A compact fourth-order finite difference scheme for the improved Boussinesq equation with damping terms

Fuqiang Lu, Zhiyao Song and Zhuo Zhang

Key Laboratory of Virtual Geographic Environment (Nanjing Normal University), Ministry of Education, Nanjing, 210023, China
State Key Laboratory Cultivation Base of Geographical Environment Evolution (Jiangsu Province), Nanjing, 210023, China
Jiangsu Center for Collaborative Innovation in Geographical Information Resource Development and Application, Nanjing, 210023, China

Emails: lufuqiang2000@163.com, Zhiyaosong@sohu.com, mercury1214@126.com

Abstract

In this paper, a compact finite difference method is presented for solving the initial boundary value problems for the improved Boussinesq equation with damping terms. The fourth-order equation can be transformed into a first-order ordinary differential system, and then, the classical Padé approximation is used to discretize spatial derivative in the non-linear partial differential equations. The resulting coefficient matrix for the semi-discrete scheme is tri-diagonal and can be solved efficiently. In order to maintain the same order of convergence, the classical fourth-order Runge-Kutta method is the preferred method for explicit time integration. Soliton-type solutions are used to evaluate the accuracy of the method, and various numerical experiments are designed to test the different effects of the damping terms.


Key words: Compact finite difference method, Improved Boussinesq equation, Stokes damping, Hydrodynamic damping, Runge-Kutta method.

1. Introduction

The Boussinesq equation was first presented by Joseph Boussinesq in 1873, which is a fourth-order nonlinear PDE belonging to the Kdv family. This equation mainly describes the propagation of long waves on the free surface of shallow water under gravity conditions, which has been widely used in math-physical field related to nonlinear wave phenomena, such as ion-sound in plasma and, nonlinear lattice waves [1, 2]. In general, the equation has the following form:

$$u_{tt}(x,t) = u_{xx}(x,t) + qu_{xxxx}(x,t) + (u^2(x,t))_{xx},$$

where \( q = \pm 1 \). When \( q = 1 \), it is the so-called bad Boussinesq equation and is ill-posed; whereas when \( q = -1 \), it is called a good Boussinesq equation and is well-posed. Many well known methods, such as the inverse scattering transform method, bilinear formalism, and the Bäcklund transformation method, can be used to handle the completely integrable Boussinesq equation [3]. Bogolubsky [4, 5] showed that the bad Boussinesq equation describes a spurious
instability of short wavelengths and its Cauchy problem is incorrect. Consequently Makhankov [2] demonstrated that the bad Boussinesq equation could be approximated using the improved Boussinesq equation (IBq for short) by replacing the term $u_{xxxx}$ with $u_{xxtt}$:

$$u_{tt} - u_{xx} - (u^2)_{xx} - u_{xxtt} = 0.$$  

(1.1)

The Eq. (1.1) arises in acoustic waves on elastic rods with circular cross-sections when transverse motion and nonlinearity are examined [6], moreover, the improved Boussinesq equation is used to describe wave propagation at right angles to the magnetic field and is more suitable for numerical simulation than the bad Boussinesq equation. To some extent, it has been explored theoretically by several authors. Abdou [7] obtained the generalized solution and period solution of (1.1) using the Exp-function method. The sine or cosine ansatz method was utilized by Wazwaz [8] to find many compact and non-compact solutions of (1.1) with variants. In spite of these special soliton-type solutions, it is difficult to find an analytical solution satisfying a particular choice of initial conditions in most situations. Therefore, the main approach depends on solving the problem numerically.

The finite difference method is popular owing to its simplicity and ease of manipulation. Bogolubsky [5], Iskandar and Jain [9] were the first to study (1.1) using a three-level implicit nonlinear scheme with second-order accuracy. Later, Zoheiry [10] provided a three-level iteration relation based on the implicit compact difference scheme to improve the accuracy in space. Bratsos [11] used the typical second-order difference method to reduce (1.1) to a system of ordinary differential equations and developed a predict-correction scheme based on two different proper Padé approximations for the same matrix index term. The stability analysis was also presented. The scheme is complicated and the computational effort required is considerable.

The finite element method is also widely used for its geometric flexibility. Dursun and Irk [12] proposed two difference schemes and two quintic B-spline finite element collocation methods based on the second and third-order time discretization methods, these schemes were compared with each other subsequently. Lin and Wu [13] presented a finite element scheme based on the linear B spline. Inc and Evans [14] solved (1.1) using the domain decomposition method, which is a series expansion method, and the solution is expressed as a convergent series. An approximation for the solution is obtained by truncating the series after retaining a sufficient number of terms. Indeed, it is more accurate than the finite element method but has a great computational load.

The finite volume element method has widespread use due to its capability for local conservation. Zhang and Lu [15] developed a quadratic finite volume element method for (1.1). Wang and Zhang [16] applied this method to the stochastic damped Improved Boussinesq equation. This method seems to yield better results than the ordinary central finite difference method, but has the same second-order accuracy.

In this paper, we consider the damped Improved Boussinesq equation of the following form:

$$u_{tt} - u_{xx} - (u^2)_{xx} - u_{xxtt} = -\tau_s u_t + \tau_h u_{txx},$$  

(1.2)

where in the right-hand terms, $\tau_s u_t, \tau_h u_{txx}$ denote the effect of Stokes damping and hydrodynamic damping [17,18] respectively, and the coefficients should satisfy $\tau_s \geq 0$ and $\tau_h \geq 0$. It is clear that the damped IBq (1.2) reduces to the IBq (1.1) when $\tau_s = \tau_h = 0$.

The compact finite difference scheme [21,22] has enjoyed great popularity in computational dynamics and electromagnetic and computational acoustics due to its high accuracy, compact stencils, and high resolution compared with the ordinary central finite difference scheme. We
can use a compact difference method to solve Eq. (1.2) in order to validate our method and study the effect of the damping terms on the dynamical evolution of the soliton-type traveling wave solution. This paper is organized as follows. In Section 2, it is necessary to transform the fourth-order equation into a first-order differential system. Then, a fourth-order compact finite difference approximation is used to discretize the space derivative terms, and a first-order ODE involving only time derivative terms is derived. In order to preserve the fourth order of the leading truncation error, a classical fourth-order Runge-Kutta method is used for explicit time stepping. Moreover, a linear stability analysis is presented. In Section 3.1, the compact finite difference method is used to simulate the propagation of solitary waves with the purpose of evaluating the accuracy and efficiency. Then, we test the effectiveness and validity of the method using several experiments. The influence of damping terms on (1.1) is provided in Section 4. The last section contains summary and conclusion.

2. The Approximation in Space and Time

Consider the damped IBq (1.2) with the following initial value:

\[ u(x,0) = f(x), \quad u_t(x,0) = g(x), \quad x \in [a, b], \quad (2.1) \]

and Dirichlet boundary conditions:

\[ u(a,t) = u(b,t) = 0, \quad t \geq t_0, \quad (2.2) \]

or Neumann boundary conditions:

\[ u_x(a,t) = u_x(b,t) = 0, \quad t \geq t_0. \quad (2.3) \]

Let \( v = u_t \). Eq. (1.2) is then transformed into the following first-order differential system:

\[ \begin{cases} u_t = v, \\ u_t - v_{txx} - u_{xx} - (u^2)_{xx} = -\tau_s v + \tau_h v_{xx}. \end{cases} \quad (2.4) \]

2.1. A compact finite difference for space derivatives

Assuming that a smooth function \( f(x) \) is defined on a uniform grid of interval \( \Delta x \), a fourth-order compact finite difference scheme [19,20] can be used to approximate the second derivative

\[ \frac{1}{12} \left( \frac{\partial^2 f}{\partial x^2} \right)_{i-1} + 10 \frac{\partial^2 f}{\partial x^2} \bigg|_{i} + \frac{\partial^2 f}{\partial x^2} \bigg|_{i+1} = \frac{f_{i-1} - 2f_i + f_{i+1}}{(\Delta x)^2} + O(\Delta x)^4, \]

\[ \frac{\partial^2 f}{\partial x^2} \bigg|_{i} + \frac{(\Delta x)^2}{12} \frac{\partial^2 f}{\partial x^2} \bigg|_{i-1} - 2 \frac{\partial^2 f}{\partial x^2} \bigg|_{i} + \frac{\partial^2 f}{\partial x^2} \bigg|_{i+1} = \frac{f_{i-1} - 2f_i + f_{i+1}}{(\Delta x)^2} + O(\Delta x)^4, \]

\[ \left( 1 + \frac{(\Delta x)^2}{12} \delta_x^2 \right) \frac{\partial^2 f}{\partial x^2} \bigg|_{i} = \frac{f_{i-1} - 2f_i + f_{i+1}}{(\Delta x)^2} + O(\Delta x)^4, \]

\[ \frac{\partial^2 f}{\partial x^2} \bigg|_{i} = \left( 1 + \frac{(\Delta x)^2}{12} \delta_x^2 \right)^{-1} \delta_x^2 f + O(\Delta x)^4. \quad (2.5) \]

where we use a second-order central finite difference operator:

\[ \delta_x^2 f = \frac{f_{i-1} - 2f_i + f_{i+1}}{(\Delta x)^2}. \]
As shown in Fig. 2.1, the region of interest $[a, b]$ can be composed into a uniform grid with nodes as follows:

$$a = x_0 < x_1 < x_2 < \cdots < x_{N-1} < x_N = b.$$  

![Fig. 2.1. Grid nodes with a uniform interval, where $x_{-1}, x_{N+1}$ are two ghost nodes.](image)

Let $u_t = \dot{u}$, $v_t = \dot{v}$. Applying the compact finite difference scheme (2.5) to approximate the second sub-equation in the system (2.4) gives

$$\dot{v} + \tau_s v = \left(1 + \frac{(\Delta x)^2}{12} \delta_x^2\right)^{-1} \delta_x^2 (\dot{v} + u + u^2 + \tau_h v) + (\Delta x)^4,$$  

(2.6)

$$\left(1 + \frac{(\Delta x)^2}{12} \delta_x^2\right)(\dot{v} + \tau_s v) = \delta_x^2 (\dot{v} + u + u^2 + \tau_h v) + (\Delta x)^4.$$  

(2.7)

Omitting the leading truncation error yields

$$\frac{(\Delta x)^2}{12} \left((\dot{v}_{i-1} + 10 \dot{v}_i + \dot{v}_{i+1}) + \tau_s (v_{i-1} + 10 v_i + v_{i+1})\right)$$

$$\approx (\dot{v}_{i-1} - 2 \dot{v}_i + \dot{v}_{i+1}) + \tau_h (v_{i-1} - 2 v_i + v_{i+1})$$

$$+ (u_{i-1} - 2 u_i + u_{i+1}) + \left((u_{i-1})^2 - 2(u_i)^2 + (u_{i+1})^2\right).$$  

(2.8)

which leads to

$$\frac{(\Delta x)^2}{12} \left((\dot{v}_{i-1} + 10 \dot{v}_i + \dot{v}_{i+1}) - (\dot{v}_{i-1} - 2 \dot{v}_i + \dot{v}_{i+1})\right)$$

$$= (u_{i-1} - 2 u_i + u_{i+1}) + \left((u_{i-1})^2 - 2(u_i)^2 + (u_{i+1})^2\right)$$

$$+ \tau_h (v_{i-1} - 2 v_i + v_{i+1}) - \tau_s (\Delta x)^2 \left(v_{i-1} + 10 v_i + v_{i+1}\right).$$  

(2.9)

Note that the compact scheme (2.9) is only valid at internal nodes. In most cases, the accuracy would be degenerated at the boundary nodes due to insufficient information. We now consider (2.9) with Dirichlet boundary conditions (2.2) and Neumann boundary conditions (2.3) separately.

### 2.2. Approximation of Dirichlet boundary conditions

We consider (2.4) with Dirichlet boundary conditions (2.2). In this case, all unknown solutions lie at internal nodes. If we write Eq. (2.9) at grid points $x_i, i = 1, ..., N - 1$, then we obtain a nonlinear first-order system of ordinary differential equations as follows

$$\dot{U} = V,$$  

(2.10a)

$$\dot{V} = ((\Delta x)^2 A - B)^{-1} [B(U + U \ast U) + (\tau_h B - \tau_s (\Delta x)^2 A)V].$$  

(2.10b)
where
\[ \dot{U} = [\dot{u}_1, \dot{u}_2, \ldots, \dot{u}_{N-1}]^T, \quad \dot{V} = [\dot{v}_1, \dot{v}_2, \ldots, \dot{v}_{N-1}]^T. \]

and the matrix \( A, B \) are \((N-1) \times (N-1)\) tri-diagonal of the form
\[
\begin{bmatrix}
10 & 1 & & & \\
1 & 10 & 1 & & \\
& \ddots & \ddots & \ddots & \\
& & 1 & 10 & 1
\end{bmatrix}, \quad \begin{bmatrix}
-2 & 1 & & & \\
1 & -2 & 1 & & \\
& \ddots & \ddots & \ddots & \\
& & 1 & -2 & 1
\end{bmatrix}.
\] (2.11)

### 2.3. Approximation of Neumann boundary conditions

We consider (2.4) with Neumann boundary conditions (2.3). Similar to the method proposed in [23, 24], we add two ghost values \( u(x_{-1}, t) \) and \( u(x_{N+1}, t) \) by applying a second-order approximation to the first-order partial derivatives in Eq. (2.3):
\[
\begin{align*}
\frac{u(x_{-1}, t) - u(x_{1}, t)}{2\Delta x} &= O(\Delta x^2), \\
\frac{u(x_{N+1}, t) - u(x_{N-1}, t)}{2\Delta x} &= O(\Delta x^2).
\end{align*}
\] (2.12a, 2.12b)

Applying Neumann boundary conditions (2.3) and discarding the leading truncation error yields
\[
\begin{align*}
\left. u(x, t) \right|_{x=x_{-1}} &= u(x, t) \big|_{x=x_1} \\
\left. u(x, t) \right|_{x=x_{N+1}} &= u(x, t) \big|_{x=x_{N-1}}
\end{align*}
\] (2.13a, 2.13b)

Taking partial derivatives with respect to variable \( t \) in the above two sub-equations gives
\[
\begin{align*}
\frac{\partial u(x, t)}{\partial t} \big|_{x=x_{-1}} &= \frac{\partial u(x, t)}{\partial t} \big|_{x=x_1}, \\
\frac{\partial^2 u(x, t)}{\partial t^2} \big|_{x=x_{-1}} &= \frac{\partial^2 u(x, t)}{\partial t^2} \big|_{x=x_1},
\end{align*}
\] (2.14)

Taking partial derivatives with respect to variable \( x \) in the above two sub-equations gives
\[
\begin{align*}
\frac{\partial^2 u(x, t)}{\partial t^2} \big|_{x=x_{-1}} &= \frac{\partial^2 u(x, t)}{\partial t^2} \big|_{x=x_1},
\end{align*}
\] (2.15)

When \( i = 0 \), Eq. (2.9) becomes
\[
\frac{(\Delta x)^2}{12}(\ddot{v}_{-1} + 10\ddot{v}_0 + \ddot{v}_1) - (\dddot{v}_{-1} - 2\dddot{v}_0 + \dddot{v}_1)
= (u_{-1} - 2u_0 + u_1) + \left( (u_{-1})^2 - 2(u_0)^2 + (u_1)^2 \right)
+ \tau_s (v_{-1} - 2v_0 + v_1) - \tau_s \frac{(\Delta x)^2}{12} (v_{-1} + 10v_0 + v_1).
\] (2.16)

Inserting (2.14) into (2.16) yields
\[
\begin{align*}
\left( \frac{(\Delta x)^2}{12} - 2\dddot{v}_0 + \dddot{v}_1 \right)
= (-2u_0 + 2u_1) + \left( -2(u_0)^2 + 2(u_1)^2 \right)
+ \tau_s (-2v_0 + 2v_1) - \tau_s \frac{(\Delta x)^2}{12} (10v_0 + 2v_1).
\end{align*}
\] (2.17)
Similarly, when \( i = N \), we get
\[
\frac{(\Delta x)^2}{12} (2\hat{v}_{N-1} + 10\hat{v}_N - 2\hat{v}_N) = (2u_{N-1} - 2u_N) + \left[ 2(u_{N-1})^2 - 2(u_N)^2 \right] + \tau_s (\Delta x)^2 \frac{(\Delta x)^2}{12} (2v_{N-1} + 10v_N). \tag{2.18}
\]
If we note:
\[
\hat{U} = [\hat{u}_0, \hat{u}_1, \ldots, \hat{u}_N]^T, \quad \hat{V} = [\hat{v}_0, \hat{v}_1, \ldots, \hat{v}_N]^T.
\]
In matrix notation, the above system can be rewritten as:
\[
\hat{U} = \mathbf{V}, \quad \hat{V} = ((\Delta x)^2 \mathbf{A} - \mathbf{B})^{-1} [\mathbf{B}(\mathbf{U} + \mathbf{U} \cdot \mathbf{U}) + (\tau_s \mathbf{B} - \tau_s (\Delta x)^2 \mathbf{A}) \mathbf{V}],
\]
where matrices \( \mathbf{A}, \mathbf{B} \) are \((N+1) \times (N+1)\) tri-diagonal of the form (2.11).

Remark 2.1. In the case of the Neumann boundary conditions, the compact finite difference method degenerates to second-order accuracy at the boundary grid nodes. Without special instructions, (2.9) with Dirichlet boundary conditions would be solved for all numerical simulations.

2.4. Time discretization

Many well-known explicit methods can be used to solve the above first-order system, in order to preserve the same order of leading truncation errors in space, here the classical fourth-order Runge-Kutta (RK4 for short) method is used for explicit time integration, and the algorithms are given as follows:

\[
\begin{align*}
K_{u1} &= V^k; \\
K_{v1} &= C \cdot (U^k + (U^k \cdot U^k)) + D \cdot V^k; \\
K_{u2} &= K_{u1} + \Delta t/2 \cdot K_{v1}; \\
K_{v2} &= C \cdot ((U^k + \Delta t/2 \cdot K_{u1}) + (U^k + \Delta t/2 \cdot K_{u1}) \cdot (U^k + \Delta t/2 \cdot K_{u1})) + D \cdot K_{u2}; \\
K_{u3} &= K_{u1} + \Delta t/2 \cdot K_{v2}; \\
K_{v3} &= C \cdot ((U^k + \Delta t/2 \cdot K_{u2}) + (U^k + \Delta t/2 \cdot K_{u2}) \cdot (U^k + \Delta t/2 \cdot K_{u2})) + D \cdot K_{u3}; \\
K_{u4} &= K_{u1} + \Delta t \cdot K_{v3}; \\
K_{v4} &= C \cdot ((U^k + \Delta t \cdot K_{u3}) + (U^k + \Delta t \cdot K_{u3}) \cdot (U^k + \Delta t \cdot K_{u3})) + D \cdot K_{u4}; \\
U^{n+1} &= U^n + \Delta t/6 \cdot (K_{u1} + 2 \cdot K_{u2} + 2 \cdot K_{u3} + K_{u4}); \\
V^{n+1} &= V^n + \Delta t/6 \cdot (K_{v1} + 2 \cdot K_{v2} + 2 \cdot K_{v3} + K_{v4}),
\end{align*}
\]
where matrix
\[
C = ((\Delta x)^2 \mathbf{A} - \mathbf{B})^{-1} \cdot \mathbf{B}, \quad D = ((\Delta x)^2 \mathbf{A} - \mathbf{B})^{-1} (\tau_s \mathbf{B} - \tau_s (\Delta x)^2 \mathbf{A}),
\]
\( \Delta t \) is the time step, and \( U^k, V^k \) are the values of \( U, V \) at time level \( k \) respectively. It is natural that the initial conditions are approximated as:
\[
\begin{align*}
U^0 &= [u(x_0, 0), u(x_1, 0), \ldots, u(x_N, 0)], \\
V^0 &= [v_t(x_0, 0), v_t(x_1, 0), \ldots, v_t(x_N, 0)].
\end{align*}
\]
2.5. Linear stability analysis

Undoubtedly, the energy method is a better choice for stability analysis of the proposed discrete scheme if the considered equation is nonlinear. While the energy method has a drawback, that is, for each new problem, it requires fresh insight to construct an appropriate energy and to show that the proposed discrete scheme preserves a bound on that energy. Moreover, now the solved equation has a kind of blow-up solution in some cases even if the given initial conditions are smooth and have a finite bound, in such case, when the time evolves toward some fixed time, the solution would approach to infinity at some point in the computation domain. Therefore, it is intractable to construct an appropriate energy and the proposed discrete scheme preserves a bound on that energy.

As an alternative method, the eigenvalues of the matrix would be used for a linear stability analysis of the presented compact scheme. The solved equation is nonlinear, so the problem is linearized firstly, and then the eigenvalues of the matrix can be used to obtain the approximate stable conditions.

The two damping terms denote smooth effects for the solution under some desirable conditions. If the compact scheme for IBq is stable, then the scheme for IBq with damping terms is also stable. For the sake of convenience, we set the parameters that control the damping terms \( \tau_s = \tau_h = 0 \).

**Lemma 2.1.** ([25]) Consider RK4 (classical fourth-order Runge-Kutta) method to solve the following first-order equation:

\[
\begin{align*}
\frac{\partial u}{\partial t} &= \lambda u, \quad t \geq 0, \\
u(0) &= u_0,
\end{align*}
\]

where \( \lambda \) is a complex number. If the RK4 method is stable, then a necessary condition is

\[|\lambda \Delta t| \leq 2\sqrt{2}.\]

**Lemma 2.2.** ([26]) If the \( N \)th-order matrix \( A \) is tri-diagonal of the form

\[
A = \begin{bmatrix}
a & b \\
c & a & b \\
& \ddots & \ddots & \ddots \\
& & c & a & b \\
& & & c & a
\end{bmatrix}
\]

where elements \( a, b, c \) are real and \( bc > 0 \), then the right eigenvalues of \( A \) are given by

\[
\lambda_s = a + 2b \sqrt{\frac{c}{b}} \cos \left( \frac{k\pi}{N+1} \right), \quad k = 1, \ldots, N.
\]

**Theorem 2.1.** If the RK4 method is utilized for explicit time stepping of the first-order equation (2.10), then a necessary condition for the stability of the proposed method is

\[
\Delta t \leq \frac{\sqrt{2}}{3(1 + M)} (\Delta x)^2.
\]
Proof. Only the second sub-equation in system (2.10) is considered. First, we linearize the nonlinear term in the second sub-equation in system (2.10) by taking $(u^n_j)^2$ as $M(u^n_j)$, where $M$ is a constant satisfying:

$$\max_j \{u^n_j\} \leq M.$$ 

It is obvious that coefficient matrices $A$, $B$ and $C$ are tri-diagonal and symmetric in (2.10), in which

$$C = (\Delta x)^2A - B.$$ 

We record the eigenvalues of matrices $B$ and $C$ as $\lambda(B)$ and $\lambda(C)$ respectively. Using Lemma 2.1 gives

$$\left| \frac{(1 + M)\lambda(B)}{\lambda(C)} \Delta t \right| \leq 2\sqrt{2}.$$ 

Applying Lemma 2.2, we get

$$(1 + M)\Delta t \frac{|-2 + 2 \cos(\frac{k\pi}{N+1})|}{|\frac{10(\Delta x)^2}{12} + 2(\frac{(\Delta x)^2}{12} - 1) \cos(\frac{k\pi}{N+1})|} \leq 2\sqrt{2},$$

which gives

$$\Delta t \leq \frac{2\sqrt{2}}{1 + M} \frac{8(\Delta x)^2/12}{4} = \frac{\sqrt{2}}{3(1 + M)}(\Delta x)^2.$$ 

This completes the proof. \(\Box\)

3. The Improved Boussinesq Equation

Currently, there is no analytical solution available for the damped IBq; therefore, the IBq is chosen to test the accuracy and efficiency of our method. We set the parameters that control damping effects $\tau_s = \tau_h = 0$.

3.1. Validation of the Method

We now evaluate the accuracy and efficiency of the method. Let $u_j^k$, $U_j^k$ are the analytical solution and approximation solution of the equation, respectively, the following norms are calculated:

$$L^\infty = \|u^k - U^k\|_{\infty} = \max_j |u^k_j - U^k_j|,$$ 

(3.1a)

$$L^2 = \|u^k - U^k\|_2 = \left( \frac{1}{N} \sum_j (u^k_j - U^k_j)^2 \right)^{\frac{1}{2}}.$$ 

(3.1b)

The computational orders of the method presented in this paper (denoted by C-order) are defined using the following formula [21]:

$$\frac{\log(E_1/E_2)}{\log(h_1/h_2)}.$$
where $E_1, E_2$ are the errors corresponding to grids with mesh sizes $h_1$ and $h_2$, respectively.

**Propagation of a single wave.** We choose the initial conditions as follows:

$$u(x,0) = A \text{sech}^2\left(\sqrt{\frac{A}{6}} x - x_0 \right),$$  
(3.2a)

$$u_t(x,0) = 2A \sqrt{\frac{A}{6}} \text{sech}^2\left(\sqrt{\frac{A}{6}} x - x_0 \right) \tanh\left(\sqrt{\frac{A}{6}} x - x_0 \right),$$  
(3.2b)

where parameters $A, x_0, c$ denote the amplitude, initial position, and wave speed, respectively, and $c = \sqrt{1 + 2A}$. This problem has a traveling wave solution of analytical form:

$$u(x,t) = A \text{sech}^2\left(\sqrt{\frac{A}{6}} x - x_0 - ct \right).$$

We choose $A = 0.5, x_0 = 0$, the problem is discretized on $x \in [20, 80]$ with evenly spaced knots, and it is solved until $t = 40$. Then, we demonstrate the convergence and accuracy of the compact finite difference method. We compare our method with the finite volume quadratic element method (FVM2 for short) presented in [15]. In fact, FVM2 is equivalent to the central finite difference method on a mesh twice times refined. Tables 3.1 and 3.2 give clear approximation error information on the $L^\infty$ error and $L^2$ error of these two methods under different resolutions for space and time, respectively. We can see from Table 3.1 that the approximation convergence order of the compact finite difference scheme in $L^\infty$ norm is approximately 4.

### Table 3.1: $L^\infty$ error and $L^2$ error of the compact finite difference scheme.

<table>
<thead>
<tr>
<th>$\Delta x$</th>
<th>$\Delta t$</th>
<th>$L^\infty$ error</th>
<th>C-order</th>
<th>$L^2$ error</th>
<th>C-order</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.0</td>
<td>0.5</td>
<td>7.3240E-3</td>
<td>—</td>
<td>1.3132E-2</td>
<td>—</td>
</tr>
<tr>
<td>1.0</td>
<td>0.25</td>
<td>5.0300E-4</td>
<td>3.86</td>
<td>1.1420E-3</td>
<td>3.52</td>
</tr>
<tr>
<td>0.5</td>
<td>0.125</td>
<td>3.1000E-5</td>
<td>4.02</td>
<td>1.0000E-4</td>
<td>3.52</td>
</tr>
<tr>
<td>0.25</td>
<td>0.0625</td>
<td>1.9350E-6</td>
<td>4.00</td>
<td>8.7775E-6</td>
<td>3.51</td>
</tr>
</tbody>
</table>

### Table 3.2: $L^\infty$ error and $L^2$ error of the finite volume quadratic element method.

<table>
<thead>
<tr>
<th>$\Delta x$</th>
<th>$\Delta t$</th>
<th>$L^\infty$ error</th>
<th>C-order</th>
<th>$L^2$ error</th>
<th>C-order</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.0</td>
<td>0.5</td>
<td>2.4524E-2</td>
<td>—</td>
<td>5.4779E-3</td>
<td>—</td>
</tr>
<tr>
<td>1.0</td>
<td>0.25</td>
<td>5.7148E-3</td>
<td>2.10</td>
<td>1.2866E-3</td>
<td>2.09</td>
</tr>
<tr>
<td>0.5</td>
<td>0.125</td>
<td>1.4056E-3</td>
<td>2.02</td>
<td>3.1761E-4</td>
<td>2.02</td>
</tr>
<tr>
<td>0.25</td>
<td>0.0625</td>
<td>3.5037E-4</td>
<td>2.00</td>
<td>7.9754E-5</td>
<td>1.99</td>
</tr>
</tbody>
</table>

However, the convergence order in the $L^2$ norm doesn’t reach 4, which suggests a slight loss of the accuracy, indicating that perhaps the ratio of the time-step and the space-step is not optimal. From Table 3.2, FVM2 is accurate to the second order. At each time iteration, the computational cost of the compact scheme is nearly four-fold that of FVM2. The compact scheme achieves more accuracy and efficiency than FVM2 on the same mesh. Therefore, the compact scheme is used in the following simulations, and the default resolution is $\Delta x = 0.5, \Delta t = 0.125$ with a compromise between efficiency and accuracy.

The compact finite difference scheme was used to solve this problem in order to test its capacity for conservation. The problem is discretized on $x \in [20, 80]$ with evenly spaced knots.
A Compact Fourth-order Scheme for the Boussinesq Equation

Fig. 3.1. $A = 0.5$, propagation of a single wave. The contour line on the right panel starts from 0.05 to 0.5 with a level step of 0.1.

Fig. 3.2. $A = 0.5$, The evolution of relative error of the conserved invariant $\int_{-\infty}^{+\infty} u(x, t)dx$, the left panel shows the evolution of the amplitude of wave.

(only solutions located in the range $x \in [20, 80]$ are plotted), and the evolution of the single solitary wave without damping obtained over the time interval $[0, 40]$ is shown in Fig. 3.1. The left panel shows that the solitary wave propagates towards the right without any change in form. In addition, the amplitude of wave is not smoothed noticeably, and no secondary waves are produced at the end of the solitary wave. It follows that the solitary wave continues to evolve at the initial velocity. The compact finite difference scheme is capable of modeling the propagation of the solitary wave.

From [27], we know that the IBq (1.1) has a conserved invariant of mass $I(t) = \int_{-\infty}^{+\infty} u(x, t)dx$, and we define its relative error as follows:

$$err(t) = \frac{\int_{-\infty}^{+\infty} Udx}{\int_{-\infty}^{+\infty} u(x, 0)dx} - 1.$$ 

Fig. 3.2 shows the time evolution of the amplitude of wave and the relative error $err(t)$, and it can be seen that the scale of the error is close to $10^{-12}$, that is, the variation of $I(t)$ is trivial. These results show that the chosen subinterval $[-100, 100]$ is sufficient to conserve the invariant $I(t)$ and that our schemes are efficient for the considered problem. Therefore, all simulations are plotted over subinterval $(x, t) \subset [-80, 80] \times [0, 40]$ if a specific region is not given.
3.2. Numerical experiments

We chose three kinds of different initial conditions to test our method and provide some physical interpretations of single-wave splitting and double-wave interaction from a numerical perspective.

**Experiment 1:** Single wave splitting. The initial conditions are given as follows:

\[ u(x, 0) = A \sech^2 \left( \frac{\sqrt{A x - x_0}}{c} \right), \quad u_t(x, 0) = 0, \]

where \( A = 0.5 \) and \( x_0 = 0, c = \sqrt{1 + 2A/3} \).

The problem was solved over the region: \(-80 \leq x \leq 80, 0 \leq t \leq 40\). The numerical simulation is shown in Fig. 3.3, which shows the initial stationary wave of amplitude 0.5 splitting up into two smaller diverging waves, one traveling towards the left and the other one towards the right. The amplitudes of these two solitary waves are slightly smaller than half the wave height. It is also seen that secondary waves with amplitudes greater than 0.01 are produced between the diverging distance.

**Experiment 2:** Interaction of the double solitary waves.

First, we give an example of takeover of the double solitary waves, that is, the two waves travel in the same direction. In this case, the amplitude of the latter wave should be greater than that of the former one; otherwise, they would not interact.

The initial conditions are provided as follows:

\[
\begin{align*}
    u(x, 0) &= A_1 \sech^2 \left( \frac{1}{c_1} \sqrt{\frac{A_1}{6}(x - x_1^0)} \right) + A_2 \sech^2 \left( \frac{1}{c_2} \sqrt{\frac{A_2}{6}(x - x_2^0)} \right), \\
    u_t(x, 0) &= 2A_1 \sqrt{\frac{A_1}{6}} \sech \left( \frac{1}{c_1} \sqrt{\frac{A_1}{6}(x - x_1^0)} \right) \tanh \left[ \frac{1}{c_1} \sqrt{\frac{A_1}{6}(x - x_1^0)} \right] \\
    &\quad + 2A_2 \sqrt{\frac{A_2}{6}} \sech \left( \frac{1}{c_2} \sqrt{\frac{A_2}{6}(x - x_2^0)} \right) \tanh \left[ \frac{1}{c_2} \sqrt{\frac{A_2}{6}(x - x_2^0)} \right],
\end{align*}
\]

where \( c_1 = \sqrt{1 + 2A_1/3} \) and \( c_2 = \sqrt{1 + 2A_2/3} \).

We set the initial positions of two waves \( x_1^0 = -20, x_2^0 = -10 \), and solve the problem over the domain \(-50 \leq x \leq 120, 0 \leq t \leq 70\). Fig. 3.4 exhibits the takeover of two solitary waves at
different time levels. It is noted that in the nonlinear interaction region, the maximum joint amplitude is less than the largest amplitude of the two solitary waves. Moreover, an oscillating tail with an amplitude greater than 0.01 is generated.

Fig. 3.4. Takeover of two solitary waves with $A_1 = 2.0, A_2 = 0.5$. The contour line on the right panel starts from 0.01 to 0.5 with a level step of 0.1 and from 0.6 to 2.5 with a level step of 0.4.

Next, we provide some examples for head-on collision of double waves with equal amplitudes, that is, the two waves travel in reverse directions. The initial conditions are the same as Eq. (3.3), but $c_1 = \sqrt{1 + 2A_1/3}$ and $c_2 = -\sqrt{1 + 2A_2/3}$. We set the initial positions of two waves as $x_0^1 = -20, x_0^2 = 40$, and solve the problem over the domain $-80 \leq x \leq 80, 0 \leq t \leq 40$.

Fig. 3.5. Head-on collision with $A_1 = 0.2, A_2 = 0.2$. The contour line on the right panel starts from 0.01 to 0.4 with a level step of 0.08.

From the simulation of head-on collision of double waves with equal amplitudes, we see that when the amplitude is not great than 0.5 in the case of Fig. 3.5 and Fig. 3.6, the collision is elastic, i.e., the loss of wave energy is nearly negligible in the collision process. Therefore, the amplitudes of the produced secondary oscillations are smaller than 0.02, which are invisible and can be neglected. In other cases, such as in Fig. 3.7, the interaction is inelastic, and the wave energy suffers considerable loss during the process. The lost energy is mainly converted to the energy of secondary waves. Moreover, it is shown that the strongest secondary wave has an amplitude larger than 0.02.

**Experiment 3:** Solution blow-up. We model a kind of blow-up solution [28, 29]. The IBq (1.1) with Neumann boundary conditions is considered on $x \in [0, 1]$, and the initial conditions
Fig. 3.6. Head-on collision with $A_1 = 0.5, A_2 = 0.5$. The contour line on the right panel starts from 0.01 to 1.0 with a level step of 0.1.

Fig. 3.7. Head-on collision with $A_1 = 0.8, A_2 = 0.8$. The contour line on the right panel starts from 0.01 to 1.6 with a level step of 0.2.

Fig. 3.8. Solution changes at different time $t$.

are as follows:

$$u(x, 0) = -1000 - 1000 \cos(\pi x) - 1000 \cos(2\pi x),$$

$$u_t(x, 0) = -1000 \cos(\pi x) - 1000 \cos(2\pi x).$$

From [29] it is known there exists a $\bar{T} > 0$ such that a unique local solution $u \in C^2([0, \bar{T}); H^2(0, 1)) \cap$
$H^1_t(0, 1))$ exists and satisfies
\[ u(0, t) \to -\infty, \quad \|u(\cdot, t)\|_{L^2(0, 1)} \to +\infty, \quad \text{as } t \to \tilde{T}^- . \]

We use the compact finite difference scheme to simulate this problem, and the space step chosen is 0.05. Because the solution changes drastically, we have to set a very small time step ($10^{-3}$) to generate desirable results until $t = 0.052$. The numerical solution at different values of $t$ is shown in Fig. 3.8.

The diagram shows that the solution between $t \in [0.04, 0.06]$ decreases far more quickly than that between $t \in [0, 0.04]$, the blow-up occurs at about $t = 0.052$.

4. The Effect of the Damping Terms

In this section, we use the compact finite difference scheme to study the effect of damping terms on the propagation of a single wave. All the simulations are solved over the domain $(x, t) \subset [100, 100] \times [0, 40]$. Only solutions located in the range $x \in [50, 80]$ are plotted for a clearer visualization.

4.1. Stokes damping

Firstly, we consider the evolution of a single solitary wave in the presence of Stokes damping. $\tau_h = 0$ is set for hydrodynamical damping. The damped IBq (1.2) is solved with the initial conditions defined in Eq. (3.2) and we use $\Delta x = 0.5, \Delta t = 0.125$ in the simulation. Fig. 4.1 presents the evolution of the wave profile (left) and level curves (right) with $\tau_s = 0.2$. It can be seen that the wave becomes a dwarf and breaks down drastically, and the wave profile is destroyed at about $t = 20$. When $\tau_s = 0.4$ or even bigger value is chosen, we find that the wave stands still, but the wave is destroyed instantly, as perhaps the condition of oscillation is not satisfied [30], the long-wave components of a wave packet with a wave number that does not satisfy the condition do not propagate. The variations of solitary wave height (left) and the relative error $\text{err}(t)$ (right) for $\tau_s = 0.2$ and $\tau_s = 0.02$ are shown in Fig. 4.2. We see that the variation of $\text{err}(t)$ for $\tau_s = 0.2$ and $\tau_s = 0.02$ is very close in value to the results in Fig. 3.2, with a scale of $10^{-12}$ in our simulation. These results illustrate that Stokes damping will not change the value of the conserved quantity $I(t)$.

![Fig. 4.1. Single wave propagation with Stokes damping coefficient $\tau_s = 0.2$. The contour line on the right panel starts from 0.01 to 0.5 with a level step of 0.05.](image-url)
4.2. The hydrodynamic damping

We test the effect of hydrodynamic damping on the dynamics of the single wave. $\tau_s = 0$ is set for Stokes damping. The coefficients $\tau_h = 0.2$ and $\tau_h = 0.02$ are chosen for hydrodynamical damping and other parameters are the same as those in the above simulation. Fig. 4.3 presents the propagation for a single solitary wave (left) and level curves (right) when $\tau_h = 0.2$ is considered. We can see that the profile is slightly smoothed by hydrodynamical damping. However, the solitary wave does not lag behind, which implies that the wave speed is influenced
trivially. The variations of solitary wave height (left) and the relative error $err(t)$ (right) for $\tau_0 = 0.2$ and $\tau_0 = 0.02$ are shown in Fig. 4.4. It is clear that the variation of $err(t)$ for $\tau_0 = 0.2$ and $\tau_0 = 0.02$ is very close in value to the results in Fig. 3.2, which has a scale of $10^{-12}$ in our simulation. These results demonstrate that the hydrodynamic damping will cause a trivial loss of the conserved quantity $\int_{-\infty}^{+\infty} u^2 dx$. From Fig. 4.2 (left) and Fig. 4.4 (left), it follows that Stokes damping has a stronger smoothing effect than the hydrodynamical damping under the same condition.

5. Summary and Conclusions

In this paper, a fourth-order compact finite difference scheme has been applied for solving the initial boundary problems of the damped Improved Boussinesq equation. The scheme can be implemented with relatively less effort, and the resulting tri-diagonal coefficient matrix can be solved efficiently. Some numerical experiments including single-wave splitting, wave interaction and a type of blow-up behaviour could be simulated well using this method. In addition, we test the influence of two different damping terms on the IBq by the evolution of a single wave, which demonstrates that Stokes damping decreases the wave height more drastically than hydrodynamic damping. Moreover, the damping terms make no difference to the conserved invariant.

Acknowledgments. This project is partly supported by the State Key Program of National Natural Science of China (No. 41231173), a grant from the National Natural Science foundation of China (Grant No. 41306078). The project is also funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions.

References


