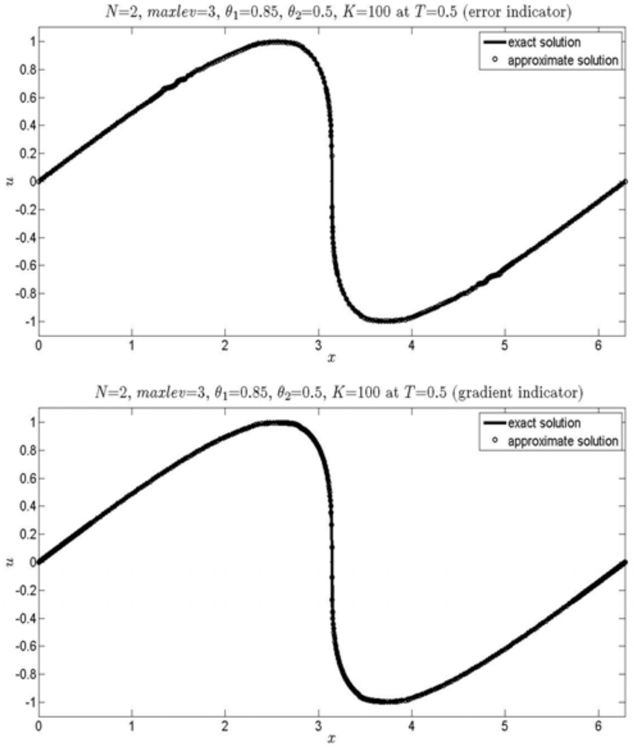

Figure 7: The comparisons between the exact solution and the approximate solution at $T = 0.5$, for $maxlev = 2$ (top), using error indicator (top) and gradient indicator (bottom).

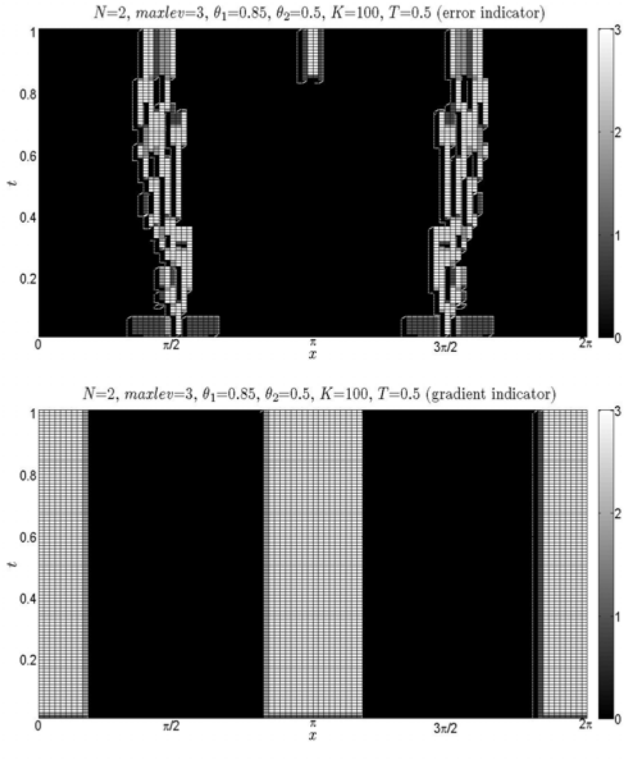

Figure 8: The adaptive area for $N = 2$, $K = 100$, $(\theta_1, \theta_2) = (0.85, 0.5)$ using error indicator (top) and gradient indicator (bottom).

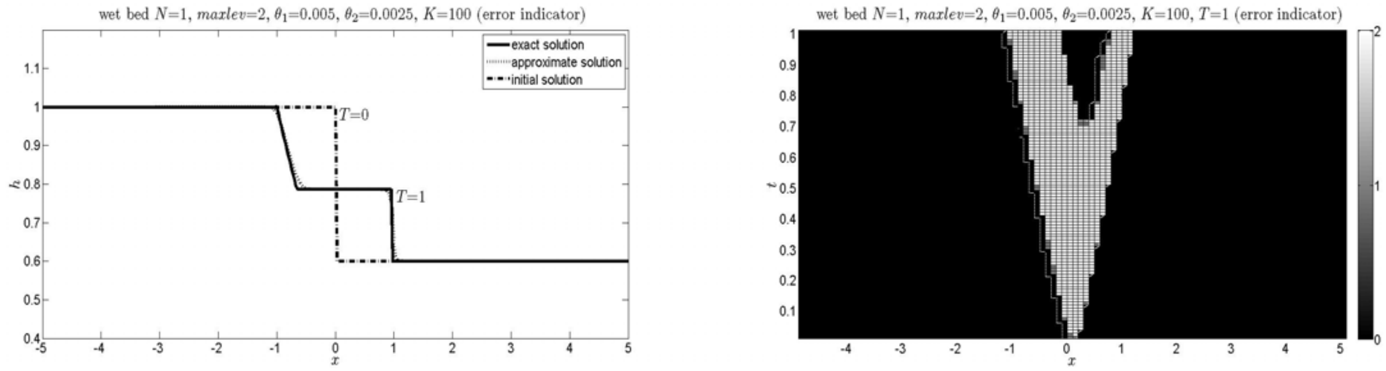

Figure 9: The comparison between the exact solution (solid line) and the approximate solution (dot line) at $T = 1$ for $maxlev = 2$ (left), and adaptive level in the x-t plane (right) by error indicator.

In Figure 9 (top), we set $N = 1$, $K = 100$, and $(\theta_1, \theta_2) = (0.005, 0.0025)$. It is shown that at $T = 1$, the sharp front and rarefaction can be captured correctly. Various cells have been detected as troubled cells. They are divided to be smaller sub-cells in the high gradient zones during time integration. The smallest mesh spacing occurs when the adaptive level is 2. The level of mesh for each cell in the time domain is shown in Figure 9 (bottom). It is shown that, the error indicator can detect troubled cells which are moving with the high gradient solution profile. The troubled cells are associate with the sharp-front and rarefaction areas. The color bar shows the values of mesh level for each time step. It look likes a v-shape. We have also investigated the numerical accuracy by using gradient indicator. Its function is similar to the error indicator. That is it can detect the troubled cell zones like a v-shape, see in Figure 10.

The numerical results for various values of $maxlev$ and polynomial degree when comparing with the RKDG method without adaptive mesh strategy are shown in Tables XI (error indicator) and XII (gradient indicator). It can be seen that for a fixed degree of polynomial, the RMS error decreases as the $maxlev$ increases, and it shows that the usage of many starting cells K provides higher accuracy. By fixing the polynomial degree N = 1, the case of K = 200 and $maxlev$ = 2 gives to the smallest RMS error for both the error and gradient indicators applied. When using polynomial degree N=2, case of $maxlev$ = 4 gives the smallest RMS error. Troubled cells have been detected so that smaller subcells can capture the high gradient of moving shock and rarefaction wave. For the same values of $maxlev$ and K, increasing the polynomial degree from 1 to 2 results to slighly decreasing the RMS error. It impiles that increasing the number of cells has much more impact to solution accuracy than the increasing the order of approximation in the case of moving shock.

Dry bed case

For dry bed problem, the initial water levels are assumed as

$$h(x,t) = \begin{cases} 1 & \text{if } x \leq 0, \\ 0 & \text{if } x > 0. \end{cases} \tag{46}$$

The initial velocity is assumed to be zero meaning that fluid is initially at rest.

In Figure 11 (top), we set $N = 2$, $K = 50$, and $(\theta_1, \theta_2) = (0.005, 0.0025)$, various cells have been detected as troubled cells which they can capture the rarefaction correctly. They are divided to be smaller sub-cells in the high gradient zones during time integration. The smallest mesh spacing occurs when the adaptive level is 4. The level of mesh for each cell in the time domain is shown in Figure 11 (bottom). In this case the troubled cells are associate with the rarefaction areas that are developed from the initial condition. We have also investigated the numerical accuracy by using the gradient indicator. The results is similar, see in Figure 12.

The numerical results for various values of $maxlev$ and polynomial degree when comparing with the RKDG method without adaptive mesh strategy are shown in Tables XIII (error indicator) and XIV (gradient indicator). It can be seen that for a fixed degree of polynomial, the RMS error decreases as the $maxlev$ increases. By fixing the polynomial degree, the case of K = 100 and $maxlev$ = 4 results to the smallest RMS error for both the error and gradient indicators applied. For the same values of $maxlev$ and K, increasing the polynomial degree from 1 to 2 results to slighly decreasing the RMS error.

V. Conclusions

In this work, we present the adaptive RKDG method for solving the one-dimensional advection equation, the Burgers' equation and the shallow water equations. We consider the smooth initial condition and the discontinu-

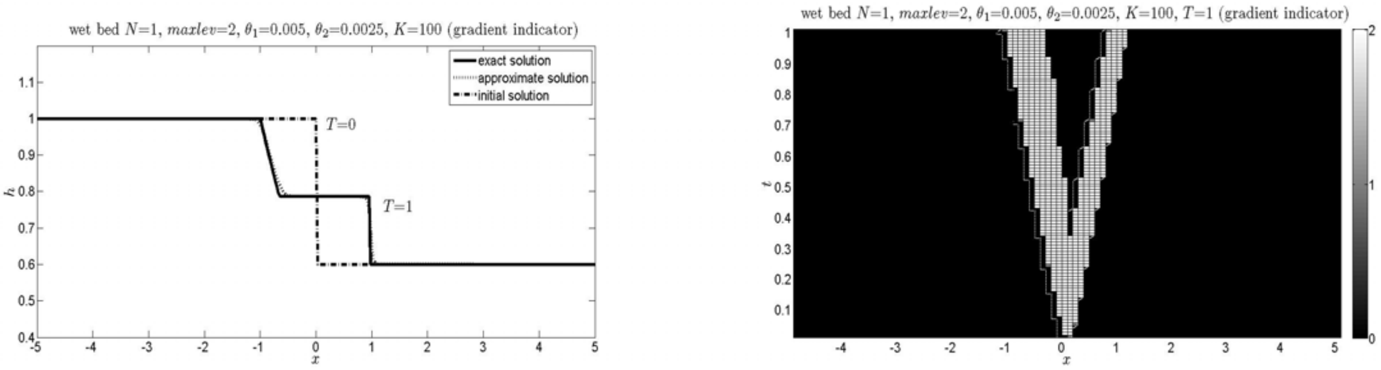

Figure 10: The comparison between the exact solution (solid line) and the approximate solution (dot line) at $T = 1$ for $maxlev = 2$ (left), and adaptive level in the x-t plane (right) by gradient indicator.

Table XI: The RMS error using $N = 1$ and $N = 2$ with $K = 50, 100$, and 200 and $(\theta_1, \theta_2) = (0.005, 0.0025)$ for wet bed case, error indicator is applied.

$maxlev$	$K = 50$	$K = 100$	$K = 200$	$maxlev$	$K = 50$	$K = 100$	$K = 200$
P^1	0.0230	0.0145	0.0088	P^2	0.0184	0.0137	0.0091
P^1_1	0.0145	0.0088	0.0060	P^2_1	0.0137	0.0089	0.0058
P^1_2	0.0111	0.0075	0.0053	P^2_2	0.0093	0.0060	0.0043
P^1_3	0.0115	0.0079	0.0057	P^2_3	0.0072	0.0050	0.0035
P^1_4	0.0128	0.0088	0.0056	P^2_4	0.0067	0.0046	0.0028

Table XII: The RMS error using $N = 1$ and $N = 2$ with $K = 50, 100$, and 200 $(\theta_1, \theta_2) = (0.005, 0.0025)$ for wet bed problem, gradient indicator is applied.

$maxlev$	$K = 50$	$K = 100$	$K = 200$	$maxlev$	$K = 50$	$K = 100$	$K = 200$
P^1	0.0230	0.0145	0.0088	P^2	0.0184	0.0137	0.0091
P^1_1	0.0145	0.0088	0.0060	P^2_1	0.0121	0.0085	0.0074
P^1_2	0.0111	0.0075	0.0053	P^2_2	0.0100	0.0061	0.0043
P^1_3	0.0113	0.0077	0.0056	P^2_3	0.0072	0.0049	0.0035
P^1_4	0.0123	0.0085	0.0053	P^2_4	0.0063	0.0043	0.0026

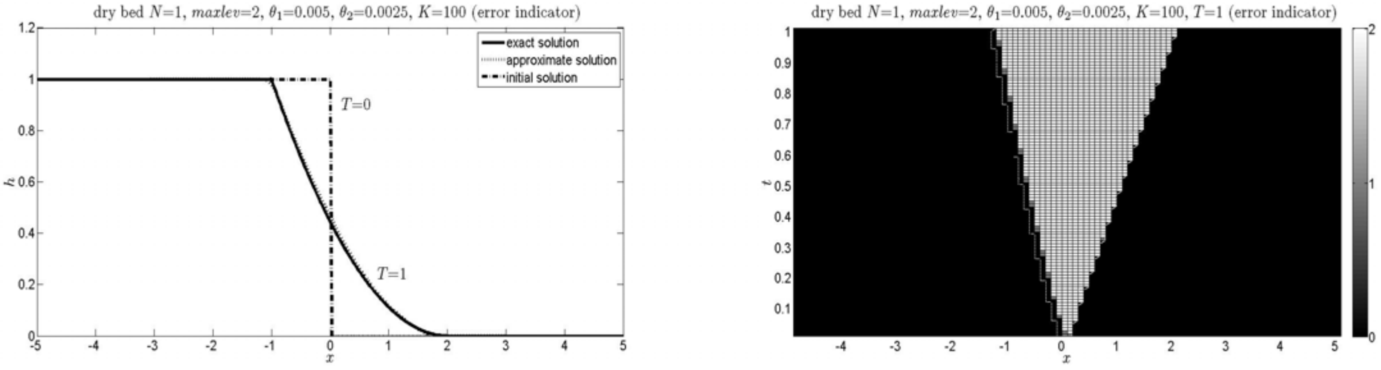

Figure 11: The comparison between the exact solution (solid line) and the approximate solution (dot line) at $T = 1$, $maxlev = 2$ (left) and adaptive area (right) using error indicator.

Table XIII: The RMS error using $N = 1$ and $N = 2$ with $K = 25, 50$, and 100 and $(\theta_1, \theta_2) = (0.005, 0.0025)$ for dry bed case, error indicator is applied.

$maxlev$	$K = 25$	$K = 50$	$K = 100$	$maxlev$	$K = 25$	$K = 50$	$K = 100$
P^1	0.0289	0.0288	0.0154	P^2	0.0489	0.0208	0.0115
P^1_1	0.0287	0.0154	0.0082	P^2_1	0.0208	0.0111	0.0061
P^1_2	0.0265	0.0139	0.0071	P^2_2	0.0151	0.0082	0.0043
P^1_3	0.0259	0.0134	0.0068	P^2_3	0.0144	0.0076	0.0039
P^1_4	0.0258	0.0133	0.0067	P^2_4	0.0140	0.0074	0.0038

Table XIV: The RMS error using $N = 1$ and $N = 2$ with $K = 25, 50$, and 100, $(\theta_1, \theta_2) = (0.005, 0.0025)$ for dry bed problem, gradient indicator is applied.

$maxlev$	$K = 25$	$K = 50$	$K = 100$	$maxlev$	$K = 25$	$K = 50$	$K = 100$
P^1	0.0289	0.0288	0.0154	P^2	0.0489	0.0208	0.0115
P^1_1	0.0287	0.0154	0.0082	P^2_1	0.0210	0.0112	0.0061
P^1_2	0.0264	0.0138	0.0071	P^2_2	0.0150	0.0080	0.0042
P^1_3	0.0258	0.0132	0.0067	P^2_3	0.0141	0.0077	0.0038
P^1_4	0.0257	0.0131	0.0066	P^2_4	0.0142	0.0077	0.0038

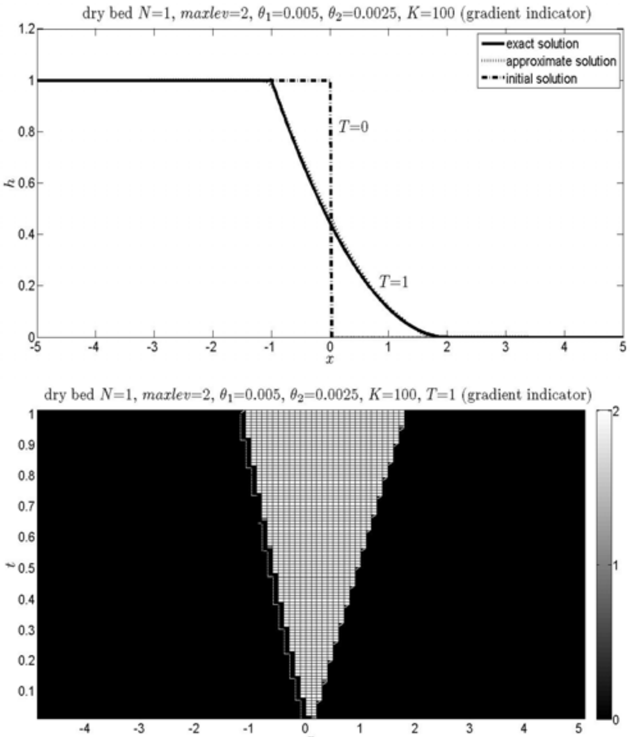

Figure 12: The comparison between the exact solution (solid line) and the approximate solution (dot line) at $T = 1$ for $maxlev = 2$ (top), and adaptive level in the x-t plane (bottom) by gradient indicator.

ous initial condition for the advection equation, the initial sine wave for the Burgers' equation, and for shallow water equations, we consider two problems which are wet and dry bed problems with discontinuous initial condition.

There are two types of adaptive algorithms that are the adaptive polynomial for solving the smooth initial condition and the adaptive mesh for solving the discontinuous initial condition. We also present two types of indicators which are the error and gradient indicators. The indicators are used to detect troubled cells in the computational domain before applying the adaptive criteria during the time integration.

The adaptive polynomial is appropriate for increasing the accuracy of numerical solutions in the case of the smooth solutions, see the results from the advection equation. Our presented approach can increase automatically the degree of polynomial basis for troubled cells, and revesely it can decrease automatically the degree of polynomial if that cells are usual. The highest accuracy can be obtained if the maximum degree is used. However, the results from the adaptive polynomial for the Burgers' equation show that increasing order of polynomial basis cannot improve accuracy for sharp-front solution, we should apply the adaptive mesh criteria instead of adaptive polynomial for this case. Since solution from the shallow water equations also have sharp-front and high gradient area, then we apply only adaptive mesh ctiteria for solving the equations. The numerical results by the adaptive mesh method for the advection equation, the Burgers' equation and the shallow water equations are presented. It is found that the solution accuracy increases as the maximum level increases. For these two criterias, the values of θ_1 and θ_2 are depended on desired order of accuracy. Our presented adaptive mesh method are successfully applied to capture some shock interfaces, rarefaction and high gradient areas in our model problems.

ACKNOWLEDGMENT

This research is supported by Science Achivement Scholarship Thailand(SAST) and Center of Advanced Studies in Industrial Technology, Faculty of Engineering, Kasetsart University.

REFERENCES

[1] A. Harten and H. Tal-Ezer, "On a Fourth Order Accurate Implicit Finite Difference Scheme for Hyperbolic conservation Laws.II Five-point Scheme," *J. Comput. Phys.* 41, 1981, pp. 329–356.

[2] B. Cockburn, C.–W. Shu, C. Johnson and E. Tadmor, "Advanced Numerical Approximation of Nonlinear Hyperbolic Equations" in *Lectures Given at the 2nd Session of the Centro Internazionale Matematico Estivo (C.I.M.E.) held in Cetraro, Italy , June 23-28, 1997,* Springer –Verlag, Telos, 1998.

[3] B. Cockburn and C.–W. Shu, "TVB Runge-Kutta Local projection Discontinuous Finite Element Method for conservation Laws II. General Framework," *J. Math. Comput.* 52, 1989, pp. 411–435.

[4] B. van Leer, "Toward the ultimate conservation difference scheme ii.," *J. Comput. Phys.* 14, 1974, pp. 361–376.

[5] B. van Leer, "Toward the ultimate conservation difference scheme v.," *J. Comput. Phys.* 32, 1979, pp. 1–136.

[6] C.–W. Shu, "TVD time discretizations," *SIAM J. Sci. Stat. Comput.* 9, 1988, pp. 1073–1084.

[7] D. Schwanenberg, R. Kiem, and J. Kongeter, "A discontinuous Galerkin method for the shallow-water equations with source terms. Discontinuous Galerkin Methods: Theory, Computations and Applications, B. Cockburn, G. E. Karniadaki, and C.-W. Chu, Eds." in *Lecture Notes in Computational Science and Engineering,* Springer, 2000, pp. 1419–424.

[8] E.–F. Toro, *iemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction,* Springer, 1999.

[9] E.–F. Toro, *Shock-Capturing Methods for Free-Surface Shallow Flows ,* John Willy and sons LTD, 2001.

[10] E.–J. Kubatko, S. Bunya, C. Dawson and J.–J. Westerink, "Dynamic p-adaptive Runge-Kutta discontinuous Galerkin methods for the shallow water equations," *Comput. Methods Appl. Mech. Engrg.* 198, 2009, pp. 1766–1774.

[11] G. Manzini, "A second-order TVD implicit-explicit finite volume method for time-dependent convection reaction equations," *Math. Comput. Simulat.* 79, 2009, pp. 2403–2428.

[12] H. Zhu and J. Qiu, "Adaptive Runge-Kutta discontinuous Galerkin methods using different indicators: One-dimensional case," *J. Comput. Phys.* 228, 2009, pp. 6957–6976.

[13] J.–D. Logan, *An introduction to nonlinear partial differential equations,* John Willy, 2008.

[14] J.–S. Hesthaven and T. Warburton, *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications,* Springer, 2008.

[15] P.–A. Tassi and C.–A. Vionnet, "Discontinuous Galerkin Method for the one dimensional simulation of shallow water flows," *Mecànica Computational Vol. XXII, Bahía Blanca* 22, 2003, pp. 2403–2428.

[16] R.–J. Leveque, *Finite Volume Method for Hyperbolic Problem,* Cambridge University Press, 2005.

[17] R.–J. Leveque, *Numerical Methods for Conservation Laws,* Birkhӓuser Verlag Press, 1992.

[18] Z. Xu, J. Xu, and C.–W. Shu, "A high order adaptive finite element method for solving nonlinear hyperbolic conservation laws," *Technical Report 2010-14*, Scientific Computing Group, BrownUniversity, Providence, RI, USA, Apr., 2010,