Applied Mathematics E-Notes, 8(2008), 159-170 © ISSN 1607-2510
Available free at mirror sites of http://www.math.nthu.edu.tw/∼amen/

Numerical Investigation For Solitary Solutions Of The Boussinesq Equation*

Kamel Al-Khaled and Ameina S. Nusier†

Received 7 February 2007

Abstract

Boussinesq equation has recently been found to possess an interesting soliton-interaction mechanism. This paper provides a technical description of the application of the Adomian's decomposition solution and the Galerkin interpolation methods based on Sinc functions to derive soliton wave solutions for problems involving the Boussinesq equation. Numerical experiments are also reported.

1 Introduction

In recent years, studies of nonlinear waves in media with dispersion have attracted the attention of many mathematicians and physicists. One of the equations describing such processes is the Boussinesq one which was derived in [3] and found to possess an interesting soliton-interaction mechanism. It can be written as

$$\mathbf{A}u = au_{xxxx} + u_{xx} + b(u^2)_{xx}, \ \ x \in \mathbf{R}, \ t > 0 \tag{1}$$

where the differential operator $\mathbf{A}$ is $\mathbf{A} = \partial^2/\partial t^2$, and $u = u(x,t)$ is an elevation of the free surface of fluid. Subscripts denote partial derivatives, a and b are real constants depend on the depth of fluid and the characteristic speed of propagation of long waves. Equation (1) is referred to as the "good" Boussinesq equation, or the nonlinear beam equation. The related equation

$$u_{tt} = -\alpha u_{xxxx} + u_{xx} + \beta(u^2)_{xx}, \ \ x \in \mathbf{R}, \ t > 0 \tag{2}$$

known as the "bad" Boussinesq equation has been studied by Hirota [7]. The "bad" version arises in the study of water waves. Specifically, it is used to describe a two-dimensional flow of a body of water over a flat bottom with air above the water, assuming that the water waves have small amplitudes and the water is shallow. It also appeared in a posterior study of the Fermi-Pasta-Ulam problem, which was performed to show that the finiteness of thermal conductivity of an anharmonic lattice was related to nonlinear forces in the springs, for more details see [18]. Equation (1)

*Mathematics Subject Classifications: 35Q51, 35Q99, 35Q53.

†Department of Mathematics and Statistics, Jordan University of Science and Technology IRBID 22110, Jordan

arises in the modeling of nonlinear strings. It should be emphasized that the Boussinesq equation admits both right- and left-running wave solutions while the well-known Korteweg-de-Vries (KdV) governs waves traveling in only one direction [2]. The finite difference method has been used to solve the good Boussinesq [22], where stability and convergence has been also studied.

There is extensive literature on equations (1) and (2) (see, for example [7, 13, 16], and the references there) which has been studied from various points of view. In [13] an exact formula is given for the interaction of solitary waves. Numerical experience is also reported. Hirota [7] has deduced conservation laws and has examined N-soliton interactions. The representation of periodic waves as sums of solitons has been given by Whitham [16]. In [9] the authors studied the Boussinesq equation from the point of view of a multiple-time reduction perturbation method. As a consequence of the elimination of the secular producing terms through the use of the KdV hierarchy, they showed that the solitary-wave of the Boussinesq equation is a solitary-wave satisfying simultaneously all equations of the KdV hierarchy. In [10], an alternative method is used to generate the bidirectional soliton solutions of the classical Boussinesq system. In [5], the “good” Boussinesq equations is studied numerically using the implicit finite-difference scheme, soliton solutions are shown to exist for finite range of amplitude size. Putting $u = v - 1/2$, it can easily be shown that the linear term u_{xx} is removed from equation (1) resulting in the equation

$$v_{tt} = -v_{xxxx} + (v^2)_{xx}. \tag{3}$$

For the purpose of numerical analysis, it is simpler to use equation (3) instead of (1).

A common procedure for solving Equation (1) is a Galerkin discretization of the spatial variable with time-dependent coefficients. This gives a system of ordinary differential equations typically solved by difference techniques. In contrast, the method of the present works implements a Galerkin scheme achieves considerable numerical success through an adroit choice of basis functions. The basis elements are products of sinc functions composed with suitable conformal maps. A thorough review of sinc function properties is found in [14, 11].

The sine-Galerkin method is space and time is developed. The judicious choice of a conformal map provides an approximate solution to equation (1) valid on the domain $(x, t) \in \mathbf{R} \times (0, \infty)$. A drawback is that the method produces a full linear system in contrast to the banded structures associated with finite difference and finite element techniques. The structure of the system, including symmetry, results in computational efficiency. Also, we solve the Boussinesq equation without discretization by using the Adomian's Decomposition Method (ADM), the techniques has many advantages over classical techniques, mainly, it avoids linearization in order to find solutions of given nonlinear equations. It also avoids discretization and provides an efficiently numerical solution with high accuracy, minimal calculations, avoidance of physically unrealistic assumptions [1].

The organization of the article is as follows. In section 2 we make use of the truncated Painleve' expansion and obtain certain type of soliton solutions. Section 3 of this paper outlines the sinc function properties pertinent in the derivation of the discrete linear system. In particular the Galerkin approximation and the weighted inner product are defined. Error bounds are derived in [14] and are used here to facilitate description of the parameter selections for implementation of the discrete system. In section 4 we approach the Boussinesq equation differently, but effectively, by using an alternative technique, we will use the ADM introduced by Adomian [1] to construct a series solution of (1). Numerical results are presented in section 5. The example discussed in this section best exhibits the features necessary for the practical implementation of the ADM. Concluding remarks are given in the last section.

2 General Formulas of Soliton Solutions

In this section, we make use of the truncated Painleve' expansion method to obtain certain soliton type explicit solutions. A non-linear partial differential equation of the form

$$H(u^{(i)}, u_x^{(i)}, u_t^{(i)}, u_{xt}^{(i)}, ...) = 0, \quad i = 1, 2, ..., m \tag{4}$$

where $u^{(i)} = u^{(i)}(x,t), \quad i = 1, 2, ..., m$ are dependent variables, and subscripts denote partial derivatives, is said to possess the Painleve' property when the solutions of the non-linear partial differential equation (4) are *single valued* about the movable, singularity manifold which is *non-characteristic*. In other words, all solutions of equation (4) can be expressed as Laurent series

$$u^{(i)}(x,t) = \sum_{j=0}^{\infty} u_j^{(i)}(x,t)\phi(x,t)^{j+\alpha_i}, \quad i = 1, 2, ..., m, \tag{5}$$

with sufficient number of arbitrary functions among $u_j^{(i)}(x,t)$ in addition to $\phi, u_j^{(i)}(x,t)$ are analytic functions, α_i are negative integers.

It is very tedious to study whether a given partial differential equation passes the Painleve test, thus the application of computer algebra can be very helpful in such calculations. Various researchers have developed computer programs for the Painleve test for non-linear partial differential equations, such as the WTC method [19], the Kruskal's simplification method [8], and Wkptest [20]. Using one of these, the conclusion will be exactly the same. For purposes of finding soliton solutions of equation (1), we give the detailed description of WTC-Kruskal algorithm as mentioned in [21] which contains four steps.

Step 1. Leading order analysis

To determine leading order exponents α_i and coefficients $u_0^{(i)}$, letting

$$u^{(i)}(x,t) = u_0^{(i)}\phi^{\alpha_i}, \quad i = 1, 2, ..., m,$$

and inserting it into (4), then balancing the minimal power terms, one can obtain all possible $(\alpha_i, u_0^{(i)})$. If the only possible α_i are not integers, then the algorithm stops. Otherwise, one has to go on with the next step.

Step 2. To generate truncated expansions

For each pair of $(\alpha_i, u_0^{(i)})$ from step 1, calculate the possible truncated expansions in the form

$$u^{(i)}(x,t) = u_0^{(i)}\phi^{\alpha_i} + u_1^{(i)}\phi^{\alpha_i+1} + ... + u_{-\alpha_i}^{(i)}\phi^0, \ i = 1, 2, ..., m.$$

If $u_k^{(i)}, (k = 0, ..., -\alpha_i - 1)$ cannot be determined, then the series (5) cannot be truncated at constant terms.

Step 3. To find the resonances

If all possible α_i are integers, calculate resonances r for each pair of $(\alpha_i, u_0^{(i)})$ by inserting

$$u^{(i)}(x,t) = u_0^{(i)}\phi^{\alpha_i} + u_r^{(i)}\phi^{\alpha_i + r},\ i = 1, 2, ..., m,$$

into (4) and balancing the most singular terms. For a single partial differential equation, all resonances will be distinct integers, whether positive or negative.

Step 4. To verify compatibility conditions

The final step is to compute the coefficients at the non-resonances, and verify the compatibility condition at every positive resonances. For each branch of Equation (4) we insert the truncated expansions

$$u^{(i)}(x,t) = \sum_{j=0}^{rmax} u_j^{(i)}\phi^{j+\alpha_i},\ i = 1, 2, ..., m$$

into (4), where $rmax$ is the largest resonance. For the rapidness of verifying compatibility conditions, we may use $u_k^{(i)}$ $(k = 0, 1, ..., -\alpha_i - 1)$. For more details of using this algorithm see [21] and the references therein.

For our case, and in order to obtain soliton solutions of equation (1), let

$$u(x,t) = \phi^{\alpha}(x,t)u_0(x,t). \tag{6}$$

Substituting equation (6) into equation (1) and balancing the highest-order contributions from the linear term (i.e., u_{xxxx}) with the highest order contributions from the nonlinear terms (i.e., $(u^2)_{xx}$), we get $\alpha = -2$. Hence the most singular terms will vanish if

$$u_0 = \frac{15}{2}\phi_x^2. \tag{7}$$

Next, to find u_1, let

$$u(x,t) = \phi^{-2}[u_0 + \phi u_1]. \tag{8}$$

When substituting equation (8) into equation (1) with *Mathematica*, we let the coefficients of like powers of ϕ to vanish (ϕ^{-5} in this case), so we get

$$u_1 = \frac{-495}{156}\phi_{xx}. \tag{9}$$

Therefore, a truncated series solution of equation (1) is given by

$$u(x,t) = \frac{15}{2}\frac{\phi_x^2}{\phi^2} - \frac{495}{156}\frac{\phi_{xx}}{\phi}. \tag{10}$$

Thus, we may continue to find a family of exact analytical solutions to equation (1). We note that once a Painleve' transformation is discovered, and a set of *seed* solutions is given, one will be able to find an infinite number of solutions by the repeated applications of the transformation. For more details of applications see [17].

To solve equation (1) with $a = 1$ and $b = 1$, a trial solution

$$\phi(x,t) = 1 + \exp[kx \mp k\sqrt{1+k^2}t]$$

where k is an arbitrary constant, is substituting into the constrains (equations (7)-(10) of $\phi(x,t)$ gives the general single soliton solution (see [15])

$$u(x,t) = 2\frac{k^2 \exp[kx + k\sqrt{1+k^2}t]}{(1 + \exp[kx + k\sqrt{1+k^2}t])^2}. \tag{11}$$

With the truncated Painleve' expansion analysis, we found a family of exact analytical solution (11) of equation (1). This exact solution will be used to find the accuracy of our approximate solution to be presented in the next sections.

3 Sinc Solution of The Boussinesq Equation

A space-time tensor product Galerkin method for the approximation of the solution of equation (1) may be summarized as follows. Select a set of basis functions

$$\{S_i(x), S_j^*(t)\} \tag{12}$$

and define an approximate solution as

$$u_{m_x,m_t}(x,t) = \sum_{i=-M_x}^{N_x} \sum_{j=-M_t}^{N_t} u_{ij} S_i(x) S_j^*(t), \quad \begin{array}{c} m_x = M_x + N_x + 1 \\ m_t = M_t + N_t + 1 \end{array}. \tag{13}$$

The unknown coefficients $\{u_{ij}\}$ in Equation (13) are to be determined. For the sinc-Galerkin method, the basis functions are derived from the Whittaker cardinal (sinc) function.

$$sinc(x) = \frac{\sin(\pi x)}{\pi x}, \quad x \in \mathbf{R}. \tag{14}$$

Define the translated sinc function by

$$S(k,h)(x) = sinc\Big(\frac{x - kh}{\pi x}\Big), \quad h > 0. \tag{15}$$

To construct basis functions on the intervals $(-\infty, \infty)$ and $(0, \infty)$, respectively, consider the conformal maps

$$\phi(z) = z$$

and

$$\psi(w) = \ln(\sinh(w)),$$

this latter map being preferable, over $\psi(w) = \ln(w)$ for solution of initial value problems on the interval $(0, \infty)$, due to its broader range of convergence. The map ψ carries the infinite wedge

$$\mathcal{D}_w = \{w = t + is : |arg(w)| < d \leq \pi/2\}$$

onto the infinite strip

$$\mathcal{D}_s = \{\zeta = \xi + i\eta; : |\eta| < d\}.$$

The map ϕ carries the infinite strip $\mathcal{D}_s$ into itself. These regions are depicted in [14, p. 67]. The compositions

$$S_i(x) = S(i, h_x) \circ \phi(x) = S(i, h_x)$$

and

$$S_j^*(t) = S(j, h_t) \circ \psi(t)$$

define the basis elements for equation (12) on $(-\infty, \infty)$ and $(0, \infty)$, respectively. The "mesh sizes" h_x and h_t represent the mesh sizes in $\mathcal{D}_s$ for the uniform grids $\{kh_x\}$, $-\infty < k < \infty$ and $\{ph_t\}$, $-\infty < p < \infty$. The sinc grid points $x_k \in (-\infty, \infty)$ in $\mathcal{D}_s$ and $t_p \in (0, \infty)$ in $\mathcal{D}_w$ are the inverse images of the equispaced grids; that is,

$$x_k = \phi^{-1}(kh_x) = kh_x$$

and

$$t_p = \psi^{-1}(ph_t) = \sinh^{-1}(e^{ph_t}).$$

The following convenient notation will be useful in formulating the discrete system. Let

$$\delta_{kj}^{(4)} = h^4 \frac{d^4}{d\phi^4}[S(k,h) \circ \phi(x)]|_{x=x_j} = \begin{cases} \frac{\pi^4}{5}, & j = k \\ \\ \frac{-4(-1)^{j-k}}{(j-k)^4}[6 - \pi^2(j-k)^2], & j \neq k \end{cases} \tag{16}$$

$$\delta_{kj}^{(2)} = h^2 \frac{d^2}{d\phi^2}[S(k,h) \circ \phi(x)]|_{x=x_j} = \begin{cases} \frac{-\pi^2}{3}, & j = k \\ \\ \frac{-2(-1)^{j-k}}{(j-k)^2}, & j \neq k \end{cases} \tag{17}$$

and

$$\delta_{kj}^{(0)} = [S(k,h) \circ \phi(x)]|_{x=x_j} = \begin{cases} 1, & j = k \\ \\ 0, & j \neq k \end{cases} \tag{18}$$

denote the evaluation of the basis functions and their derivatives with respect to the map ϕ. And $I^\ell, \ell = 0, 2, 4$ are $m \times m$ matrices whose jk-th entry is given by $\delta_{kj}^{(\ell)}$ from (16)-(18).

For the purposes of this paper, the following simple problem provides all necessary information.

EXAMPLE 1. Consider solving the boundary value problem

$$\begin{array}{c} \mathbf{L}u = -u''(x) + q(x)u(x) = g(x), \quad -\infty < x < \infty \\ \lim_{x \to \mp\infty} u(x) = 0. \end{array} \tag{19}$$

via the Galerkin procedure with a sinc basis, we assume an approximate solution of the form

$$u_m(x) = \sum_{k=-M}^{N} u_k sinc\left(\frac{x-kh}{h}\right), \quad m = M+N+1$$

where m denotes the number of basis functions in the expansion. Orthogonalization of the residual against each basis function using a weighted inner product yields

$$\langle \mathbf{L}u - g, S_j\rangle = 0, \ \ -M \le j \le N$$

where

$$\langle f, g\rangle = \int_{-\infty}^{\infty} f(x)g(x)w(x)dx$$

and, $w(x) = (\phi(x))^{-r}, \ r \ge 0$. For the purposes of this paper it is sufficient to consider the weight exponent $r = 0$. Integrating by parts to remove all derivatives from u, and applying the sinc quadrature rule yields the sinc system (see, [11]).

The basis used to solve (1) is a product of sinc basis functions in the x and y directions. The sinc basis functions are given by $S_i(x)$ and $S_j^*(t)$. Assume a product approximate solution of the form

$$u_{m_x,m_t}(x,t) = \sum_{j=-M_x}^{N_x}\sum_{k=-M_t}^{N_t} u_{jk}S_j(x)S_k^*(t).$$

Here $m_x = M_x + N_x + 1, m_t = M_t + N_t + 1$, and $h_x > 0, h_t > 0$. The sinc nodes x_j and t_k are chosen so that $x_j = \phi^{-1}(jh_x)$ and $t_k = \psi^{-1}(kh_t)$. Orthogonalization of the residual against each basis function

$$\langle \mathbf{A}u - u_{tt}, S_pS_q\rangle = 0, \ \ -M_x \le p \le N_x, \ -M_t \le q \le N_t$$

uses the weighted inner product

$$\langle v, w\rangle = \int_{-\infty}^{\infty}\int_{0}^{\infty} v(x,t)w(x,t)(\phi'(x))^{-1/2}(\psi'(t))^{-1/2}dtdx.$$

Integrating by parts is used to remove all partial derivatives from u_{m_x,m_t}, and applying the sinc quadrature rule (found in [14]) yields the discrete sinc system. Similar complete development can be found in [12]. The following notation will be necessary for writing down the system. For a function $f(x)$ defined on an interval, denote by $\vec{f}$ the $m_x \times 1$ vector $\vec{f} = [f(x_{-M_x}), ..., f(x_{N_x})]^T$, and let the $m_x \times m_x$ diagonal matrix be $\mathcal{D}(f) = diag[f(x_{-M_x}), ..., f(x_{N_x})]$.

Similar definitions can be used for the variable t. The fundamental sinc-Galerkin matrices for the second derivative are the $m_x \times m_x$ matrix

$$\Gamma_x^2 = \left\{\frac{-1}{h_x^2}I^{(2)} + \mathcal{D}\left(\frac{1}{4}\right)\right\}$$

and the $m_t \times m_t$ matrix

$$\Gamma_t^2 = \left\{ \frac{-1}{h_t^2} I^{(2)} + \mathcal{D}\left(\frac{1}{4}\right) \right\} \mathcal{D}(\sqrt{\psi'}).$$

For the fourth derivative,

$$\Gamma_x^4 = \frac{1}{h_x^4} \left\{ I^{(4)} - \frac{5}{2} h_x^2 I^{(2)} + \frac{9}{16} h_x^4 I_x^{(0)} \right\}.$$

Then, the discrete sinc-Galerkin system for (1) in Sylvester system is

$$\mathcal{D}\left((\psi')^{3/2}\right) \Gamma_t^2 U = U \left[a\Gamma_x^4 + \Gamma_x^2 \right]^T + b\,(U \circ U)\,(\Gamma_x^2)^T$$

where the coefficient matrix $U = [u_{jk}]$ is a matrix of size $m_x \times m_t$, and the notation $\circ$ to denote the Hadamard matrix multiplication.

4 Adomian Decomposition Method

In this section we outline Adomian decomposition method (ADM) as presented in [1, 6, 15] for the Boussinesq equation. In this study we shall consider equation (1) in the standard form

$$\mathbf{A}u = au_{xxxx} + u_{xx} + b(u^2)_{xx}, \quad x \in \mathbf{R},\ t > 0 \tag{20}$$

the solution of which is to be obtained subject to the initial conditions

$$u(x,0) = f(x), \quad u_t(x,0) = g(x),$$

Assuming the inverse of the operator $\mathbf{A}_t^{-1}$ exists and it can be conveniently be taken as the definite integral with respect to t from 0 to t, i.e., $\mathbf{A}_t^{-1} = \int_0^t \int_0^t (.) dtdt$. Thus applying the inverse operator $\mathbf{A}_t^{-1}$ to (20) yields

$$\mathbf{A}_t^{-1}\mathbf{A}_t u = \mathbf{A}_t^{-1}[au_{xxxx} + u_{xx} + b(u^2)_{xx}].$$

Therefore, it follows that

$$u(x,t) = f(x) + tg(x) + \mathbf{A}_t^{-1}[au_{xxxx} + u_{xx} + b(u^2)_{xx}]. \tag{21}$$

We then decompose the unknown function $u(x,t)$ by a sum of components defined by the decomposition series

$$u(x,t) = \sum_{n=0}^{\infty} u_n(x,t) \tag{22}$$

and the nonlinear term $(u^2)_{xx}$, can be expressed in the form of B_n Adomian's polynomials; thus

$$F(u) = (u^2)_{xx} = \sum_{n=0}^{\infty} B_n \tag{23}$$

where these polynomials can be calculated for all forms of nonlinearity according to specific algorithms constructed in [1]. In this case we use the general form of formula for

$$B_n = \frac{1}{n!}\frac{d^n}{d\lambda^n}\left[F(\sum_{i=0}^{\infty}\lambda^i u_i)\right]_{\lambda=0}, \quad n = 0, 1, 2, ...,$$

this formula is easy to set computer code to get as many polynomials as we need in the calculation of the numerical as well as explicit solutions. Substituting (22), (23) into (20) gives

$$\sum_{n=0}^{\infty} u_n(x,t) = f(x) + tg(x) + \mathbf{A}_t^{-1}\left[(\sum_{n=0}^{\infty} u_n)_{xx} + b\sum_{n=0}^{\infty} B_n + a(\sum_{n=0}^{\infty} u_n)_{xxxx}\right].$$

We can give the first few Adomian polynomials as:

$$u_0(x,t) = f(x), \quad u_1(x,t) = tg(x) + \mathbf{A}_t^{-1}\left[bB_0 + u_{0_{xx}} + au_{0_{xxxx}}\right]$$

and

$$u_{k+1}(x,t) = \mathbf{A}_t^{-1}\left[bB_k + u_{k_{xx}} + au_{kxxxx}\right], \; k \geq 1$$

where

$$B_0 = (u_0^2)_{xx}; \; B_1 = 2u_0u_{1_{xx}} + 4u_{0_x}u_{1_x} + 2u_{0_{xx}}u_1,$$

and

$$B_2 = 2u_0u_{2_{xx}} + 4u_{0_x}u_{2_x} + 2u_{0_{xx}}u_2 + 2u_1u_{1_{xx}} + 2(u_{1_x})^2,$$

and so on.

5 Numerical Results

To illustrate the efficiency of the proposed numerical methods in studying the model equation (1). Equation (1), with $a = b = 1$ has a special solution which describes solitary waves (soliton-type solution) as given in (11), subject to the conditions

$$u(x,0) = \frac{2k^2\exp(kx)}{(1+\exp(kx))^2}, \quad u_t(x,0) = -2\frac{k^3\sqrt{1+k^2}\exp(kx)(\exp(kx)x-1)}{(1+\exp(kx))^3}. \tag{24}$$

This exact solution is used ($k = 1$) as a test for the numerical scheme. Based on the ADM, we construct the solution $u(x,t)$ as

$$\lim_{n\to\infty}\phi_n = u(x,t), \quad \phi_n(x,t) = \sum_{k=0}^{n-1} u_k(x,t), \; n \geq 0. \tag{25}$$

To find the solution of equation (1) subject to the conditions in (24), as mentioned in the previous section we obtain

$$u_0(x,t) = u(x,0), \;\; u_1(x,t) = tu_t(x,0) + \int_0^t\int_0^t [B_0 + u_{0_{xx}} + u_{0_{xxxx}}]dtdt$$

t_j	$x = 10$	$x = 15$	$x = 20$
0.1	2.93053E-11	1.46531E-15	9.69664E-19
0.3	7.89810E-10	3.19107E-13	1.92285E-15
0.5	4.91537E-09	1.00076E-11	6.63803E-14

Table 1: Numerical Results for ADM in Comparison with Analytical Solution

t_j	$x = 10$	$x = 15$	$x = 20$
0.1	2.51345E-06	5.84712E-06	6.67059E-06
0.3	2.65745E-06	6.28414E-06	5.34138E-06
0.5	2.33807E-06	5.61615E-05	2.17532E-05

Table 2: Numerical Results for Sinc Method in Comparison with Analytical Solution

and

$$u_{k+1}(x,t) = \int_0^t \int_0^t [B_k + u_{k_{xx}} + u_{k_{xxxx}}] dtdt, \quad k \geq 1.$$

The approximate solution $u_a(x,t)$ in a series can be written as

$$u_a(x,t) = u_0(x,t) + u_1(x,t) + u_2(x,t) + u_3(x,t)$$

where the exact solution of (1) subject to (24) is given as in (11). The decomposition series (25) solutions generally converge very rapidly in real physical problems. The theoretical treatment of convergence of the decomposition series has been considered in [4]. They obtained some results about the speed of convergence. The sinc method converges exponentially to the exact solution [14]. In order to verify numerically which of the proposed methodology lead to higher accuracy, Tables $1, 2$ show the difference of analytical solution and numerical solution of the absolute error. It is to be noted that 4 terms were used in evaluating the approximate solution using ADM. While in the sinc method, the parameters $M_x = N_t = 16, \alpha = 1$ were used. Examining Tables $1, 2$ closely shows the improvements obtained by using ADM. Higher accuracy can be obtained by evaluating more components. It is very clear from Table 1 that the errors very much depend on the value x. This can explain the nature of the series method.

6 Discussions

On the basis of the numerical results obtained in sections 3 and 4, we conclude that the proposed numerical scheme is proved to be efficient in studying the solitary wave solutions of the Boussinesq equation. A clear conclusion can be drawn from the numerical results that ADM algorithm provides highly accurate numerical solutions without spatial discretization for nonlinear partial differential equations. Finally, we point out that the one-dimensional solutions obtained by ADM can be generalized to two and higher-dimensional solutions, where the equation takes the general form

$$u_{tt} = [Q(u)]_{xx} + [R(u)]_{yy} + \sum_{i=1}^{m} b_i u_{(2i+2)x} + \sum_{j=1}^{k} c_j u_{(2j+2)y}$$

where $R(u) = u^s$, and s, r, b_i, c_j are all real numbers.

References

[1] G. Adomian, Solving Frontier Problems of Physics: The decomposition method, Kluwer Academic Publishers, Boston, 1994

[2] K. Al-Khaled, Sinc numerical solution for solitons and solitary waves, J. Comput. Appl. Math. 130(1-2)(2001), 283–292.

[3] J. Boussinesq, Theorie des ondes et des remous qui se propagent le long d'un canal rectangulaire horizontal, et communiquant au liquide contene dans ce canal des vitesses sensiblement pareilles de la surface au fond, J. Math. Pure Appl., 17(2)(1872), 55–108.

[4] Y. Cherruault, G. Saccomandi and B. Some, New results for convergence of Adomian's method applied to integral equations, Math. Comput. Modeling 16(2)(1992), 85–93.

[5] H. El-Zoheiry, Numerical investigation for the solitary wave interaction of the "good" Boussinesq equation, Appl. Numer. Math.45(2-3)(2003), 161–173.

[6] M. A. Hajji and K. Al-Khaled, Analytic studies and numerical simulations of the generalized Boussinesq equation, Appl. Math. Comput., 191(2)(2007), 320–333.

[7] R. Hirota, Exact N-soliton solution of the wave equation of long waves in shallow-water and in nonlinear lattices, J. Math. Phys. 14(1973), 810–814.

[8] M. Jimbo, M. D. Kruskal and T. Miwa, The Painleve' test for the self-dual Yang-Mills equations, Phys. Lett. A 92(1989), 89.

[9] R. A. Kraenkel, M. A. Manna, J. C. Montero and J. G. Pereira, Boussinesq solitary-wave as a multiple-time solution of the Korteweg-de Vries hierarchy, J. Math. Phys. 36(12)(1995), 6822–6828.

[10] Y. S. Li and J. E. Zhang, Bidirectional soliton solutions of the classical Boussinesq system and AKNS system, Chaos, Solitons and Fractals, 16(2003), 271–277.

[11] J. Lund and K. L. Bowers, Sinc Methods for Quadrature and Differential Equations, SIAM, Philadelphia, 1992.

[12] N. J. Lybeck and K. L. Bowers, The Sinc-Galerkin Schwarz alternating method for Poisson's equation, Prog. Syst. Control Theory., 20(1995), 247–258.

[13] V. S. Manoranjan, A. S. Mitchell and J. L. Morris, Numerical solution of the "good" Boussinesq equation, SIAM J. Sci. Stat. Comput 5(1984), 946–957.

[14] F. Stenger, Numerical Methods Based on Sinc and Analytic Functions, Springer-Verlag, New York, 1993.

[15] A. M. Wazwaz, Construction of soliton solutions and periodic solutions of the Boussinesq equation by the modified decomposition method, Chaos, Solitons and Fractals, 12(2001), 1549–1556.

[16] G. B. Whitham, Comments on periodic waves and solitons, IMA J. Appl. Math. 32(1-3)(1984), 353–366.

[17] W. P. Hong, On Backlund transformation for a generalized Burgers equation and solitons solutions, Phys. Lett. A, 268(2000), 81–84.

[18] N. J. Zabusky and M. D. Kruskal, Interaction of "soliton" in a collisionless plasma and the recurrence of initial states, Phys. Rev. Lett. 15(1965), 240–243.

[19] J. Weiss, M. Tabor, G. Carnevale, The Painleve' property of partial differential equations, J. Math. Phys. 24(1983), 522–526.

[20] G. Q. Xu and Z. B. Li, Symbolic computation of the Painleve test for nonlinear partial differential equations using Maple, Comput. Phys. Commun. 161(2004), 65–75.

[21] G. Q. Xu and Z. B. Li, PDEPtest: a package for the Painleve' test of nonlinear partial differential equations, Appl. Math. Comput. 169(2005), 1364–1379.

[22] T. Ortega and J. M. Sanz-Serna, Nonlinear stability and convergence of finite-difference methods for the "good" Boussinesq equation, Numer. Math. 58(2)(1990), 215–229.