Thai Journal of Mathematics
Volume 17 (2019) Number 3 : 685–702
http://thaijmath.in.cmu.ac.th
ISSN 1686-0209

Numerical Simulation of Water-Quality Model on Flooding Using Revised Lax-Diffusive and Modified Siemieniuch-Gladwell Methods

Kanawoot Subklay[1] **and Nopparat Pochai**

Department of Mathematics, Faculty of Science
King Mongkut's Institute of Technology Ladkrabang
Bangkok 10520, Thailand

Centre of Excellence in Mathematics, Commission on Higher Education
(CHE), Si Ayutthaya Road, Bangkok 10400, Thailand
e-mail : darkenesis1@hotmail.com (K. Subklay)
nop_math@yahoo.com (N. Pochai)

Abstract : In 2011, Thailand has been confronted a largest flooding. The mass of water has been drenched from many main and branch rivers to cover wide areas. The residents who lived in the flooding area have to build a manmade sandbag dike to protect their village. The flooding has been taken for a long time meanwhile the flooding water becomes contaminated. There are some residents in their flooding area want to drain their contaminated water to a nearest area. They have been destroyed their sandbag dike. Consequently, the dispute among residents is occurred. In this research, a mathematical simulation of a water-quality on a long period flooding using a couple of two models is proposed. The first model is the one-dimensional shallow water equations that provide the water elevation and velocity. The second model is a one-dimensional advection-dispersion equation that provides the water pollutant concentrations after the sandbag dike has been destroyed. A revised Lax-diffusive is used to approximate the solution of the first model. Consequently, the numerical solutions of the second model are obtained by using the traditional and modified Siemieniuch-Gladwell schemes.

[1]Corresponding author.

Keywords : finite differences; Lax-diffusive scheme; revised Lax-diffusive scheme; one-dimensional; Dam-break model; shallow water equations; dispersion model; advection-dispersion equation.
2010 Mathematics Subject Classification : 76R50; 39A14; 35Q30; 35L40.

1 Introduction

Two mathematical models are used to explain the situation. The first model is one-dimensional hydrodynamic model that gives the velocity and elevation of water. The second model is dispersion model that gives the pollutant concentration of water after the sandbag-dike has been destroyed. In the recent years, there has been many research magnitude on the evolution of numerical models to simulate dam-break flows. For example, In [1] used Lax-Friendrichs to compare MacCormack and MacCormack TVD with a one dimension (1D) dam-break flow simulation. In [2] used central scheme for 1D and two dimension (2D) dam-break simulation. In [3] used finite volume method for numerical solution of shallow water equations in dam-break with flat topography. In [4] used 2D finite volume multiblock flow solver. The model is based on Flux Vector Splitting method. In [5] used a robust and effective flux-vector splitting method to simulate dam-break problem base on finite volume method on a cartesian grid. In [6] used smoothed particle hydrodynamics (SPH) to solve shallow-water dam break flow in open channels. In [7] used a new well-balanced unstraggered central finite volume scheme for 1D and 2D dam break over a rectangular bump. In [8] used well-balanced hydrostatic upwind schemes for dam-break approximations. The dam-break model is used to Explain unsteady dike failure flow. In [9] used Implicit (PriessMan) and Explicit (Lax Diffusive) methods for Saint-Venent Equations to Simulate Flood wave in Natural Rivers. In this work, the revised Lax diffusive technique is used to solve dike failure problem.

2 Model Formulation

In this section, the couple of mathematical models are used to describe. The first model is one-dimensional hydrodynamic model that provide water elevation and water velocity. The second model is advection-diffusion-equation that give water pollutant concentration. The second model need to input water velocity from the first model.

2.1 A Dam-Break Model

The Navier-Stokes equations over the flow depth with two assumptions, the hydrostatic pressure distributions and a small bottom slope, are govern the one-dimensional hydrodynamic equations. Since the dam-break flow the model has the high level velocity can be considered as the advection-dominated shallow water

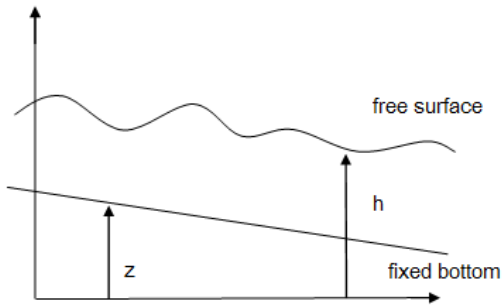


Figure 1: The shallow water system.

flows. These are the eddy viscosity terms can be omitted. The model equations can be described in the system of partial differential equations:

$$\partial_x \begin{pmatrix} h \\ hu \end{pmatrix} + \partial_t \begin{pmatrix} hu \\ hu^2 + \frac{1}{2}gh^2 \end{pmatrix} = \begin{pmatrix} 0 \\ -gh\partial_x z \end{pmatrix} \tag{2.1}$$

where x is the longitudinal length along a stream (m), t is time (s) and $h(x,t)$ is the depth of water (m). The velocity of water is defined by $u(x,t)$ (m/s) and $z(x)$ is the function of bottom topography (m). g is gravitational constant $(9.8\ m/s^2)$ for all $x \in [0, L]$. The initial conditions are given by

$$u(x,0) = 0, \text{ for all } 0 \le x \le L, \tag{2.2}$$

and

$$h(x,0) = \begin{cases} h_l & \text{if } 0 \le x \le \frac{L}{2}, \\ h_r & \text{if } \frac{L}{2} < x \le L, \end{cases} \tag{2.3}$$

where $h_l > h_r$, the water flow from the upstream to downstream at $t = 0$. The Neumann boundary conditions are also given by

$$u_x(0,t) = u_x(L,t) = 0, \tag{2.4}$$

$$h_x(0,t) = h_x(L,t) = 0. \tag{2.5}$$

2.2 Dispersion Model

The governing equations are the one-dimensional advection-diffusion equations (ADE). This equations give a water pollutant concentration and a simplified form is shown in [10] as,

$$\frac{\partial C}{\partial t} + u\frac{\partial C}{\partial x} = D\frac{\partial^2 C}{\partial x^2}, \tag{2.6}$$

where $C(x,t)$ is the pollutant concentration of water at the point $x(m)$ and time $t(s)$. D is the diffusion coefficient and $u(x,t)$ is the velocity component (m/s) for all $x \in [0,L]$. The initial conditions are given by

$$C(x,0) = C_0 \text{ for all } 0 \le x \le L. \tag{2.7}$$

The boundary conditions are provided by

$$C(0,t) = C_t \text{ for all } t > 0, \tag{2.8}$$

$$C_x(L,t) = C_R \text{ for all } t > 0, \tag{2.9}$$

where $C(t)$ is the function depends on t. C_0 and C_R are constants.

3 Numerical Techniques

The water height and water velocity are obtained by the dam-break model. The dispersion model have to input velocity field from the first model. The second model which provides the pollutant concentration field.

3.1 A Revised Lax-Diffusive Method for a Dam-Break Model

In this section, the revised method of a traditional Lax-diffusive method for the dam-break model of [9] is proposed. We now discretize Eq.(2.1) by dividing the interval $[0,L]$ into M subintervals such that $M\Delta x = L$ and the interval $[0,T]$ into N subintervals such that $N\Delta t = T$. We can then approximate $h(x_m,t_n)$ by h_m^n, value of the difference approximation of $h(x,t)$ at point $x = m\Delta x$ and $t = n\Delta t$, where $0 \le m \le M$ and $0 \le n \le N$, and similarly defined for u_m^n. The grid point (x_m,t_n) is defined by $x_m = m\Delta x$ for all $m = 0,1,2,\dots,M$ and $t_n = n\Delta t$ for all $n = 0,1,2,\dots,N$ in which M and N are positive integers.

We will modify f^* from two points averaged [9] to be the three points averaged. The discretization of Eq.(2.1) is base on a Lax-diffusive scheme. The semi-discrete scheme is applied to Eq.(2.1) and using a uniform spatial grid $(x_m,t_n) = (m\Delta x, n\Delta t)$, we can define

$$f_x \approx \frac{f_{m+1}^n - f_{m-1}^n}{2\Delta x}, \tag{3.1}$$

$$f_t \approx \frac{f_m^{n+1} - f_m^*}{\Delta t}, \tag{3.2}$$

where

$$f^* = \frac{f_{m+1}^n + f_m^n + f_{m-1}^n}{3}. \tag{3.3}$$

The partial derivative of h and u with respect to x and t are approximated by using Eqs.(3.1-3.3), respectively. We can see that Eq.(2.1) is written in a matrix form as

$$A_t + B_x + C = 0, \tag{3.4}$$

where

$$A = \begin{pmatrix} h \\ hu \end{pmatrix}, B = \begin{pmatrix} hu \\ hu^2 + \frac{1}{2}gh^2 \end{pmatrix}, C = \begin{pmatrix} 0 \\ -gh\partial_x z \end{pmatrix}. \tag{3.5}$$

It follows that Eq.(3.5) can be written by the uniform spatial grids as

$$A_m^n = \begin{pmatrix} h_m^n \\ h_m^n u_m^n \end{pmatrix}, B_m^n = \begin{pmatrix} h_m^n u_m^n \\ h_m^n (u_m^n)^2 + \frac{1}{2}g(h_m^n)^2 \end{pmatrix}, C = \begin{pmatrix} 0 \\ -gh_m^n \partial_x z \end{pmatrix}. \tag{3.6}$$

Substituting the finite difference approximations of Eqs.(3.1-3.2) and Eq.(3.3) into Eq.(3.4), we obtain that

$$A_m^{n+1} = \frac{\Delta t}{2\Delta x}(B_{m-1}^n - B_{m+1}^n) + A^* \tag{3.7}$$

where $A^* = \begin{pmatrix} h^* \\ (hu)^* \end{pmatrix}$. Substituting Eq.(3.6) into Eq.(3.7), we can see that

$$\begin{aligned} \begin{pmatrix} h_m^{n+1} \\ h_m^{n+1} u_m^{n+1} \end{pmatrix} =& \frac{\Delta t}{2\Delta x} \begin{pmatrix} h_{m-1}^n u_{m-1}^n - h_{m+1}^n u_{m+1}^n \\ h_{m-1}^n (u_{m-1}^n)^2 - h_{m+1}^n (u_{m+1}^n)^2 + \frac{1}{2}g((h_{m-1}^n)^2 - (h_{m+1}^n)^2) \end{pmatrix} \\ &+ \frac{1}{3} \begin{pmatrix} h_{m-1}^n + h_m^n + h_{m+1}^n \\ h_{m-1}^n u_{m-1}^n + h_m^n u_m^n + h_{m+1}^n u_{m+1}^n \end{pmatrix}. \end{aligned} \tag{3.8}$$

for all $1 \leq m < M$ and $0 \leq n \leq N-1$. For upper boundary, where $m = 0$, plug the known value of the left boundary by $u_{-1}^n = u_0^n$ and $h_{-1}^n = h_0^n$ into Eq.(3.8) in the right-hand side, we obtain

$$\begin{aligned} \begin{pmatrix} h_1^{n+1} \\ h_1^{n+1} u_1^{n+1} \end{pmatrix} =& \frac{\Delta t}{2\Delta x} \begin{pmatrix} h_0^n u_0^n - h_1^n u_1^n \\ h_0^n (u_0^n)^2 - h_1^n (u_1^n)^2 + \frac{1}{2}g((h_0^n)^2 - (h_1^n)^2) \end{pmatrix} \\ &+ \frac{1}{3} \begin{pmatrix} 2h_0^n + h_1^n \\ 2h_0^n u_0^n + h_1^n u_1^n \end{pmatrix}. \end{aligned} \tag{3.9}$$

For lower boundary, where $m = M$, substituting the approximate unknown value of the right boundary by boundary conditions, we can let $u_{M+1}^n = u_M^n$ and $h_{M+1}^n = h_M^n$ by rearranging, we obtain

$$\begin{aligned} \begin{pmatrix} h_M^{n+1} \\ h_M^{n+1} u_M^{n+1} \end{pmatrix} =& \frac{\Delta t}{2\Delta x} \begin{pmatrix} h_{M-1}^n u_{M-1}^n - h_M^n u_M^n \\ h_{M-1}^n (u_{M-1}^n)^2 - h_M^n (u_M^n)^2 + \frac{1}{2}g((h_{M-1}^n)^2 - (h_M^n)^2) \end{pmatrix} \\ &+ \frac{1}{3} \begin{pmatrix} h_{M-1}^n + 2h_M^n \\ h_{M-1}^n u_{M-1}^n + 2h_M^n u_M^n \end{pmatrix}. \end{aligned} \tag{3.10}$$

The stability condition of the scheme needed CFL number as [9],

$$C_n = u_{max}(\frac{\Delta t}{\Delta x}) \leq 1. \tag{3.11}$$

3.2 Numerical Method for a Dispersion Model

We consider both implicit and explicit methods for solving advection-diffusion equations. The well-known traditional methods are also introduced [10].

3.2.1 The Modified Siemieniuch-Gladwell Implicit Procedure

We can then approximate $C(x_i, t_n)$ by C_i^n, the value of the difference approximation of $C(x,t)$ at point $x = i\Delta x$ and $t = n\Delta t$, where $0 \le i < M$ and $0 \le n < N$. The grid point (x_n, t_n) is defined by $x_i = i\Delta x$ for all $i = 0, 1, 2, ..., M$ and $t_n = n\Delta t$ for all $n = 0, 1, 2, ..., N$ in which M and N are positive integers. Taking the modified Siemieniuch-Gladwell technique [10] into Eq.(2.6), by the following discretizations:

$$\begin{aligned}\frac{\partial C}{\partial t} \approx& \frac{(2\beta - \alpha_m^n)}{4}\frac{(C_{m-1}^{n+1} - C_{m-1}^n)}{\Delta t} + \frac{(2 - 2\beta + \alpha_m^n)}{2}\frac{(C_m^{n+1} - C_m^n)}{\Delta t} \\ &+ \frac{(2\beta - \alpha_m^n)}{4}\frac{(C_{m+1}^{n+1} - C_{m+1}^n)}{\Delta t},\end{aligned} \tag{3.12}$$

$$\frac{\partial C}{\partial x} \approx \frac{(C_{m+1}^n - C_{m-1}^n)}{4\Delta x} + \frac{(C_{m+1}^{n+1} - C_{m-1}^{n+1})}{4\Delta x}, \tag{3.13}$$

$$\frac{\partial^2 C}{\partial x^2} \approx \frac{1}{2}\frac{(C_{m+1}^{n+1} - 2C_m^{n+1} + C_{m-1}^{n+1})}{(\Delta x)^2} + \frac{1}{2}\frac{(C_{m+1}^n - 2C_m^n + C_{m-1}^n)}{(\Delta x)^2}, \tag{3.14}$$

$$u \approx \widetilde{u}_m^n, \tag{3.15}$$

where $u = \widetilde{u}_m^n$ are obtained by the revised Lax-diffusive method Eq.(3.8). Substituting Eqs.(3.12-3.15) into Eq.(2.6), we have

$$-\alpha_m^n C_{m-1}^{n+1} + (2+\alpha_m^n) C_m^{n+1} = 2\beta C_{m-1}^n + (2-4\beta+\alpha_m^n) C_m^n + (2\beta - \alpha_m^n) C_{m+1}^n, \tag{3.16}$$

where $\alpha_m^n = u = \widetilde{u}_m^n(\frac{\Delta t}{\Delta x})$ and $\beta = D\frac{\Delta t}{(\Delta x)^2}$ for all $1 \le m < M$ and $0 \le n < N$. For the left boundary condition, $m = 0$, the known value on the left boundary are approximated by $C_{-1}^n = C_0^n$ and $C_{-1}^{n+1} = C_0^{n+1}$, we can see that

$$2C_0^{n+1} = (2 - 2\beta + \alpha_0^n)C_0^n + (2\beta - \alpha_0^n)C_1^n. \tag{3.17}$$

Similarly, the right boundary condition, $m = M$, the known value on the left boundary are approximated by $C_{M+1}^n = C_M^n$ and $C_{M+1}^{n+1} = C_M^{n+1}$ into Eq.(3.16) in the right-hand side, we have

$$-\alpha_M^n C_{M-1}^{n+1} + (2 + \alpha_M^n)C_M^{n+1} = 2\beta C_{M-1}^n + (2 - 2\beta)C_M^n. \tag{3.18}$$

The stability of modified Siemieniuch-Gladwell procedure is [10],

$$0 < \beta \le \frac{1 + \alpha_m^n}{2}. \tag{3.19}$$

3.2.2 The BTCS-Type Implicit Method

Consider the Backward in time cecter in space(BTCS) scheme for the advection-diffusion equation by the following discretizations:

$$\frac{\partial C}{\partial t} \approx \frac{(C_m^{n+1} - C_m^n)}{\Delta t}, \tag{3.20}$$

$$\frac{\partial C}{\partial x} \approx \frac{(C_{m+1}^{n+1} - C_{m-1}^{n+1})}{2\Delta x}, \tag{3.21}$$

$$\frac{\partial^2 C}{\partial x^2} \approx \frac{(C_{m+1}^{n+1} - 2C_m^{n+1} + C_{m-1}^{n+1})}{(\Delta x)^2}, \tag{3.22}$$

$$u \approx \widetilde{u}_m^n. \tag{3.23}$$

Substituting Eqs.(3.20-3.23) into Eq.(2.6), we have

$$-(\alpha_m^n + 2\beta)C_{m-1}^{n+1} + 2(1 + 2\beta)C_m^{n+1} + (\alpha_m^n - 2\beta)C_{m+1}^{n+1} = 2C_m^n, \tag{3.24}$$

for all $1 \leq m < M$ and $0 \leq n < N$. For the left boundary condition, $m = 0$, the known value on the left boundary are approximated $C_{-1}^{n+1} = C_0^{n+1}$, we can see that

$$(2 - \alpha_0^n)C_0^{n+1} + (\alpha_0^n - 2\beta)C_1^{n+1} = 2C_0^n. \tag{3.25}$$

Similarly, the right boundary condition, $m = M$, the known value on the left boundary are approximated $C_{M+1}^{n+1} = C_M^{n+1}$, we have

$$-(\alpha_M^n + 2\beta)C_{M-1}^{n+1} + (2 + 2\beta + \alpha_M^n)C_M^{n+1} = 2C_M^n. \tag{3.26}$$

The stability of BTCS scheme is unconditionally stable [10].

3.2.3 The Upwind Implicit Formula

Consider the upwind implicit scheme for the advection-diffusion equation by the following discretizations:

$$\frac{\partial C}{\partial t} \approx \frac{(C_m^{n+1} - C_m^n)}{\Delta t}, \tag{3.27}$$

$$\frac{\partial C}{\partial x} \approx \frac{(C_m^{n+1} - C_{m-1}^{n+1})}{\Delta x}, \tag{3.28}$$

$$\frac{\partial^2 C}{\partial x^2} \approx \frac{(C_{m+1}^{n+1} - 2C_m^{n+1} + C_{m-1}^{n+1})}{(\Delta x)^2}, \tag{3.29}$$

$$u \approx \widetilde{u}_m^n. \tag{3.30}$$

Substituting Eqs.(3.27-3.30) into Eq.(2.6), we have

$$-(\alpha_m^n + \beta)C_{m-1}^{n+1} + (1 + 2\beta + \alpha_m^n)C_m^{n+1} - \beta C_{m+1}^{n+1} = C_m^n, \tag{3.31}$$

for all $1 \leq m < M$ and $0 \leq n < N$. For the left boundary condition, $m = 0$, the known value on the left boundary are approximated by $C_{-1}^{n+1} = C_0^{n+1}$, we can see that

$$(1+\beta)C_0^{n+1} - \beta C_1^{n+1} = C_0^n. \tag{3.32}$$

Similarly, the right boundary condition, $m = M$, the known value on the left boundary are approximated $C_{M+1}^{n+1} = C_M^{n+1}$, we have

$$-(\alpha_M^n + \beta)C_{M-1}^{n+1} + (1+\beta+\alpha_M^n)C_M^{n+1} = C_M^n. \tag{3.33}$$

The stability of upwind implicit scheme is unconditionally stable [10].

3.2.4 The Crank-Nicolson Type Technique

Consider the Crank-Nicolson scheme for the advection-diffusion equation by the following discretizations:

$$\frac{\partial C}{\partial t} \approx \frac{(C_m^{n+1} - C_m^n)}{\Delta t}, \tag{3.34}$$

$$\frac{\partial C}{\partial x} \approx \frac{(C_m^{n+1} - C_{m-1}^{n+1})}{\Delta x}, \tag{3.35}$$

$$\frac{\partial^2 C}{\partial x^2} \approx \frac{(C_{m+1}^{n+1} - 2C_m^{n+1} + C_{m-1}^{n+1})}{(\Delta x)^2}, \tag{3.36}$$

$$u \approx \widetilde{u}_m^n. \tag{3.37}$$

Substituting Eqs.(3.34-3.37) into Eq.(2.6), we have

$$\begin{aligned} &-(\alpha_m^n + 2\beta)C_{m-1}^{n+1} + 4(1+\beta)C_m^{n+1} + (\alpha_m^n - 2\beta)C_{m+1}^{n+1} \\ &\quad = (\alpha_m^n + 2\beta)C_{m-1}^n + 4(1-\beta)C_m^n + (2\beta - \alpha_m^n)C_{m+1}^n, \end{aligned} \tag{3.38}$$

for all $1 \leq m < M$ and $0 \leq n < N$. For the left boundary condition, $m = 0$, the known value on the left boundary are approximated $C_{-1}^{n+1} = C_0^{n+1}$ and $C_{-1}^n = C_0^n$, we can see that

$$(4+2\beta-\alpha_0^n)C_0^{n+1} + (\alpha_0^n - 2\beta)C_1^{n+1} = (4-2\beta+\alpha_0^n)C_0^n + (2\beta-\alpha_0^n)C_1^n. \tag{3.39}$$

Similarly, the right boundary condition, $m = M$, the known value on the left boundary are approximated $C_{M+1}^{n+1} = C_M^{n+1}$ and $C_{M+1}^n = C_M^n$, we have

$$-(\alpha_M^n + 2\beta)C_{M-1}^{n+1} + (4+2\beta+\alpha_M^n)C_M^{n+1} = (\alpha_M^n + 2\beta)C_{M-1}^n + (4-2\beta-\alpha_M^n)C_M^n. \tag{3.40}$$

The stability of Crank-Nicolson scheme is unconditionally stable [10].

3.2.5 The FTCS-Type Scheme

The traditional forward time central space scheme is considered the following discretizations:

$$\frac{\partial C}{\partial t} \approx \frac{(C_m^{n+1} - C_m^n)}{\Delta t}, \tag{3.41}$$

$$\frac{\partial C}{\partial x} \approx \frac{(C_{m+1}^n - C_{m-1}^n)}{2\Delta x}, \tag{3.42}$$

$$\frac{\partial^2 C}{\partial x^2} \approx \frac{(C_{m+1}^n - 2C_m^n + C_{m-1}^n)}{(\Delta x)^2}, \tag{3.43}$$

$$u \approx \widetilde{u}_m^n. \tag{3.44}$$

Substituting Eqs.(3.41-3.44) into Eq.(2.6), we have

$$C_m^{n+1} = \frac{1}{2}(2\beta + \alpha_m^n)C_{m-1}^n + (1 - 2\beta)C_m^n + \frac{1}{2}(2\beta - \alpha_m^n)C_{m+1}^n, \tag{3.45}$$

for all $1 \leq m < M$ and $0 \leq n < N$. For the left boundary condition, $m = 0$, the known value on the left boundary are approximated $C_{-1}^n = C_0^n$, we can see that

$$C_0^{n+1} = (1 + \frac{1}{2}\alpha_0^n - \beta)C_0^n + \frac{1}{2}(2\beta - \alpha_0^n)C_1^n. \tag{3.46}$$

Similarly, the right boundary condition, $m = M$, the known value on the left boundary are approximated $C_{M+1}^n = C_M^n$, we have

$$C_M^{n+1} = \frac{1}{2}(2\beta + \alpha_M^n)C_{M-1}^n + (1 - \frac{1}{2}\alpha_M^n - \beta)C_M^n. \tag{3.47}$$

The stability of FTCS scheme is [10]

$$\frac{(\alpha)^2}{2} \leq \beta \leq \frac{1}{2}. \tag{3.48}$$

3.2.6 The Upwind Explicit Formula

The upwind explicit scheme is considered by the following discretizations:

$$\frac{\partial C}{\partial t} \approx \frac{(C_m^{n+1} - C_m^n)}{\Delta t}, \tag{3.49}$$

$$\frac{\partial C}{\partial x} \approx \frac{(C_m^n - C_{m-1}^n)}{\Delta x}, \tag{3.50}$$

$$\frac{\partial^2 C}{\partial x^2} \approx \frac{(C_{m+1}^n - 2C_m^n + C_{m-1}^n)}{(\Delta x)^2}, \tag{3.51}$$

$$u \approx \widetilde{u}_m^n. \tag{3.52}$$

Substituting Eqs.(3.49-3.52) into Eq.(2.6), we have

$$C_m^{n+1} = (\beta + \alpha_m^n)C_{m-1}^n + (1 - 2\beta - \alpha_m^n)C_m^n + \beta C_{m+1}^n, \tag{3.53}$$

for all $1 \leq m < M$ and $0 \leq n < N$. For the left boundary condition, $m = 0$, the known value on the left boundary are approximated $C_{-1}^n = C_0^n$, we can see that

$$C_0^{n+1} = (1 - \beta)C_0^n + \beta C_1^n. \tag{3.54}$$

Similarly, the right boundary condition, $m = M$, the known value on the left boundary condition are approximated $C_{M+1}^n = C_M^n$, we have

$$C_M^{n+1} = (\beta + \alpha_M^n)C_{M-1}^n + (1 - \beta - \alpha_M^n)C_M^n. \tag{3.55}$$

The stability of upwind explicit scheme is [10]

$$\frac{\alpha^2 - \alpha}{2} \leq \beta \leq \frac{1 - \alpha}{2}. \tag{3.56}$$

3.2.7 The Lax-Wendroff Technique

The Lax-wendroff scheme is considered by the following discretizations:

$$\frac{\partial C}{\partial t} \approx \frac{(C_m^{n+1} - C_m^n)}{\Delta t}, \tag{3.57}$$

$$\frac{\partial C}{\partial x} \approx \alpha_m^n \frac{(C_m^n - C_{m-1}^n)}{\Delta x} + (1 - \alpha_m^n)\frac{(C_{m+1}^n - C_{m-1}^n)}{2\Delta x}, \tag{3.58}$$

$$\frac{\partial^2 C}{\partial x^2} \approx \frac{(C_{m+1}^n - 2C_m^n + C_{m-1}^n)}{(\Delta x)^2}, \tag{3.59}$$

$$u \approx \widetilde{u}_m^n. \tag{3.60}$$

Substituting Eqs.(3.57-3.60) into Eq.(2.6), we have

$$C_m^{n+1} = \frac{1}{2}(2\beta + \alpha_m^n + (\alpha_m^n)^2)C_{m-1}^n + (1 - 2\beta - (\alpha_m^n)^2)C_m^n + \frac{1}{2}(2\beta - \alpha_m^n + (\alpha_m^n)^2)C_{m+1}^n, \tag{3.61}$$

for all $1 \leq m < M$ and $0 \leq n < N$. For the left boundary condition, $m = 0$, the known value on the left boundary are approximated $C_{-1}^n = C_0^n$, we can see that

$$C_0^{n+1} = (1 - \beta + \frac{1}{2}\alpha_0^n - \frac{1}{2}(\alpha_0^n)^2)C_0^n + \frac{1}{2}(2\beta - \alpha_0^n + (\alpha_0^n)^2)C_1^n. \tag{3.62}$$

Similarly, the right boundary condition, $m = M$, the known value on the left boundary are approximated $C_{M+1}^n = C_M^n$, we have

$$C_M^{n+1} = \frac{1}{2}(2\beta - \alpha_M^n + (\alpha_M^n)^2)C_{M-1}^n + (1 - \beta - \frac{1}{2}\alpha_M^n - \frac{1}{2}(\alpha_M^n)^2)C_M^n. \tag{3.63}$$

The stability of upwind explicit scheme is [10]

$$0 < \beta \leq \frac{1 - \alpha^2}{2}. \tag{3.64}$$

4 Numerical Experiments

Suppose that the measurement of pollutant concentration C in a dam-break flow stream is considered. A stream is aligned with longitudinal distance, 2000 (m) total length. There is a dam-break which discharges waste water into the flooding area at middle point of the domain and the pollutant concentrations at the discharge point are assumed:

$$C(1000,t) = (1 - |\sin t|)\frac{T-t}{T} \text{ for all } 0 \leq t < T, \tag{4.1}$$

$$C_x(2000,t) = 0, \tag{4.2}$$

$$C(x,0) = \begin{cases} 1\ kg/m^3 & \text{if } x = 1000, \\ 0.1\ kg/m^3 & \text{if } 1000 < x \leq 2000. \end{cases} \tag{4.3}$$

The elevation and velocity of water are obtained by the dam-break model that we assume the initial and boundary conditions by several cases as below,
Case A: Dam-break on wet bed (pollutant discharging into the high flooding area)

$$h(x,0) = \begin{cases} 1\ m & \text{if } x = 1000, \\ 0.75\ m & \text{if } 1000 < x \leq 2000, \end{cases} \tag{4.4}$$

where $u(x,0) = 0$ for all $1000 \leq x \leq 2000$ and $u_x(0,t) = u_x(2000,t) = 0$.
Case B: Dam-break on wet bed (pollutant discharging into the medium flooding area)

$$h(x,0) = \begin{cases} 1\ m & \text{if } x = 1000, \\ 0.50\ m & \text{if } 1000 < x \leq 2000, \end{cases} \tag{4.5}$$

where $u(x,0) = 0$ for all $1000 \leq x \leq 2000$ and $u_x(0,t) = u_x(2000,t) = 0$.
Case C: Dam-break on wet bed (pollutant discharging into the low flooding area)

$$h(x,0) = \begin{cases} 1\ m & \text{if } x = 1000, \\ 0.25\ m & \text{if } 1000 < x \leq 2000, \end{cases} \tag{4.6}$$

where $u(x,0) = 0$ for all $1000 \leq x \leq 2000$ and $u_x(0,t) = u_x(2000,t) = 0$. The physical parameters of the polluted system are diffusion coefficient $D = 1.00(m^2/s)$. In the analysis conducted in this study, meshes the stream into 1000 elements with $\Delta x = 2$, and time increment is $0.1(s)$ with $\Delta t = 0.1$, characterizing a one-dimensional flow. Using the modified Lax-diffusive method Eq.(3.7) to obtain the velocity and elevation of water when sandbag-dike is destroyed. We can get the water velocity $u(x,t)$ on Tables 2, 4 and 6 in 3 cases of the high level flooding area, the medium level flooding area and the low level flooding area, respectively. We also get the water elevation $h(x,t)$ on Tables 1, 3 and 5, respectively. Next, it can be plug the approximate water velocity into their implicit and explicit methods such as Modified Siemieniuch-Gladwell method, BTCS method, Upwind implicit method, Crank-Nicolson method, FTCS method, Upwind explicit method, Lax-wendroff method. The approximation of pollutant concentrations C of several schemes are shown in Tables 7-13. The comparison of approximated pollutant concentrations of FTCS, Upwind explicit and Lax-wendroff with several dam-break cases A, B and C are shown in Fig. 5.

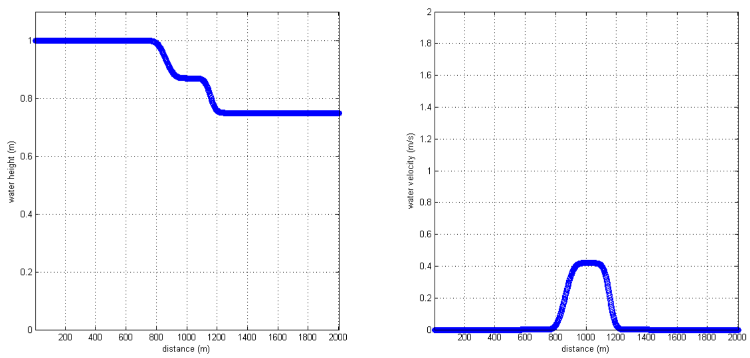


Figure 2: (a) The water elevation $h(x,t)(m)$ and (b) The water velocity $u(x,t)(m/sec)$ of case A (wet bed with depth ratio 0.75) at $t = 50$ sec

Table 1: The water elevation of water flow $h(x,t)m$ where $h_l = 1m$ and $h_l = 0.75m$ (Case A)

$t(sec), x(m)$	0	200	400	600	800	1000	1200	1400	1600	1800	2000
10	1.0000	1.0000	1.0000	1.0000	1.0000	0.8746	0.7500	0.7500	0.7500	0.7500	0.7500
20	1.0000	1.0000	1.0000	1.0000	1.0000	0.8710	0.7500	0.7500	0.7500	0.7500	0.7500
30	1.0000	1.0000	1.0000	1.0000	1.0000	0.8700	0.7500	0.7500	0.7500	0.7500	0.7500
40	1.0000	1.0000	1.0000	1.0000	0.9992	0.8698	0.7504	0.7500	0.7500	0.7500	0.7500
50	1.0000	1.0000	1.0000	1.0000	0.9904	0.8698	0.7571	0.7500	0.7500	0.7500	0.7500

Table 2: The water velocity of water flow $u(x,t)m/s$ where $h_l = 1m$ and $h_l = 0.75m$ (Case A)

$t(sec), x(m)$	0	200	400	600	800	1000	1200	1400	1600	1800	2000
10	0.0000	0.0000	0.0000	0.0000	0.0000	0.3856	0.0000	0.0000	0.0000	0.0000	0.0000
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.4134	0.0000	0.0000	0.0000	0.0000	0.0000
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.4185	0.0000	0.0000	0.0000	0.0000	0.0000
40	0.0000	0.0000	0.0000	0.0000	0.0026	0.4196	0.0013	0.0000	0.0000	0.0000	0.0000
50	0.0000	0.0000	0.0000	0.0000	0.0301	0.4198	0.0257	0.0000	0.0000	0.0000	0.0000

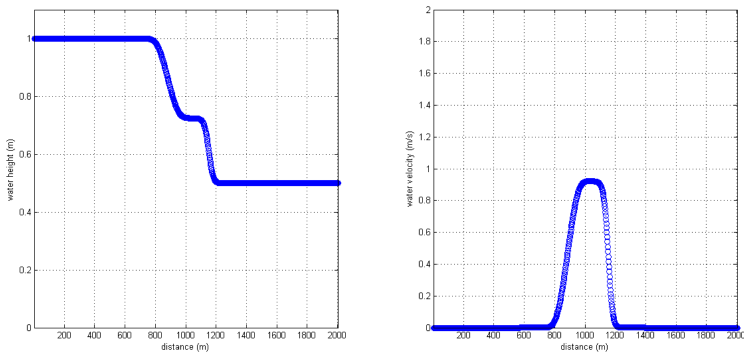


Figure 3: (a) The water elevation $h(x,t)(m)$ and (b) The water velocity $u(x,t)(m/sec)$ of case B (wet bed with depth ratio 0.5) at $t = 50$ sec

Table 3: The water elevation of water flow $h(x,t)m$ where $h_l = 1m$ and $h_r = 0.5m$ (Case B)

$t(sec), x(m)$	0	200	400	600	800	1000	1200	1400	1600	1800	2000
10	1.0000	1.0000	1.0000	1.0000	1.0000	0.7491	0.5000	0.5000	0.5000	0.5000	0.5000
20	1.0000	1.0000	1.0000	1.0000	1.0000	0.7353	0.5000	0.5000	0.5000	0.5000	0.5000
30	1.0000	1.0000	1.0000	1.0000	0.9999	0.7294	0.5000	0.5000	0.5000	0.5000	0.5000
40	1.0000	1.0000	1.0000	1.0000	0.9986	0.7269	0.5001	0.5000	0.5000	0.5000	0.5000
50	1.0000	1.0000	1.0000	1.0000	0.9852	0.7259	0.5047	0.5000	0.5000	0.5000	0.5000

Table 4: The water velocity of water flow $u(x,t)m/s$ where $h_l = 1m$ and $h_r = 0.5m$ (Case B)

$t(sec), x(m)$	0	200	400	600	800	1000	1200	1400	1600	1800	2000
10	0.0000	0.0000	0.0000	0.0000	0.0000	0.7945	0.0000	0.0000	0.0000	0.0000	0.0000
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.8762	0.0000	0.0000	0.0000	0.0000	0.0000
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.9027	0.0000	0.0000	0.0000	0.0000	0.0000
40	0.0000	0.0000	0.0000	0.0000	0.0044	0.9138	0.0006	0.0000	0.0000	0.0000	0.0000
50	0.0000	0.0000	0.0000	0.0000	0.0465	0.9189	0.0207	0.0000	0.0000	0.0000	0.0000

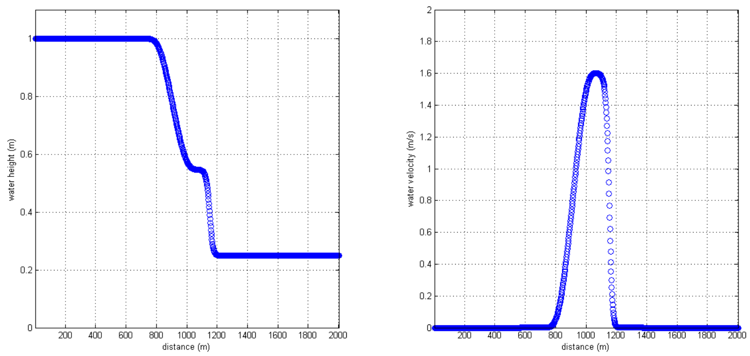


Figure 4: (a) The water elevation $h(x,t)(m)$ and (b) The water velocity $u(x,t)(m/sec)$ of case C (wet bed with depth ratio 0.25) at $t = 50$ sec

Table 5: The water elevation of water flow $h(x,t)m$ where $h_l = 1m$ and $h_r = 0.25m$ (Case C)

$t(sec), x(m)$	0	200	400	600	800	1000	1200	1400	1600	1800	2000
10	1.0000	1.0000	1.0000	1.0000	1.0000	0.6307	0.2500	0.2500	0.2500	0.2500	0.2500
20	1.0000	1.0000	1.0000	1.0000	1.0000	0.6049	0.2500	0.2500	0.2500	0.2500	0.2500
30	1.0000	1.0000	1.0000	1.0000	1.0000	0.5901	0.2500	0.2500	0.2500	0.2500	0.2500
40	1.0000	1.0000	1.0000	1.0000	0.9982	0.5808	0.2500	0.2500	0.2500	0.2500	0.2500
50	1.0000	1.0000	1.0000	1.0000	0.9821	0.5744	0.2512	0.2500	0.2500	0.2500	0.2500

Table 6: The water velocity of water flow $u(x,t)m/s$ where $h_l = 1m$ and $h_r = 0.25m$ (Case C)

$t(sec), x(m)$	0	200	400	600	800	1000	1200	1400	1600	1800	2000
10	0.0000	0.0000	0.0000	0.0000	0.0000	1.1995	0.0000	0.0000	0.0000	0.0000	0.0000
20	0.0000	0.0000	0.0000	0.0000	0.0000	1.3517	0.0000	0.0000	0.0000	0.0000	0.0000
30	0.0000	0.0000	0.0000	0.0000	0.0000	1.4222	0.0000	0.0000	0.0000	0.0000	0.0000
40	0.0000	0.0000	0.0000	0.0000	0.0055	1.4655	0.0001	0.0000	0.0000	0.0000	0.0000
50	0.0000	0.0000	0.0000	0.0000	0.0561	1.4952	0.0076	0.0000	0.0000	0.0000	0.0000

Table 7: The pollutant concentration $C(x,t)(Kg/m^3)$ by Modified Siemieniuch-Gladwell schemes (Case A)

$t(sec), x(m)$	1000	1100	1200	1300	1400	1500	1600	1700	1800	1900	2000
300	0.6411	0.6838	0.1012	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000
600	0.3663	0.4924	0.6715	0.1630	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000
900	0.1741	0.2734	0.4426	0.6334	0.3129	0.1017	0.1000	0.1000	0.1000	0.1000	0.1000
1200	0.0565	0.1236	0.2382	0.3958	0.5863	0.4518	0.1188	0.1000	0.1000	0.1000	0.1000

Table 8: The pollutant concentration $C(x,t)(Kg/m^3)$ by BTCS schemes (Case A)

$t(sec), x(m)$	1000	1100	1200	1300	1400	1500	1600	1700	1800	1900	2000
300	0.6411	0.3926	0.1006	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000
600	0.3663	0.2795	0.3832	0.1320	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000
900	0.1741	0.1552	0.2513	0.3606	0.2069	0.1009	0.1000	0.1000	0.1000	0.1000	0.1000
1200	0.0565	0.0702	0.1352	0.2247	0.3334	0.2755	0.1096	0.1000	0.1000	0.1000	0.1000

Table 9: The pollutant concentration $C(x,t)(Kg/m^3)$ by Upwind implicit schemes (Case A)

$t(sec), x(m)$	1000	1100	1200	1300	1400	1500	1600	1700	1800	1900	2000
300	0.6411	0.7330	0.1056	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000
600	0.3663	0.5540	0.7232	0.2052	0.1002	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000
900	0.1741	0.3079	0.4984	0.6885	0.3632	0.1072	0.1000	0.1000	0.1000	0.1000	0.1000
1200	0.0565	0.1394	0.2688	0.4463	0.6430	0.4899	0.1422	0.1004	0.1000	0.1000	0.1000

Table 10: The pollutant concentration $C(x,t)(Kg/m^3)$ by Crank-Nicolson schemes (Case A)

$t(sec), x(m)$	1000	1100	1200	1300	1400	1500	1600	1700	1800	1900	2000
300	0.6411	0.5044	0.1008	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000
600	0.3663	0.3613	0.4940	0.1435	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000
900	0.1741	0.2007	0.3248	0.4655	0.2472	0.1011	0.1000	0.1000	0.1000	0.1000	0.1000
1200	0.0565	0.0907	0.1748	0.2904	0.4306	0.3430	0.1130	0.1000	0.1000	0.1000	0.1000

Table 11: The pollutant concentration $C(x,t)(Kg/m^3)$ by FTCS schemes (Case A)

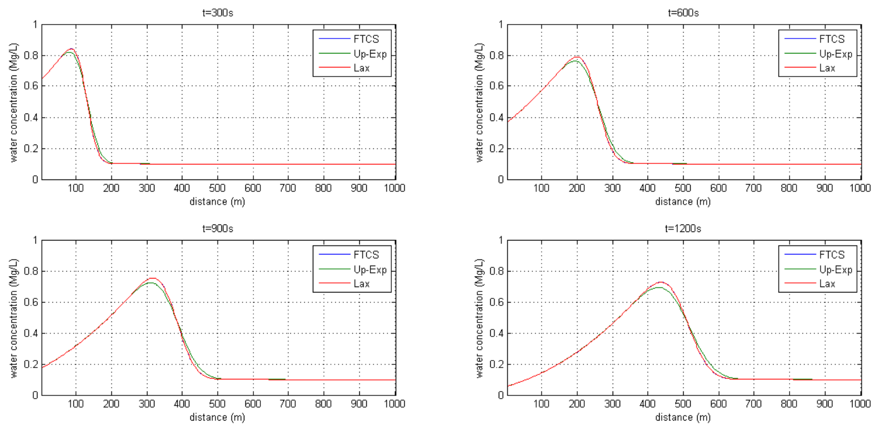
$t(sec), x(m)$	1000	1100	1200	1300	1400	1500	1600	1700	1800	1900	2000
300	0.6461	0.7907	0.1010	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000
600	0.3701	0.5836	0.7877	0.1672	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000
900	0.1766	0.3249	0.5248	0.7469	0.3418	0.1016	0.1000	0.1000	0.1000	0.1000	0.1000
1200	0.0579	0.1475	0.2832	0.4695	0.6930	0.5115	0.1200	0.1000	0.1000	0.1000	0.1000

Table 12: The pollutant concentration $C(x,t)(Kg/m^3)$ by Upwind explicit schemes (Case A)

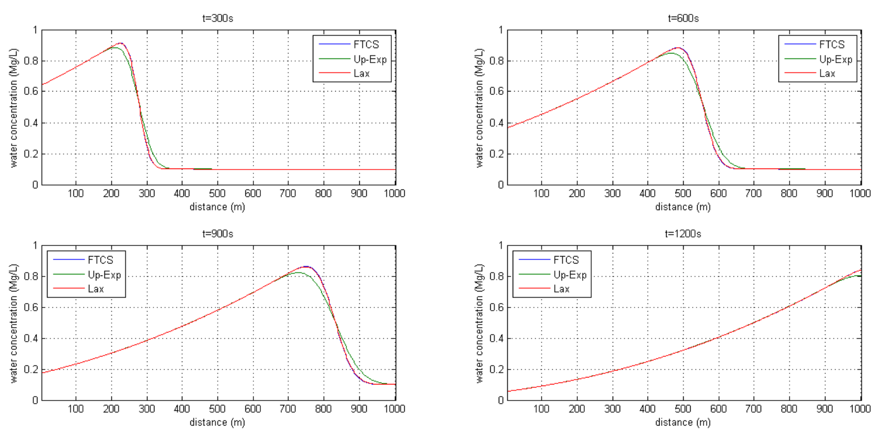
$t(sec), x(m)$	1000	1100	1200	1300	1400	1500	1600	1700	1800	1900	2000
300	0.6461	0.7571	0.1046	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000
600	0.3701	0.5841	0.7553	0.2018	0.1001	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000
900	0.1766	0.3254	0.5257	0.7220	0.3665	0.1065	0.1000	0.1000	0.1000	0.1000	0.1000
1200	0.0579	0.1480	0.2842	0.4709	0.6757	0.5027	0.1407	0.1003	0.1000	0.1000	0.1000

Table 13: The pollutant concentration $C(x,t)(Kg/m^3)$ by Lax-wendroff schemes (Case A)

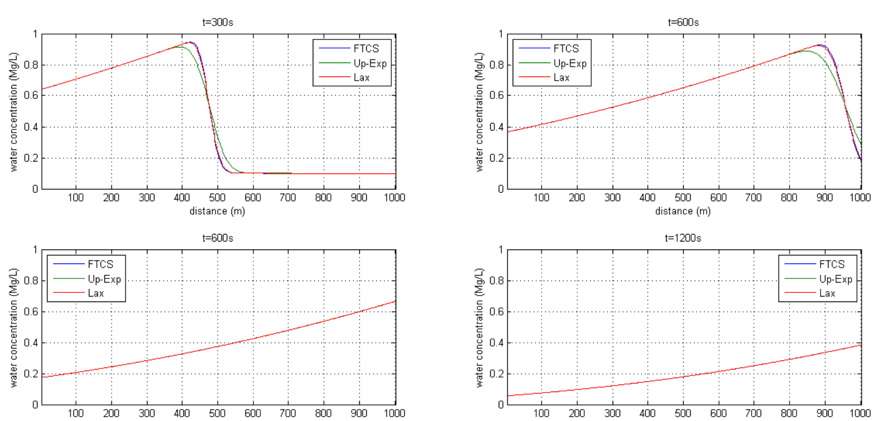
$t(sec), x(m)$	1000	1100	1200	1300	1400	1500	1600	1700	1800	1900	2000
300	0.6461	0.7898	0.1011	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000
600	0.3701	0.5836	0.7870	0.1680	0.1001	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000
900	0.1766	0.3249	0.5248	0.7463	0.3424	0.1017	0.1000	0.1000	0.1000	0.1000	0.1000
1200	0.0579	0.1475	0.2832	0.4695	0.6926	0.5113	0.1204	0.1000	0.1000	0.1000	0.1000



(a) case A (depth ratio 0.75)



(b) case B (depth ratio 0.5)



(c) case C (depth ratio 0.25)

Figure 5: The comparison of pollutant concentration for all cases with $\Delta x = 2, \Delta t = 0.1$ at difference time by explicit methods.

5 Discussion

The elevation and velocity of water current are obtained by a revised Lax-diffusive method. The case C of sandbag-dike failure gives the highest flow velocity. The cases A-B of sandbag-dike failure give low velocity. The approximation of the pollutant concentrations of the implicit and explicit methods are shown in Tables 7-13.

6 Conclusion

If the villagers have received flood for longtime, the water pollutant have to be increase. The residents want to drain the water to the other areas by destroying the sandbag-dike. The other villages that never encounter flood going to receive a polluted water. We can see that the pollutant concentration level on the flooding area is not to high with the passing of time. The real-world problems require a small amount of time interval in obtaining accurate solutions. Unfortunately, the analytical solutions of the dam-break model could not be found over the entire domain. This also implies that the analytical solutions of dispersion model could not work out at any point on the entire domain as well. We propose a revised Lax-diffusive scheme by editing a simple modification to the traditional Lax-diffusive scheme. The FTCS method is limited by restriction of the stability condition. Then FTCS is not flexible in the real-world situation. The implicit schemes shows excessive dispersion effects for large time and space step lengths, significantly decreasing the efficiency of the implicit schemes. In addition implicit methods still generate a lot of large systems of linear equations. The Upwind explicit and Lax-wendroff schemes are economical to use. The proposed method show a good agreement in accuracy, the implicit schemes becomes less efficient than the explicit schemes. In this paper, the dispersion model and the dam-break model can be compounded to approximate the pollutant concentration in a stream when the current reflecting water in the stream is not uniform since the sandbag-dike becomes failure. In this paper, the technique developed, the response of the stream to the two different external inputs: the elevation of water and the pollutant concentration at the discharge point, can be obtained. The both of the implicit methods and the explicit methods can be used in the dispersion model since the scheme is very simple to implement. By the explicit finite difference formulations, we obtain that the proposed technique is applicable and economical to be used in the real-world problem due to its simplicity to program and the straight forwardness of the implementation. It is also possible to find tentative better locations and better periods of time of the different discharge points to a stream.

Acknowledgements : The authors would like to thank the reviewers for their careful reading and constructive comments. This research is supported by the Centre of Excellence in Mathematics, the Commission on Higher Education, Thailand.

References

[1] P. Garcia-Navarro, A. Fras, I. Villanueva, Dam-break flow simulation: some results for one dimension models of real cases, Journal of Hydrology 216 (1999) 227-247.

[2] G. Gottardi, M. Venutelli, Central scheme for two-dimensional dam-break flow simulation, Advances in Water Resources 27 (2004) 259-268.

[3] F. Benkhaldoun, M. Seaid, A simple finite volume method for the shallow water equations, Journal of Computational and Applied Mathematics 234 (2010) 58-72.

[4] M. Pirotton, S. Erpicum, B.J. Dewals, P. Archambeau, Dam break flow computation based on an efficient flux vector splitting, Journal of Computational and Applied Mathematics 234 (2010) 2143-2151.

[5] A. Baghlani, Simulation of dam-break problem by a robust flux-vector splitting approach in Cartesian grid, Scientia Iranica 18 (5) (2010) 1061-1068.

[6] H.M. Kao, T.J. Chang, K.H. Chang, M.H. Hsu, Numerical simulation of shallow-water dam break flows in open channels using smoothed particle hydrodynamics, Journal of Hydrology 408 (2011) 78-90.

[7] R. Touma, S. Khankan, Well-balance unstaggered central schemes for one and two-dimensional shallow water equation system, Applied Mathematics and Computation 218 (2012) 5948-5960.

[8] C. Berthon, F. Foucher, Efficient well-balanced hydrostatic schemes for shallow-water equations, Journal of Computational Physics 231 (2012) 993-5015.

[9] G. Akbari, B. Firoozi, Implicit and Explicit Numerical Simulation of Saint-Venent Equations for Simulating Flood Wave in Natural Rivers, 5th National Congress on Civil Engineering, Mashhad, Iran, 2010.

[10] M. Dehghan, Weighted finite difference techniques for the one-dimensional advection-diffusion equation, Applied Mathematics and Computation 147 (2004) 307-319.

(Received 25 September 2016)
(Accepted 22 February 2018)