Journal of Computational Physics 230 (2011) 6464–6474

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

A smart nonstandard finite difference scheme for second order nonlinear boundary value problems

Utku Erdogan [a,b,*], Turgut Ozis [a]

[a] Ege University, Science Faculty, Department of Mathematics, C-01/b, 35100 Bornova-Izmir, Turkey
[b] University of Groningen, Department of Mathematics and Computing Science, Bernoulliborg 450, Nijenborgh 9, 9747 AG Groningen, The Netherlands

ARTICLE INFO

Article history:
Received 14 December 2010
Received in revised form 21 April 2011
Accepted 25 April 2011
Available online 5 May 2011

Keywords:
Nonlinear boundary value problems
Nonstandard finite difference
Exponentially fitted difference scheme

ABSTRACT

A new kind of finite difference scheme is presented for special second order nonlinear two point boundary value problems. An artificial parameter is introduced in the scheme. Symbolic computation is proposed for the construction of the scheme. Local truncation error of the method is discussed. Numerical examples are illustrated. Numerical results show that the method is very effective.

1. Introduction

In this paper, we implement a new kind of nonstandard finite difference method for solving the two point nonlinear boundary value problem of the form:

$$y'' = f(x, y), \quad a < x < b, \tag{1}$$

subject to the boundary conditions:

$$y(a) = \alpha, \quad y(b) = \beta. \tag{2}$$

The necessary conditions for the existence and uniqueness of the solution of (1) and (2) are explained in [1]. Much research has been done on the numerical integration of nonlinear BVPs. Usually, the adopted integration methods are the finite difference method [2,3], the shooting method [4,5], the monotone iterative method [6,7], and the quasilinearization method [8,9].

In this paper, many of the ideas are motivated by the works of Mickens [24,25], Simos et al. [13–21] and Vanden Berghe and co-workers [10–12]. They developed different kinds of finite differences formulas and discretization methods for differential equations as alternatives to the standard ones.

2. Motivation

The new method is based on several simple observations. In the classical sense, the first derivative approximations $D^{+}y = \frac{y_{i+1}-y_i}{h}$, $D^{-}y = \frac{y_i-y_{i-1}}{h}$, $D^{0}y = \frac{y_{i+1}-y_{i-1}}{2h}$ can be obtained in a different way. Consider the differential equation:

* Corresponding author at: Ege University, Science Faculty, Department of Mathematics, C-01/b, 35100 Bornova-Izmir, Turkey.
E-mail addresses: utkuerdog@gmail.com (U. Erdogan), turgutozis@gmail.com (T. Ozis).

0021-9991/$ - see front matter © 2011 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2011.04.033

$$y' = f(x, y). \tag{3}$$

Using uniform mesh including $N+1$ points $a = x_0 < x_1 < \cdots < x_N = b$ and $h = (b-a)/N$, if f is frozen at $x = x_i$ in the subinterval (x_{i-1}, x_{i+1}) we get an approximation to (3)

$$\frac{d\tilde{y}}{dx} = f_i, \tag{4}$$

where $f_i = f(x_i, \tilde{y}_i)$ and $\tilde{y}(x_i) = \tilde{y}_i$. The general solution of (4) is:

$$\tilde{y}(x) = f_i x + c.$$

If the continuity conditions $\tilde{y}(x_{i-1}) = \tilde{y}_{i-1}$ and $\tilde{y}(x_{i+1}) = \tilde{y}_{i+1}$ are imposed to determine c, then:

$$\frac{\tilde{y}_{i+1} - \tilde{y}_{i-1}}{2h} = f_i.$$

Comparing with (4) we get:

$$D^0\tilde{y} = \frac{\tilde{y}_{i+1} - \tilde{y}_{i-1}}{2h}.$$

Similarly, by imposing the conditions $\tilde{y}_i(x) = \tilde{y}_i, \tilde{y}(x_{i+1}) = \tilde{y}_{i+1}$ in the interval (x_i, x_{i+1}), the classic forward difference formula is obtained.

All classical finite difference formulas have in common that they are exact for some degrees of polynomials. However, Simos and co-workers [13–21] and Vanden Berghe and co-workers [10,11] constructed new types of Runge–Kutta formulas and multistep methods that are exact for some degrees of polynomials and also for some exponential and trigonometric functions. Their methods have truncation errors up to tenth order. In [16] an excellent summary of the multistep exponential methods is provided. Mickens [24,25] constructed exact finite difference schemes for some differential equations. Many type of differential equations have been solved by this method. In [23] Ramos and Lopez presented a nonstandard method including a frozen coefficient approximation which corresponds to keeping only the constant term of the Taylor series expansion of the equation, and schemes which include both the constant and linear terms of the Taylor series expansions.

In view of the ideas mentioned above, nonstandard finite difference formulas can be developed alternatively as follows:

Consider the differential equation:

$$y' = f(x, y).$$

Freezing f at $x = x_i$ in the interval (x_i, x_{i+1}) and adding the terms giving exponential kernel, we get:

$$\tilde{y}' - w\tilde{y} = f_i - w\tilde{y}_i, \tag{5}$$

where $\tilde{y}(x_i) = \tilde{y}_i$ and w is a constant in the interval (x_i, x_{i+1}). Imposing the conditions $\tilde{y}(x_i) = \tilde{y}_i$ and $\tilde{y}(x_{i+1}) = \tilde{y}_{i+1}$, we obtain:

$$\frac{\tilde{y}_{i+1} - \tilde{y}_i}{\frac{e^{wh}-1}{w}} = f_i.$$

Therefore, new forward difference formula is:

$$\delta_i^+\tilde{y} = \frac{\tilde{y}_{i+1} - \tilde{y}_i}{\frac{e^{wh}-1}{w}}. \tag{6}$$

The factor $\phi(w,h) = \frac{e^{wh}-1}{w}$ coincides with the works of Mickens [24]. Similar expressions are called artificial viscosity term in some articles and books [26,27]. Similarly, a new backward difference formula is obtained by imposing conditions $\tilde{y}(x_i) = \tilde{y}_i$ and $\tilde{y}(x_{i-1}) = \tilde{y}_{i-1}$:

$$\delta_i^-\tilde{y} = \frac{\tilde{y}_i - \tilde{y}_{i-1}}{\frac{1-e^{-wh}}{w}}. \tag{7}$$

The formulas (6) and (7) are exact for $y = e^{wx}$. Using these formulas the approximation to the second derivative is developed as follows:

$$\delta_i^2\tilde{y} = \delta_i^+\delta_i^-\tilde{y} = \frac{\tilde{y}_{i+1} - 2\tilde{y}_i + \tilde{y}_{i-1}}{\frac{2(\cosh(wh)-1)}{w^2}} \tag{8}$$

An alternative way of deriving (8) is to construct the differential equation:

$$\tilde{y}'' - w^2\tilde{y} = f_i - w^2\tilde{y}_i, \tag{9}$$

which is an approximation to $y'' = f(x, y)$ in the interval (x_{i-1}, x_{i+1}). The general solution of (9) is:

$$\tilde{y}(x) = c_1e^{wx} + c_2e^{-wx} + \tilde{y}_i - f_i/w^2.$$

Applying the conditions $\tilde{y}(x_{i-1}) = \tilde{y}_{i-1}, \tilde{y}(x_i) = \tilde{y}_i, \tilde{y}(x_{i+1}) = \tilde{y}_{i+1}$ we get three algebraic equations. After algebraic manipulations, (8) is obtained again.

3. Outline of the method

After the uniform space discretization we propose nonstandard discrete difference operator (8) for the second derivative in (1). As a result, nonstandard finite difference approximation of the bvp (1) and (2) becomes:

$$\begin{aligned} &F[i] := \frac{y_{i+1} - 2y_i + y_{i-1}}{\psi(w,h)} - f(x_i, y_i) = 0, \quad i = 1, 2, \ldots, N-1, \\ &y_0 = \alpha, \quad y_N = \beta, \end{aligned} \tag{10}$$

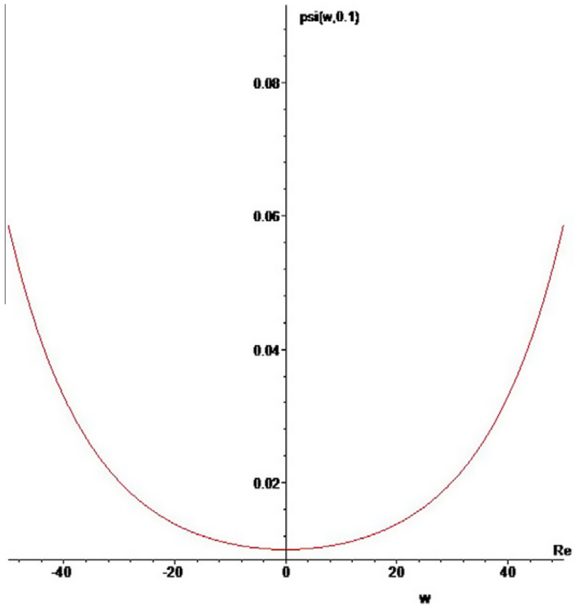

Fig. 1. Stepsize function $\psi(w, 0.1)$ versus real w.

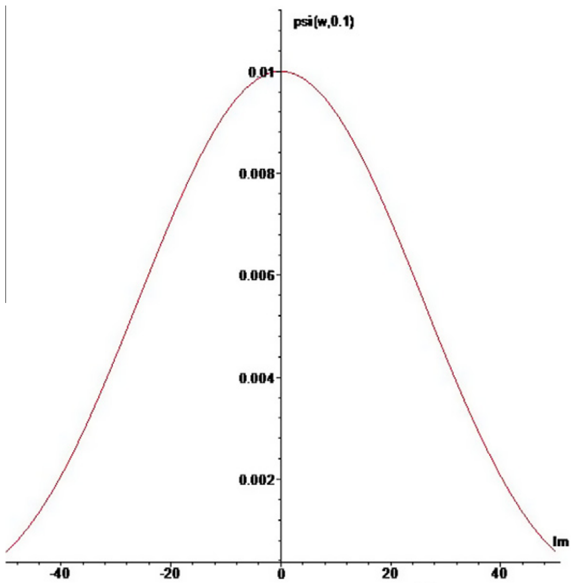

Fig. 2. Stepsize function $\psi(w, 0.1)$ versus pure imaginary w.

where the stepsize function $\psi(w,h) = \frac{2(\cosh(wh)-1)}{w^2}$ and w is a free parameter for now. In the Section 5, we will express w in terms of y_{i+1}, y_i and y_{i-1} in the interval (x_{i-1}, x_{i+1}). Therefore the notation w_i will be used instead of w.

In the Figs. 1 and 2 the stepsize function $\psi(w, 0.1)$ is plotted. Since there is the possibility of the occurence of pure imaginary w, ψ versus pure imaginary w is plotted in the Fig. 2. In both cases ψ is always real due to the fact that $cosh(xI) = cos(x)$ with $I^2 = -1$. When w or h are small, large cancellation errors may occur, so a Taylor series for ψ may be used.

$$\psi(w,h) = h^2 + \frac{h^4w^2}{12} + \frac{h^6w^4}{360} + \frac{h^8w^6}{20160} + \cdots$$

4. Some properties of the method

In this section, we introduce the algebraic order and truncation error of the present method. Finally some references are addressed for the stability issues.

Definition. [16]Any multistep method of the form:

$$\sum_{i=0}^{m} a_i y_{n+i} = h^r \sum_{i=0}^{m} b_i f(x_{n+i}, y_{n+i})$$

is called algebraic order (or exponential) of order p for the differential equation $y^r = f(x,y)$ if the associated linear operator $L(x) = \sum_{i=0}^{m} a_i u(x+ih) - h^r \sum_{i=0}^{m} b_i u^{(r)}(x+ih)$ vanishes for any linear combination of the linearly independent functions $1, x, x^2, \ldots, x^{p+r-1}$ (or $e^{w_0x}, e^{w_1x}, \ldots e^{w_{p+r-1}x}$) where w_i are real or complex number.

It is easily seen that the formula (8) is exact for any linear combination of the set $\{1, x, e^{wx}, e^{-wx}\}$. Therefore algebraic order and exponential order of the proposed method (8) are zero.

The local truncation error (*lte*) of the scheme (10) becomes:

$$\tau_i = f(x_i, y(x_i))(\frac{h^2}{\psi(w,h)} - 1) + \frac{h^4 y^{(iv)}(x_i)}{12\psi(w,h)} + \frac{h^6 y^{(vi)}(\eta_i)}{360\psi(w,h)}, \tag{11}$$

where $\eta_i \in (x_{i-1}, x_{i+1})$,

We make the following comments on the order of accuracy and consistency.

(i) In the limit as $w \to 0, \psi(w,h) \to h^2$. This reveals that the scheme (10) is reduced to the classical central discretization of (1) and (2).

(ii) For any value of w, $\phi(w,h) \approx h^2$ and $\psi(w,h) \to 0$ thus $\tau \to 0$ while $h \to 0$. In other words, the scheme (10) is consistent and second order accurate. But higher order accuracy will be gained in the Section 5. w will determined in such a way that the magnitude of the truncation error will be as small as possible. Namely, we will increase the order of the *lte* by manupilating w. Thus the accuracy of the method exceeds those of standard three point differences schemes.

For initial value problems the stability theory of the exponential fitting methods is well developed. Especially for highly oscillatory problems P stability is defined and studied. [16,22]. For the boundary value problems this stability question is less developed. In [16], Simos studied the stability of the exponential fitting method for schrodinger equation with other initial value problems. In [29], the stability of a nonlinear difference scheme which is obtained from discretization of a nonlinear boundary value problem is defined as uniform boundedness of the inverse Jacobian at the exact solution. After all, it is essentially this Jacobian matrix that is used in solving linear systems in the course of Newton's method, once we get very close to the solution.

5. Implementation of the method

In the previous section, w has been assumed constant in the interval (x_{i-1}, x_{i+1}) and independent of unknown function $y(x)$. Therefore, second order accuracy is obtained. Now we present an algorithm that gives higher accuracy to determine w depending on $y(x)$.

Our method is based on using the difference formula (8) for (1) and (2) and determining an optimal w for the given f. For a scheme to be locally exact, the *lte* (11) should be zero. This fact can be used to determine w. Setting the first and second term to zero (11) we get:

$$\tau_i = f(x_i, y(x_i))\left(\frac{h^2}{\psi(w,h)} - 1\right) + \frac{h^4 y^{iv}(\eta_i)}{12\psi(w,h)} = 0. \tag{12}$$

In this way we force the *lte* to become fourth order if all derivatives of $y(x)$ are computed exactly. we expand the term $\cosh(wx)$ in ψ to the Taylor series up to the fourth order so as to get an analytical expression for w:

$$w_i^2 \approx \frac{y^{iv}(x_i)}{f(x_i, y(x_i))}$$

w_i is allowed to be a pure imaginary number as well. For some form of f, special cases are observed. For instance, if f is only a linear function of x, then $w = 0$ and if f is only a linear function of y, then w_i will be constant in all subintervals. These cases correspond to classical finite difference scheme and exponentially fitted scheme [26] respectively.

w_i should be expressed in the scheme (10) in terms of y_{i-1}, y_i, y_{i+1} (unknown nodal values) in the interval (x_{i-1}, x_{i+1}). For this purpose, we simply express $y^{(iv)}$ in terms of only f and y' by continous Eq. (1) and then utilize approximations $y(x_i) = y_i$ and $y'(x_i) = \frac{y(x_{i+1})-y(x_{i-1})}{2h}$.

Although it may seem to the reader that all these computations and manipulations are very tedious, by the advantages of symbolic computation, we can automate the process for any f. In terms of numerical computations, there is no more load than that of the classical finite difference methods. However, numerical results are very accurate compared to the classical methods. In the next section, we introduce the schemes for each of examples as an output of Maple codes given below:

```
>scheme:=proc(f)
#righ hand side of the equation (1) is taken as an input#
global F, w;
derivative2:=f:derivative3:=diff(f,x):
diff(derivative3,x):derivative4:=algsubs(diff(u(x),x$2)=derivative2,%):
#Third and fourth derivatives of u are computed
in terms of first and second derivatives via differential equation#
simplify(derivative4/derivative2):
#Continuous formula is constructed for w#
algsubs(diff(u(x),x)=(y[i+1]-y[i-1])/(2*h),%): subs(u(x)=y[i],%):
subs(x=x[i],%): w:=sqrt(%):
#w is discretized#
algsubs(u(x)=y[i],f):discretef:=subs(x=x[i],%):
#right hand side of equation is discretized#
F[i]:=ŵ2*(y[i+1]-2*y[i]+y[i-1])-discretef*2*(cosh(w*h)-1)=0:
#Full nonstandard discretization of given differential equation#
print('w'=w):
print('F[i]'=F[i]):
return NULL:
end proc:
```

6. Numerical examples

In this section, the present method is applied to three different problems. The Maple procedure given in Section 5 produces difference schemes for each examples. They are presented in the appendix. The resulting systems of algebraic equations are solved by Newton method. In Example 3, special attention should be paid to obtain convenient initial guess since there exist many solutions.

6.1. Example 1

Consider the following nonlinear two point BVP, Troesch's problem [31]:

$$u''(x) - \lambda \sinh(\lambda u(x)) = 0, \quad 0 < x < 1, \tag{13}$$

with the boundary conditions $u(0) = 0$ and $u(1) = 1$. Troeschs problem arises from a system of nonlinear ordinary differential equations which occur in an investigation of the confinement of a plasma column by radiation pressure and it represents the well-known test problem for numerical software. Recently, the problem has been studied extensively [32–35]. Also, this problem is recommended by Boyd in [39] as a test problem in order to show efficiency of any numerical method. The closed form solution to this problem has been given in terms of the Jacobian elliptic function [32,35]. The Maple command:

```
scheme(lambda*sinh(lambda*u(x)));
```

gives the formula for w and the scheme corresponding to the Eq. (13). Inserting the boundary conditions, the resulting nonlinear system (see Appendix A) is solved to obtain discretized solutions.

In Tables 1 and 2, exact solutions for $\lambda = 0.5$ and $\lambda = 1$ and the numerical results of the present method compared with several existing methods [33–35] in the literature are presented. It is interesting that inaccurate tabulated "exact" solutions are given in [33,34]. If they had used the exact solution reported here and in [32,35] as a basis for comparison, they would

Table 1
Results for Troeschs problem (example 1) for $\lambda = 0.5$.

x	Exact solution	Decomposition method [34]	Homotopy pert. method [33]	Present method $N = 10$	Absolute errors of the present method
0.1	0.0959443493	0.0959383534	0.0959395656	0.0959443492	5. 357E−11
0.2	0.1921287477	0.1921180592	0.1921193244	0.1921287476	1. 036E−10
0.3	0.2887944009	0.2887803297	0.2887806940	0.2887944007	1. 467E−10
0.4	0.3861848464	0.3861687095	0.3861675428	0.3861848462	1. 795E−10
0.5	0.4845471647	0.4845302901	0.4845274183	0.4845471645	1. 989E−10
0.6	0.5841332484	0.5841169798	0.5841127822	0.5841332482	2. 015E−10
0.7	0.6852011483	0.6851868451	0.6851822495	0.6852011481	1. 857E−10
0.8	0.7880165227	0.7880055691	0.7880018367	0.7880165225	1. 491E−10
0.9	0.8928542161	0.8928480234	0.8928462193	0.8928542161	9. 120E−11

Table 2
Results for Troeschs problem (example 1) for $\lambda = 1$.

x	Exact solution	Decomposition method [34]	Homotopy pert. method [33]	Present method $N = 10$	Absolute errors of the present method
0.1	0.0846612565	0.084248760	0.0843817004	0.0846612556	9.303E−10
0.2	0.1701713582	0.169430700	0.1696207644	0.1701713565	1.704E−09
0.3	0.2573939080	0.256414500	0.2565929224	0.2573939059	2.176E−09
0.4	0.3472228551	0.346085720	0.3462107378	0.3472228528	2.232E−09
0.5	0.4405998351	0.439401985	0.4394422743	0.4405998333	1.804E−09
0.6	0.5385343980	0.537365700	0.5373300622	0.5385343971	9.103E−10
0.7	0.6421286091	0.641083800	0.6410104651	0.6421286094	3.015E−10
0.8	0.7526080939	0.751788000	0.7517335467	0.7526080954	1.467E−09
0.9	0.8713625196	0.870908700	0.8708835371	0.8713625215	1.868E−09

have found that their approximation methods were actually much more accurate than they realized. Numerical results of the present method are extremely accurate even for a coarse grid.

6.2. Example 2

Consider the equation:

$$u''(x) - u^2(x) = 2\pi^2 \cos(2\pi x) - \sin^4(\pi x), \quad 0 < x < 1, \tag{14}$$

subject to the boundary conditions $u(0) = u(1) = 0$ The exact solution of the problem is $u(x) = \sin^2(\pi x)$. This problem is utilized as a test example in many articles concerning with bvp's [4,36,38].

The Maple command:

```
scheme(u(x)^2+2*Pi^2*cos(2*Pi*x)-sin(Pi*x)^4);
```

gives the formula for w and the scheme corresponding to the Eq. (14) (see Appendix B.

In Table 3 absolute errors of the present method and classical Numerov method [28] are presented. Figures verify that the present method gives very accurate results.

6.3. Example 3

Consider the equation:

Table 3
Absolute errors of the Numerov method and present method for example 2.

x	Numerov method $N = 20$	Present method $N = 20$	Numerov method $N = 40$	Present method $N = 40$
0.1	2.57287E−06	4.48188E−07	1.61881E−07	2.86207E−08
0.2	1.14519E−05	4.19597E−07	7.16677E−07	2.66314E−08
0.3	2.28275E−05	6.41501E−07	1.42684E−06	4.16857E−08
0.4	3.21025E−05	1.12018E−06	2.00571E−06	7.39232E−08
0.5	3.56489E−05	1.14061E−06	2.22703E−06	7.6087E−08
0.6	3.21025E−05	1.12018E−06	2.00571E−06	7.39343E−08
0.7	2.28275E−05	6.415E−07	1.42684E−06	4.16998E−08
0.8	1.14519E−05	4.19598E−07	7.1668E−07	2.66432E−08
0.9	2.57287E−06	4.48189E−07	1.61882E−07	2.86266E−08

$$u'' + u^3 = 0. \tag{15}$$

This equation has periodic solutions expressed in terms of the Jacobi elliptic function for $-\infty < x < \infty$ [30]. The exact solution is:

$$u(x) = AJacobiSN\left(\left(\frac{\sqrt{2}}{2}x + B\right)A, i\right)$$

But as a boundary value problem with the conditions $u(0) = u(1) = 0$and $0 < x < 1$, it posseses many solutions [37]. The Maple command:

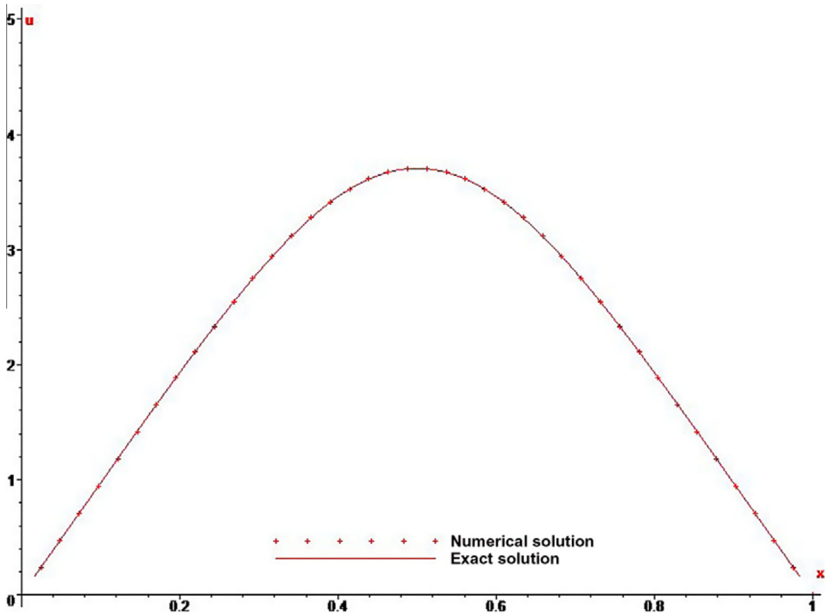

Fig. 3. Exact ($A = -3.708149356$ $B = 4.2426640686i$) and numerical solution of example 3 for $N = 41$ with initial guess RK solution of (15) with $u(0) = 0$ and $u'(0) = 9$.

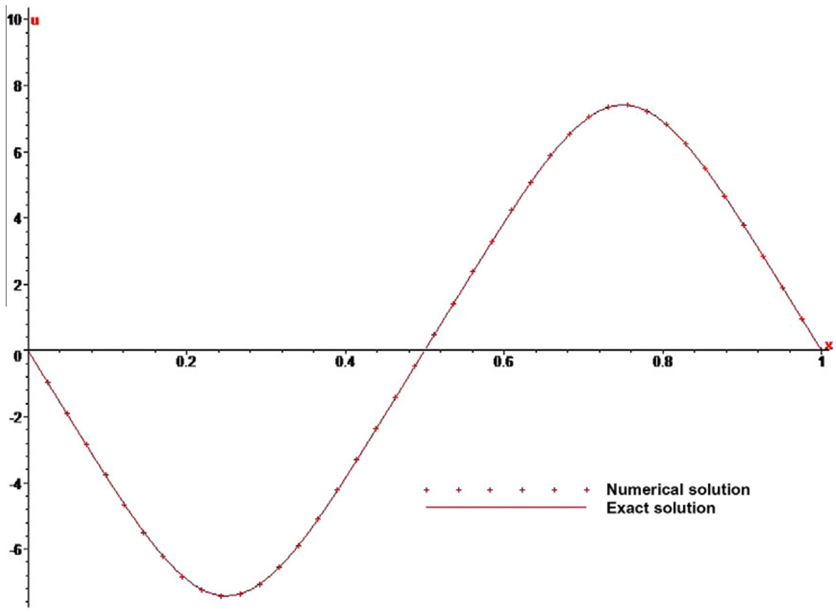

Fig. 4. Exact ($A = -7.4162987111i$ $B = -0.7071067810 - 0.7071067810i$) and numerical solution of example 3 for $N = 41$ with initial guess RK solution of (15) with $u(0) = 0$ and $u'(0) = -38$.

```
scheme(-u(x)^3);
```

gives the formula for w and the scheme corresponding to the Eq. (15) (see Appendix C).

Imposing the conditions $y_0 = y_N = 0$, the resulting discretized algebraic system has many solutions. Depending on initial guess of Newton method, we obtain different oscillatory solutions. In order to start with suitable initial guess, one can utilize a shooting method or exact solutions of the linearized equation of (15).

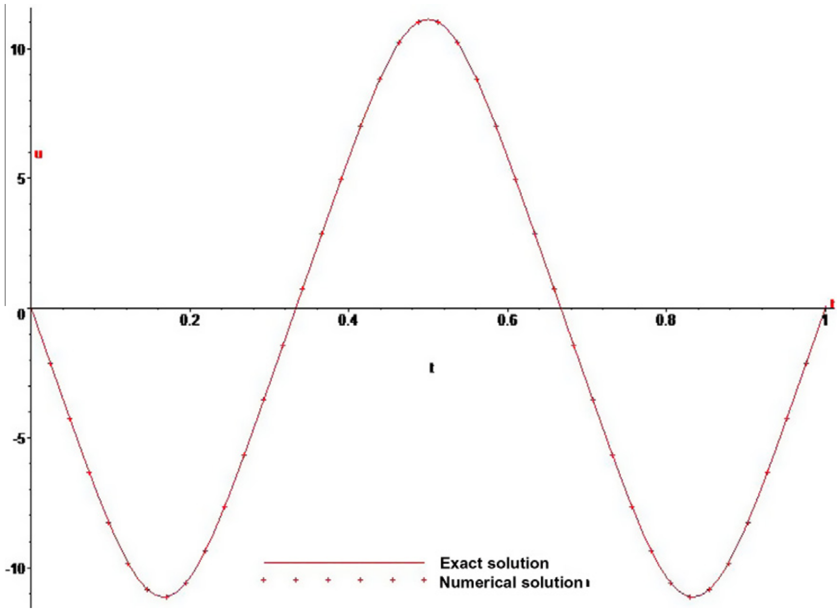

Fig. 5. Exact (A = 1112444807 B = −24.74873733) and numerical solution of example 3 for N = 41 with initial guess RK solution of (15) with $u(0) = 0$ and $u'(0) = 87$.

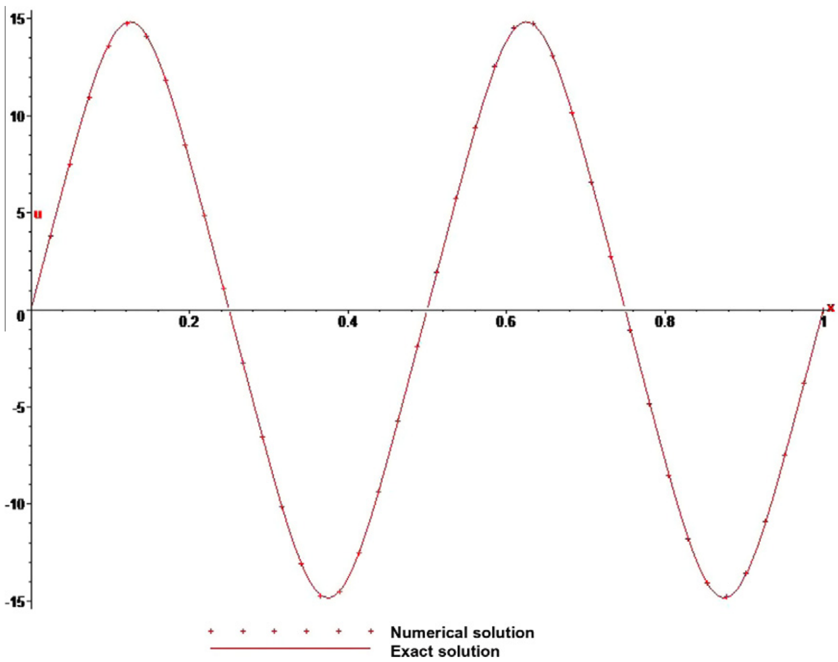

Fig. 6. Exact (A = −14.83259742 B = −0.3535533905−0.3535533905i) and numerical solution of example 3 for N = 41 with initial guess RK solution of (15) with $u(0) = 0$ and $u'(0) = 156$.

In Figs. 3–6, some exact solutions and corresponding numerical solutions with different initial guesses are plotted. Initial guesses are obtained from the Runge Kutta (RK) solutions of corresponding initial value problem of (15) with conditions $u(0) = 0$ and $u'(0) = 9, u'(0) = -38, u'(0) = -87, u'(0) = 156$, respectively. N is an odd number to prevent even-odd decoupling in this problem. Figures show that the present method is also very successful in handling oscillatory motion. It should be noted that classical finite difference methods fail to catch high oscillatory solutions even for the initial guesses which are very close to those oscillatory solutions.

7. Conclusion

A new kind of finite difference scheme is presented. It contains an artificial parameter determined by keeping the local truncation error small in magnitude. Symbolic computation is employed since it is tedious to construct formulas for each of *f*. Numerical examples shows that present method can mimic various behaviours.

Indeed the success of the present method depends upon artificial parameter w reflecting the behavior of unknown function. w is expressed in terms of the derivatives of the unknown function. For computational issues, those derivatives are discretized as well. This procedure provides a piecewise approximation to the problem. One should notice that the numerical values of the w_i's are determined after the whole discrete nonlinear system is solved. w might be real or pure imaginary. It determines whether the method is called exponential fitting or trigonometrical fitting. It is possible for w to be real in some parts of the domain and pure imaginary in the other parts of the domain.

The accuracy of the present method can be improved by including more terms of the expansion of $\cosh(wh)$ and equating more terms in the *lte* to zero and utilizing high order first derivative approximations in the discretization of w_i. However it is satisfactory to obtain these figures utilizing only three points in the difference scheme and equidistant mesh unlike [2,3].

Our future works will deal with different strategies for finding w since the present procedure imposes very strict derivative conditions on the solution. The extension of the present method to different problems is also crucial. For instance, majority of the papers concerning with exponential fitting methods for second order ode's consider the only case in which first derivative is absent. For the problem $x'' = f(t, x, x')$, exact integration of the functions e^{wx} and e^{-wx} will not work anymore. Therefore a new reference set is needed. As for PDE, different frequencies might be proposed for each of dimensions. It is possible to determine formulas for each frequencies by truncation errors in a similar way. It may be easy to construct those schemes for linear partial differential equations but it is a challenging work for nonlinear PDE s.

Acknowledgements

This work was supported by the Scientific and Technical Research Council of Turkey (TUBITAK).The authors would like to express their thanks to Dr. Fred W.Wubs for his valuable comments and advices.

Appendix A. *w* and *F* for example 1

$$w = \frac{1}{2}\sqrt{\frac{\lambda^2\left(y_{i+1}^2 - 2y_{i+1}y_{i-1} + y_{i-1}^2 + 4\cosh(\lambda y_i)h^2\right)}{h^2}},$$

$$F_i = \frac{1}{4}\frac{\lambda^2\left(y_{i+1}^2 - 2y_{i+1}y_{i-1} + y_{i-1}^2 + 4\cosh(\lambda y_i)h^2\right)(y_{i+1} - 2y_i + y_{i-1})}{h^2} - 2\lambda,$$

$$\sinh(\lambda y_i)$$

$$\left(\cosh\left(\frac{1}{2}\sqrt{\frac{\lambda^2\left(y_{i+1}^2 - 2y_{i+1}y_{i-1} + y_{i-1}^2 + 4\cosh(\lambda y_i)h^2\right)}{h^2}}h\right) - 1\right),$$

Appendix B. *w* and *F* for example 2

$$\begin{aligned} w = \frac{1}{2}\Big(&-2\Big(-4y_i^3h^2 - y_{i+1}^2 + 2y_{i+1}y_{i-1} - y_{i-1}^2 + 40\cos(\pi x_i)^2\pi^2h^2 - 32\cos(\pi x_i)^4\pi^2h^2 \\ &+4y_i\cos(\pi x_i)^4h^2 - 8y_i\pi^2\cos(2\pi x_i)h^2 + 4y_ih^2 - 8y_i\cos(\pi x_i)^2h^2 + 16\pi^4\cos(2\pi x_i)h^2 - 8\pi^2h^2\Big) \\ &\Big/\Big(\Big(y_i^2 + 2\pi^2\cos(2\pi x_i) - 1 + 2\cos(\pi x_i)^2 - \cos(\pi x_i)^4\Big)h^2\Big)\Big) \end{aligned}$$

$$F_i = -\frac{1}{2}\Big(-4y_i^3 h^2 - y_{i+1}^2 + 2y_{i+1}y_{i-1} - y_{i-1}^2 + 40\%2\,\pi^2 h^2 - 32\%3\,\pi^2 h^2 + 4y_i\%3\,h^2 - 8y_i\pi^2\%1\,h^2$$
$$+4y_i h^2 - 8y_i\%2\,h^2 + 16\pi^4\%1\,h^2 - 8\pi^2 h^2\Big)(y_{i+1} - 2y_i + y_{i-1})/\Big(\big(y_i^2 + 2\pi^2\%1 - 1 + 2\%2 - \%3\big)h^2\Big)$$
$$-2\Big(2\pi^2\%1 + y_i^2 - \sin(\pi x_i)^4\Big)\Bigg(\cosh\Bigg(\frac{1}{2}\Big(-2\Big(-4y_i^3 h^2 - y_{i+1}^2 + 2y_{i+1}y_{i-1} - y_{i-1}^2 + 40\%2\,\pi^2 h^2 - 32\%3\,\pi^2 h^2$$
$$+4y_i\%3\,h^2 - 8y_i\pi^2\%1\,h^2 + 4y_i h^2 - 8y_i\%2\,h^2 + 16\pi^4\%1\,h^2 - 8\pi^2 h^2\Big)$$
$$\Big/\Big(\big(y_i^2 + 2\pi^2\%1 - 1 + 2\%2 - \%3\big)h^2\Big)\Big)^{(1/2)}h\Bigg) - 1$$

$$\%1 := \cos(2\pi x_i),$$
$$\%2 := \cos(\pi x_i)^2,$$
$$\%3 := \cos(\pi x_i)^4.$$

Appendix C. *w* and *F* for example 3

$$w = \frac{1}{2}\sqrt{-\frac{6\left(-y_{i+1}^2 + 2y_{i+1}y_{i-1} - y_{i-1}^2 + 2y_i^4 h^2\right)}{y_i^2 h^2}}, \quad F_i - \frac{3}{2}\frac{\left(-y_{i+1}^2 + 2y_{i+1}y_{i-1} - y_{i-1}^2 + 2y_i^4 h^2\right)(y_{i+1} - 2y_i + y_{i-1})}{y_i^2 h^2}$$
$$+2y_i^3\left(\cosh\left(\frac{1}{2}\sqrt{-\frac{6\left(-y_{i+1}^2 + 2y_{i+1}y_{i-1} - y_{i-1}^2 + 2y_i^4 h^2\right)}{y_i^2 h^2}}\,h\right) - 1\right).$$

References

[1] H. Keller, Numerical Methods for Two-Point Boundary Value Problems, Blaisdell Publishing Company, Waltham, Mass, 1968.
[2] I.A. Tirmizi, E.H. Twizell, Higher-order finite difference methods for nonlinear second-order two-point boundary-value problems, Appl. Math. Lett. 15 (2002) 897–902.
[3] E.M.E. Elbarbary, M. El-Kady, Chebyshev finite difference approximation for the boundary value problems, Appl. Math. Comput. 139 (2003) 513–523.
[4] S.N. Ha, A nonlinear shooting method for two-point boundary value problems, Comput. Math. Appl. 42 (2001) 1411–1420.
[5] H. Liang, P. Weng, Existence of positive solutions for boundary value problems of second order functional difference equations, Nonlinear Anal. 326 (2007) 520–551.
[6] M. Cherpion, C. De Coster, P. Habets, A constructive monotone iterative method for second-order BVP in the presence of lower and upper solutions, Appl. Math. Comput. 123 (2001) 75–91.
[7] A. Dang Quang, Iterative method for solving a nonlinear boundary value problem, Appl. Math. Comput. 190 (2007) 1675–1682.
[8] B. Ahmad, J. Nieto, N. Shahzad, The Bellman–Kalaba–Lakshmikantham quasilinearization method for Neumann problems, J. Math. Anal. Appl. 257 (2001) 356–363.
[9] B. Ahmad, R. Ali Khan, P.W. Eloe, Generalized quasilinearization method for a second order three point boundary-value problem with nonlinear boundary conditions, Electron. J. Differ. Equat. 2002 (2002) 1–12.
[10] D. Hollevoet, M. Van Daele, C.G. Vanden Berghe, The optimal exponentially-fitted Numerov method for solving two-point boundary value problems, J. Comput. Appl. Math. 230 (2009) 260–269.
[11] G. Vanden Berghe, H. De Meyer, M. Van Daele, T. Van Hecke, Exponentially-fitted explicit Runge Kutta methods, Comput. Phys. Commun. 123 (1999) 715.
[12] G. Vanden Berghe, M. Van Daele, Trigonometric polynomial or exponential fitting-approach?, J Comput. Appl. Math. 233 (2009) 969–979.
[13] T.E. Simos, P.S. Williams, A new family of exponential fitting methods, Math. Comput. Model. 38 (2003) 571–584.
[14] A. Konguetsof, T.E. Simos, An exponentially-fitted and trigonometrically – fitted method for the numerical solution of periodic initial-value problems, Comput. Math. Appl. 45 (2003) 547–554.
[15] D.S. Vlachos, T.E. Simos, Algorithm 901: LMEF a program for the construction of linear multistep methods with exponential fitting for the numerical solution of ordinary differential equations, ACM Trans. Math. Softw. 37 (1) (2010). Article 12, January 2010.
[16] Z.A. Anastassi, T.E. Simos, Numerical multistep methods for the efficient solution of quantum mechanics and related problems, Phys. Rep. 482 (2009) 1–240.
[17] D.F. Papadopoulos, Z.A. Anastassi, T.E. Simos, An Optimized Runge–Kutta–Nystrom method for the numerical solution of the Schrodinger equation and related problems, MATCH - Commun. Math. Comput. Chem. 64 (2) (2010) 551–566.
[18] T.E. Simos, New closed Newton-cotes type formulae as multilayer symplectic integrators, J. Chem. Phys. 133 (10) (2010).
[19] I. Alolyan, T.E. Simos, High algebraic order methods with vanished phase-lag and its first derivative for the numerical solution of the Schrodinger equation, J. Math. Chem. 48 (2010) 925–958.
[20] S. Stavroyiannis, T.E. Simos, A nonlinear explicit two-step fourth algebraic order method of order infinity for linear periodic initial value problems, Comput. Phys. Commun. 181 (2010) 1362–1368.
[21] T. Monovasilis, Z. Kalogiratou, T.E. Simos, Symplectic partitioned Runge-Kutta methods with minimal phase-lag, Comput. Phys. Commun. 181 (7) (2010) 1251–1254.
[22] J.P. Coleman, L.Gr. Ixaru, *P*-stability and exponential-fitted methods for $y'' = f(x, y)$, IMA J. Numar. Anal. 16 (1996) 179–199.
[23] J.I. Ramos, C.M. Garcia-Lopez, Nonstandard finite difference equations for ODEs and 1-D PDEs based on piecewise linearization, Appl. Math. Comput. 86 (1997) 11–36.
[24] Ronald E. Mickens, Nonstandard finite difference schemes for differential equations, J. Differ. Equat. Appl. 8 (9) (2002) 823–847.

[25] R.E. Mickens, Nonstandard Finite Difference Models of Differential Equations, World Scientific, Singapore, 1994.
[26] H-G. Roos, M. Stynes, L. Tobiska, Numerical Methods for Singularly Perturbed Differential Equations, Springer-Verlag, Berlin, Heidelberg, 1996.
[27] Mohan K. Kadalbajoo, Puneet Arora, B-splines with artificial viscosity for solving singularly perturbed boundary value problems, Math. Comput. Model. 52 (2010) 654–666.
[28] Mark H. Holmes, Introduction to Numerical Methods in Differential Equations, Springer Science, New York, 2007.
[29] Randall J. LeVeque, Finite Difference Methods for Ordinary and Partial Differential Equations, SIAM, 2007.
[30] Ronald E. Mickens, Truly Nonlinear Oscillations, World Scientific Publishing Co. Pte. Ltd, Singopore, 2010.
[31] B.A. Troesch, A simple approach to a sensitive two-point boundary value problem, J. Comput. Phys. 21 (1976) 279.
[32] I.O. Govrilyuk, I.I. Lazurchak, V.L. Mokarav, A method with a controllable exponential convergence rate for nonlinear differential operator equations, Comput. Meth. Appl. Math. 9 (2009) 63–78.
[33] X. Feng, L. Mei, G. He, An efficient algorithm for solving Troesch's problem, Appl. Math. Comput. 189 (2007) 500–507.
[34] E. Deeba, S.A. Khuri, S. Xie, An algorithm for solving boundary value problems, J. Comput. Phys. 159 (2000) 125–138.
[35] Y. Lin, J.A. Enszer, M.A. Stadtherr, Enclosing all solutions of two point boundary value problems for ODE's, Comput. Chem. Eng. 32 (2008) 1714–1725.
[36] Lu-Feng Mo, Shu-Qiang Wang, A variational approach to nonlinear two-point boundary value problems, Nonlinear Anal. Theor. Methods Appl. (2008).
[37] P. Bailey, L. Shampine, P. Waltman, Nonlinear Two Point Bondary Value Problems, Academic Press, New York and London, 1968.
[38] Wei Jiang, Minggen Cui, Constructive proof for existence of nonlinear two-point boundary value problems, Appl. Math. Comput. 215 (2009) 1937–1948.
[39] John P. Boyd, One-point pseudospectral collocation for the one-dimensional Bratu equation, Appl. Math. Comput. 217 (2011) 5553–5565.