Explicit Numerov type methods with reduced number of stages.

CH. TSITOURAS*

Department of Applied Mathematics and Physical Sciences, National Technical University of Athens, Zografou Campus 15780, Athens, GREECE

ABSTRACT. We present in this paper a new approach for the derivation of hybrid explicit Numerov type methods. The new methodology does not require the intermediate use of high accuracy interpolatory nodes, since we only need the Taylor expansion of the internal points. As a consequence a sixth order method is produced at a cost of only four stages per step instead of six stages needed for the methods appeared in the literature until now. Numerical results over some well known problems in physics and mechanics indicate the superiority of the new method.

Keywords: Initial Value Problem, Numerical Solution, Two step methods, Hybrid methods.

AMS classification: 65L05, 65L06.

1. INTRODUCTION.

The initial value problem of second order

$$y'' = f(t, y), \quad y(t_0) = y_0, \quad y'(t_0) = y_0', \tag{1}$$

is of continued interest in many fields of celestial mechanics, quantum mechanics, scattering theory, theoretical physics and chemistry, and electronics (see [6, 7]).

When solving (1) numerically we have to pay attention in the algebraic order of the method used, since this is the main factor of achieving higher accuracy with lower computational cost, i.e. this is the main factor of increasing the efficiency of our effort. One of the most widely used method for solving (1) is the Numerov method which is fourth algebraic order. This method is implicit and its implementation involve computations of Jacobians and solutions of non-linear systems of equations, [1]. So many authors proposed explicit modifications of Numerov method. The algebraic orders achieved were at most only six, [2, 3, 8]. Recently Tsitouras and Simos [9], presented an explicit method of eighth algebraic order suitable for problems with oscillating solutions.

These methods require the evaluation of interpolatory off-step nodes. This technique increases the computational cost since the interpolation points share high accuracy too, something that is useless. So six stages are needed per step for a sixth order method while an eighth order method uses ten stages per step.

* URL address http://www.math.ntua.gr/people/tsitoura.html

That purposeless derivation of intermediate points is our motivation for considering another approach, similar to the one used for the construction of Runge-Kutta-Nystrom methods. Instead of spending much effort increasing the order of internal nodes we simply involve them in a scheme, where only the final result has to achieve the demanded order.

2. The new method

Let $h > 0$ and $t_n = t_0 + nh, \quad n = 0, 1, 2, ...$; We indent to construct a sixth order method for the approximation of y_{n+1}using values from two steps. i.e. $[t_{n-1}, t_n]$ and $[t_n, t_{n+1}]$. The available values are y_{n-1}, $y''_{n-1} = f_{n-1}$ and y_n while we get $y''_n = f_n = f(t_n, y_n)$ at a cost of one function evaluation.

Traditionally we also need three more values of second derivatives within the interval $[t_{n-1}, t_{n+1}]$ in order to form the required interpolant. These extra values ought to be of fourth algebraic order. If it was possible to derive them without cost then we could construct a sixth order method at a cost of four stages. Unfortunately we can not achieve this task since y_{n-1}, y''_{n-1}, y_n and y''_n are not enough information to give us interpolatory approximations of intermediate values of the desired accuracy. So the total cost increases to six stages.

Implementing the new method we only need y''_n and three extra function evaluations f_a, f_b and f_c. The new method has the form:

$$
\begin{aligned}
f_n &= f(t_n, y_n) \\
y_a &= -\frac{1}{2}y_{n-1} + \frac{3}{2}y_n + h^2\left(\frac{1}{16}f_{n-1} + \frac{5}{16}f_n\right), \; f_a = f\left(t_n + \frac{1}{2}h, y_a\right), \\
y_b &= \frac{1}{2}y_{n-1} + \frac{1}{2}y_n + h^2\left(-\frac{7}{144}f_{n-1} - \frac{5}{48}f_n + \frac{1}{36}f_a\right), \; f_b = f\left(t_n - \frac{1}{2}h, y_b\right), \\
y_c &= -y_{n-1} + 2y_n + h^2\left(-\frac{2}{9}f_{n-1} + \frac{1}{3}f_n + \frac{2}{9}f_a + \frac{2}{3}f_b\right), \; f_c = f(t_n + h, y_c), \\
y_{n+1} &= -y_{n-1} + 2y_n + h^2\left(\frac{1}{60}f_{n-1} + \frac{13}{30}f_n + \frac{4}{15}f_a + \frac{4}{15}f_b + \frac{1}{60}f_c\right).
\end{aligned}
$$

Using a symbolic manipulation package we can find the Taylor series expansions of f_a, f_b, f_c, y_{n+1} and $y(t_n + h)$. Matching the corresponding expansions up to h^7, we arrive at an expression of the form, [4]:

$$
h^2(c_{21}F_{21}) + h^3(c_{31}F_{31}) + ... + h^7(c_{71}F_{71} + ... + c_{7,10}F_{7,10}), \tag{2}
$$

where c_{ij} are expressions of the coefficients of the method while F_{ij} are elementary differentials with respect to y', f and $f^{(k)} = \frac{\partial^k f}{\partial t^k}$, $k = 1, 2, ..., 5$. Demanding $c_{21} = c_{31} = ... = c_{7,10} = 0$ we may derive the coefficients of a sixth order method.

Special care had to be taken when dealing with autonomous systems of ODEs (otherwise add $x'' = 0$, [5, pg 286]). If we have to solve a scalar ODE of the type

(1) then some of the elementary differentials appearing in (2) may compress. For example $f'f''y'^2 = f''f'y'^2$ for scalar equations while this is not true for systems of ODEs since f', f'', y' are matrices. So a non scalar function was considered for our computations (see appendix).

The truncation error of the new method for the scalar case is:

$$\begin{aligned} LTE &= h^8(-0.000066ff'^3 + 0.000372f^2f'f'' + 0.000810f'^2y'^2f'' \\ &+0.003373fy'^2f''^2 - 0.000124f^3f^{(3)} - 0.000148ff'y'^2f^{(3)} \\ &+0.001388y'^4f''f^{(3)} - 0.000372f^2y'^2f^{(4)} - 0.000173f'y'^4f^{(4)} \\ &-0.000124fy'^4f^{(5)} - 8.3\cdot 10^{-6}y'^6f^{(6)}) + O\left(h^9\right). \end{aligned}$$

The maximum absolute value of the coefficients appeared in the expression above is $3.37\cdot 10^{-3}$. The corresponding value for the sixth order method of Chawla & Rao [3], is $4.74\cdot 10^{-3}$.

3. NUMERICAL RESULTS

To illustrate our new sixth order method and to compare it with Chawla & Rao method [3], we consider two examples that are well known in the literature.

First we solved the two body gravitational problem,

$$\begin{aligned} y_1'' &= -\frac{y_1}{\sqrt{y_1^2+y_2^2}^{\,3}}, \\ y_2'' &= -\frac{y_2}{\sqrt{y_1^2+y_2^2}^{\,3}}, \\ y_1(0) &= 1/2, \quad y_2(0)=0, \quad y_1'(0)=0, \quad y_2'(0)=\sqrt{3}, \end{aligned}$$

for $t\in[0,6\pi]$. The absolute end point errors e are recorded and the values $-\log_{10}(e)$ are given in Table 1 for the same costs for both methods. We observe that the new method gains about one decimal digit for the various function evaluations used for the integration.

Then we consider the Duffing equation forced by a harmonic function [10],

$$y'' + y + y^3 = 0.002\cos(1.01t), \quad y(0) = 0.200426728067, \quad y'(0)=0,$$

for $t\in[0, 20.5\pi/1.01]$. We recorded again the same values in Table 2. More than one decimal digit is the gain now even if the method [3] was especially designed for periodic problems.

Table 1: Accurate digits for the two body problem

	stages								
	1200	1800	2400	3000	3600	4200	4800	5400	6000
C-R87	3.3	4.2	4.9	5.5	5.9	6.3	6.6	6.9	7.2
NEW	4.0	5.1	5.8	6.5	7.0	7.4	7.7	8.0	8.3

Table 2: Accurate digits for Duffing equation

	stages								
	600	900	1200	1500	1800	2100	2400	2700	3000
C-R87	4.3	5.4	6.1	6.7	7.1	7.6	7.9	8.2	8.5
NEW	5.4	6.5	7.2	7.8	8.3	8.7	9.1	9.4	9.7

4. Conclusion

A new approach for the derivation of two step hybrid methods was presented here. The results of the method we produced were very promising. However a more general theory is needed for a serious study of this case. Order conditions for the coefficients ought to be provided for better understanding the procedure. We hope to be able for this task in the very near future.

5. Appendix

Following the guidelines given in [4], we consider the system of ODEs

$$x'' = f\,(t, x, y)\,,\ y'' = g\,(t, x, y) \tag{3}$$

There is no need to use more equations in order to simulate the behavior of the method for a system of equations. Actually we may drop even t from the equations (3), since adding equation $x'' = 0$, we may avoid the independent variable.

We expand in Taylor series all the internal stages and finally we subtract the numerical approximation from the theoretical solution. We then expand in Taylor series this difference and we found an $O\,(h^8)$accuracy.

The mathematica [11], justification of the order of the method is now straightforward.

Mathematica Program

```
Clear["y*","x*","d*"];
(* 6th order, 4 stages, Numerov type method *)
(* second stage *)
x1=Simplify[Normal[Series[
{-1/2,3/2,1/16,5/16}.{x[t-h],x[t],h^2*f[x[t-h],y[t-h]],h^2*f[x[t],y[t]]},
{h,0,5}]]];
y1=Simplify[Normal[Series[
{-1/2,3/2,1/16,5/16}.{y[t-h],y[t],h^2*g[x[t-h],y[t-h]],h^2*g[x[t],y[t]]},
```

```
{h,0,5}]]];
(* third stage *)
x2=Simplify[Normal[Series[
{1/2,1/2,-7/144,-5/48,1/36}.{x[t-h],x[t],h^2*f[x[t-h],y[t-h]],h^2*f[x[t],y[t]],h^2*f[x1,y1]},
{h,0,5}]]];
y2=Simplify[Normal[Series[
{1/2,1/2,-7/144,-5/48,1/36}.{y[t-h],y[t],h^2*g[x[t-h],y[t-h]],h^2*g[x[t],y[t]],h^2*g[x1,y1]},
{h,0,5}]]];
(* fourth stage *)
x3=Simplify[Normal[Series[
{-1,2,-2/9,1/3,2/9,2/3}.
{x[t-h],x[t],h^2*f[x[t-h],y[t-h]],h^2*f[x[t],y[t]],h^2*f[x1,y1],h^2*f[x2,y2]},
{h,0,5}]]];
y3=Simplify[Normal[Series[
{-1,2,-2/9,1/3,2/9,2/3}.
{y[t-h],y[t],h^2*g[x[t-h],y[t-h]],h^2*g[x[t],y[t]],h^2*g[x1,y1],h^2*g[x2,y2]},
{h,0,5}]]];
(* Difference: Theoretical minus Numerical Solution for x[t]*)
dif=x[t+h]-
{-1,2,1/60,13/30,4/15,4/15,1/60}.
{x[t-h],x[t],h^2*f[x[t-h],y[t-h]],h^2*f[x[t],y[t]],h^2*f[x1,y1],h^2*f[x2,y2],h^2*f[x3,y3]};
(* Taylor expansion of difference *)
dif=Simplify[Normal[Series[dif,{h,0,7}]]];
(* convert all expansions with respect to x,x',f,f',f'',...,y,y',g,g',g'',... etc *)
dif=Expand[dif/.D[x[t],{t,8}]->D[f[x[t],y[t]],{t,6}]];
dif=Expand[dif/.D[x[t],{t,7}]->D[f[x[t],y[t]],{t,5}]];
dif=Expand[dif/.D[x[t],{t,6}]->D[f[x[t],y[t]],{t,4}]];
dif=Expand[dif/.D[x[t],{t,5}]->D[f[x[t],y[t]],{t,3}]];
dif=Expand[dif/.D[x[t],{t,4}]->D[f[x[t],y[t]],{t,2}]];
dif=Expand[dif/.D[x[t],{t,3}]->D[f[x[t],y[t]],{t,1}]];
dif=Expand[dif/.D[x[t],{t,2}]->f[x[t],y[t]]];
dif=Expand[dif/.D[y[t],{t,8}]->D[g[x[t],y[t]],{t,6}]];
dif=Expand[dif/.D[y[t],{t,7}]->D[g[x[t],y[t]],{t,5}]];
dif=Expand[dif/.D[y[t],{t,6}]->D[g[x[t],y[t]],{t,4}]];
dif=Expand[dif/.D[y[t],{t,5}]->D[g[x[t],y[t]],{t,3}]];
dif=Expand[dif/.D[y[t],{t,4}]->D[g[x[t],y[t]],{t,2}]];
dif=Expand[dif/.D[y[t],{t,3}]->D[g[x[t],y[t]],{t,1}]];
dif=Expand[dif/.D[y[t],{t,2}]->g[x[t],y[t]]];
dif=Expand[dif/.D[x[t],{t,2}]->f[x[t],y[t]]];
Print[dif]; (* Print zero result as difference *)
```

References

[1] M. M. Chawla and P. S. Rao, Numerov type method with minimal phase lag for the integration of second order periodic initial value problems, *J. Comput. Appl. Math.* **11** (1984) 277-281.

[2] M. M. Chawla and P. S. Rao, Numerov type method with minimal phase lag for the integration of second order periodic initial value problems II. Explicit method, *J. Comput. Appl. Math.* **15** (1986) 329-337.

[3] M. M. Chawla and P. S. Rao, An explicit sixth-order method with phase-lag of order eight for $y'' = f(t,y)$, *J. Comput. Appl. Math.* **17** (1987) 365-368.

[4] E. Fehlberg, Classical eighth and lower order Runge-Kutta-Nyström formulas with stepsize control for special second order differential equations, NASA Tech. Rep. TR R-381, March 1972, G. C. Marsal Space Flight Center, Ala. 35812.

[5] E. Hairer, S. P. Norsett and G. Wanner, Solving Ordinary Differential Equations I, *Springer-Verlag,* Berlin, 1993.

[6] L. D. Landau and F. M. Lifshitz, Quantum Mechanics, *Pergamon Press*, New York, 1965.

[7] R. L. Liboff, Introductory Quantum Mechanics, *Addison-Wesley*, Reading MA, 1980.

[8] T. E. Simos, Explicit two-step methods with minimal phase-lag for the numerical integration of special second order initial value problems and their application to the one dimensional Schrodinger equation, *J. Comput. Appl. Math.* **39** (1992) 89-94.

[9] Ch. Tsitouras and T. E. Simos, Explicit high order methods for the numerical integration of periodic initial value problems, *Appl. Math.& Comput.* **95** (1998) 15-26.

[10] R. van Dooren, Stabilization of Cowell's classical finite difference methods for numerical integration, *J Comput. Phys.* **16** (1974) 186-192.

[11] S. Wolfram, Mathematica, A system for Doing Mathematics by Computer, Second Edition, Addison-Wisley, Redwood City, CA, 1991.