PIECEWISE NTH ORDER ADOMIAN POLYNOMIAL STIFF DIFFERENTIAL EQUATION SOLVER

Amir A. Mobasher, Alabama A&M University; Arastoo Poor Biazar, University of Alabama in Huntsville; Zhengtao Deng, Alabama A&M University

Abstract

A piecewise nth order Adomian polynomial solver for initial value differential equations capable of solving highly stiff problems is presented here. This powerful technique which employs Adomian polynomials is shown to obtain nearly exact solutions for the benchmark cases studied. The high accuracies at reduced computational cost is obtained by varying the integration time and the order of the polynomials. The time step in the current solver is on the order of 10 to 30 times larger than those used by other standard techniques such as fourth-order Runge-Kutta. It was found that the piecewise continuous polynomials increases the time-step requirements as the order of polynomials increases. The current algorithm utilizes high-order Adomian polynomials to advance the solution within the stiff region with relatively large time steps. The algorithm was validated against a first-order system with and without large stiffness and a second-order, non-linear flame propagation. Results were compared with exact solutions using the Lambert W-function and fourth-order Runge-Kutta methods. Good agreement was obtained for the flame-propagation prediction. It was found that while obtaining relatively large time steps, accuracies on the order of 10^{-15} were obtained.

Introduction

A number of applications in science and engineering demand solutions of stiff differential equations. Stiffness is generally defined as the solution space that contains (very) large gradients. Such examples include problems in chemical kinetics, atmospheric sciences, biochemistry, electronics and automatic control systems, to name a few. In flame propagation, concentration of chemical species can decay at different rates; thus, a kinetic-reaction differential equation describing the species concentration usually has a broad range of time constants. The solution to this system of ordinary differential equations is dominated by the species that have the shortest time constants [1]. The stiffness of the problem requires that the error introduced during the computation be damped by the algorithm. The step size has to be extremely small. Larger step sizes may cause numerical accuracy and stability problems. In general, stiff problems may be solved by some differential equation solvers if the computational cost is not an issue. For large-scale engineering problems involving flame propagation [2], the solution of stiff problems becomes a matter of efficiency.

Currently, many elaborate schemes for the solution of stiff differential equations exist [3]. Perhaps the most widely accepted technique is the Gear [4] method. Other methods such as the Sin-Cos-Taylor-Like [5] method and the Multi-step Runge-Kutta method [6] have been used by other authors. Commercial software programs such as Matlab [2] use modifications to Runge-Kutta to address stiffness. Until recently, one overlooked approach is the method proposed by Adomian for the solution of non-linear and linear ordinary differential equations [7-11]. This approach was also employed to address a classic fluid-dynamics problem with high accuracy [12], [13]. A number of other physical problems have been recently solved by this methodology [14-16]. The Adomian Decomposition Method requires term-by-term differentiation and integration of the basic differential equations. Although this approach produces highly accurate solutions to non-linear differential equations, it has some drawbacks: 1) It requires laborious work by the user, due to successive differentiation and integration of the resulting Adomian Polynomials; 2) For stiff problems, the solution can only be partially convergent [12]; and 3) the approach is problem-specific [7].

The goal of this current study was to formulate an algorithm that would address the aforementioned issues. The authors refer to the current formulation as "Piecewise nth order Adomian Polynomial Differential Equation Solver" or, in short, Adomian Differential Equation Solver (ADES). The word "piecewise" means to break up the domain into sub-domains in order to obtain a series solution in each sub-domain utilizing Adomian polynomials of a specified order. Here, a new methodology for the solution of stiff differential equations will be presented. It will be shown that the current technique will obtain nearly exact solutions while maintaining large Δt's (as much as 30 times larger than the standard fourth-order Runge-Kutta technique). The current technique was validated against three test examples with a highly stiff nature and it was found that the solution obtained via this technique was highly accurate while the time required to obtain the solution was minimal.

Methodology

Consider the differential equation system of the form

$$\dot{u} = f(t,u) \tag{1}$$

where $f(t,u)$ is any linear or non-linear function of order n. Adomian polynomials are employed to construct the solution space for u as:

$$\begin{aligned} u &= u_0 + L^{-1}(f) \\ L^{-1} &= \int_{t1}^{t}(\cdot)dt \end{aligned} \tag{2}$$

where L^{-1} is the integration operator. The solution of the system of differential equations in Equation (1) is then expressed as:

$$u = \begin{bmatrix} u_1 \\ u_2 \\ \dots \\ u_n \end{bmatrix} = \begin{bmatrix} \sum_{j=0}^{\infty} u_{1,j} \\ \sum_{j=0}^{\infty} u_{2,j} \\ \dots \\ \sum_{j=0}^{\infty} u_{n,j} \end{bmatrix} \tag{3}$$

where $u_{i,0} = u_i(0)$ is the prescribed initial conditions and $u_{i,j+1} = L^{-1}(A_{i,j}(u_{i,0}, u_{i,1}, \dots, u_{i,j})$ is the $j+1^{th}$ term of the i^{th} equation and $A_{i,j}$ is the Adomian Polynomial [7].

This method, proposed by Adomian, integrates the Adomian polynomials from $t = 0$ to $t = \infty$. For stiff problems, the solution is valid only up to a certain time after which the solution begins to diverge. In practical terms, polynomials of order 100 and higher are needed to approximate the solution in the high gradient region. Obviously, this is beyond computational means and efficiency. To address this situation, the current technique divides the domain to m sub-domains—hereafter referred to as Piecewise Adomian Polynomials. The solution advances by inserting the end-conditions at each sub-domain as initial conditions for the subsequent sub-domain. The solution within each sub-domain is approximated by lower-order Adomian polynomials (typically orders of 2 to 20).

Results and Discussions

In this section, three well-defined problems that are highly stiff and have been referenced by other authors in benchmarking stiff differential equations will be examined.

Example 1: First-Order Linear Systems

This example considers a first-order stiff system with no non-linear terms. This problem has been examined by Ahmad [5] and is referenced here again in Equation (4).

$$\begin{cases} \dot{u} = -100u + 99v \\ \dot{v} = -v \end{cases} \quad \text{with} \quad \begin{pmatrix} u_0 \\ v_0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \tag{4}$$

Adomian polynomials take the form of

$$\begin{cases} r_0 = u_0 = 0 \\ s_0 = v_0 = 1 \end{cases} \tag{5}$$

Subsequent values of r and s are presented in Equation (6),

$$\begin{cases} r_{i+1} = -100r_i + 99s_i \\ s_{i+1} = -s_i \end{cases} \tag{6}$$

with the solution u, v presented in Equation (7).

$$\begin{cases} u = \sum_{n=0}^{\infty} r_n \frac{t^n}{n!} \\ v = \sum_{n=0}^{\infty} s_n \frac{t^n}{n!} \end{cases} \tag{7}$$

Table 1 compares the relative error for three different techniques of fourth-order Runge-Kutta (RK4), Sin-Cos-Taylor-Like (SCTL) solutions and the current formulation of the Adomian Differential Equation Solver (ADES). As shown in Table 1, RK4 generates relative errors in the range of 10^{-3} to 10^{-4} within the stiff region and SCTL generates errors of 10^{-5} to 10^{-12}, while ADES generates errors of order 10^{-12}. It must be noted that the current technique uses a Δt of 0.3, which is 30 times higher than those of RK4 and SCTL.

Table 1. Comparison of Relative Error for Three Techniques

Time(s)	Exact	RK4	SCTL	ADES
0	0	0	0	0
0.1	0.90479201811	10^{-3}	10^{-5}	10^{-12}
0.2	0.81873075102	10^{-4}	10^{-9}	10^{-12}
0.3	0.74081822068	10^{-6}	10^{-12}	10^{-12}
0.4	0.67032004604	10^{-5}	10^{-12}	10^{-12}
0.5	0.60653065971	10^{-6}	10^{-12}	10^{-12}
0.6	0.54881163609	10^{-8}	10^{-12}	10^{-12}
0.7	0.49658530379	10^{-8}	10^{-12}	10^{-12}
0.8	0.44932896412	10^{-9}	10^{-12}	10^{-12}
0.9	0.40656965974	10^{-10}	10^{-12}	10^{-12}
1.0	0.36787944117	10^{-11}	10^{-12}	10^{-12}

Example 2: First-Order Systems with large Stiffness

This example also considers a linear problem, though with large stiff regions. The ordinary differential equation (ODE) system can be described as

$$\begin{cases} \dot{u} = 998u + 1998v \\ \dot{v} = -999u - 1999v \end{cases} \quad (8)$$

To obtain the ADES solution, the initial conditions were set equal to vectors r_0 and s_0.

$$\begin{cases} r_0 = u_0 \\ s_0 = v_0 \end{cases} \quad (9)$$

Since all coefficients of r can be factored out, the general expression for r and s, as in Equation (10), can be written as:

$$\begin{cases} r_{i+1} = \quad 999r_i + 1998s_i \\ s_{i+1} = -999r_i - 1999s_i \end{cases} \quad (10)$$

And the solution u, v is given by Equation (11).

$$\begin{cases} u = \sum_{n=0}^{\infty} r_n \frac{t^n}{n!} \\ v = \sum_{n=0}^{\infty} s_n \frac{t^n}{n!} \end{cases} \quad (11)$$

Table 2 compares the relative errors between RK4 and ADES. The relative error generated by RK4 within the stiff region is in the range of 10^{-3} to 10^{-5}, while the current method generates errors of order less than 10^{-15} in all the regions.

Table 2. Comparison of Relative Error Between RK4 and ADES

Time (s)	Exact	RK4	ADES
0	0	0	0
0.001	1.63012155849531	10^{-3}	$< 10^{-15}$
0.002	1.86066871409805	10^{-3}	$< 10^{-15}$
0.003	1.94422192263888	10^{-3}	$< 10^{-15}$
0.004	1.97370033979925	10^{-4}	$< 10^{-15}$
0.005	1.98328701138628	10^{-4}	$< 10^{-15}$
0.006	1.98555717593120	10^{-4}	$< 10^{-15}$
0.007	1.98513700390092	10^{-5}	10^{-14}
0.008	1.98372836704622	10^{-5}	10^{-14}
0.009	1.98195734774168	10^{-5}	10^{-14}
0.010	1.98005426756857	10^{-6}	10^{-14}
0.020	1.96039734455236	10^{-10}	10^{-14}
0.030	1.94089106709692	10^{-14}	10^{-14}
0.040	1.92350141829273	$< 10^{-15}$	10^{-14}
0.050	1.90436225939701	$< 10^{-15}$	10^{-14}
0.060	1.88541353831420	$< 10^{-15}$	10^{-14}
0.070	1.86665336015640	$< 10^{-15}$	10^{-14}
0.080	1.848079848890170	10^{-14}	10^{-14}
0.090	1.82969114714890	10^{-14}	10^{-14}

Example 3: Flame Propagation

The final example in this section considers the problem of “Flame Propagation”, which is also solved by the Matlab differential equation solver RK4. The problem is described here again for reference in Equation (12)

$$\begin{cases} \dot{u} = v - uv \\ \dot{v} = 2v(u - v) \end{cases} \quad \text{with} \quad \begin{pmatrix} u_0 \\ v_0 \end{pmatrix} = \begin{pmatrix} \delta \\ \delta^2 \end{pmatrix} \quad (12)$$

where $\delta = 0.001$.

The exact analytical solution to the flame model is given by

$$v(t) = \frac{1}{1 + W(a\, e^{a-t})} \quad (13)$$

where $a = \frac{1}{\delta} - 1$ and the function $W(z)$ is the Lambert W function.

Unlike cases 1 and 2, this example involves second-order non-linear terms. The construction of Adomian Polynomials for this problem is shown in Equations 14-17.

$$\begin{cases} r_0 = u_0 \\ s_0 = v_0 \end{cases} \quad (14)$$

$$\begin{cases} r_1 = s_0 - r_0 s_0 \\ s_1 = 2s_0(r_0 - s_0) \end{cases} \quad (15)$$

$$\begin{cases} r_2 = s_1 - r_0 s_1 - r_1 s_0 \\ s_2 = 2s_0(r_1 - s_1) + 2s_1(r_0 - s_0) \end{cases} \quad (16)$$

$$\begin{cases} r_3 = s_2 - r_0 s_2 - r_1 s_1 - r_2 s_0 \\ s_3 = 2s_0(r_2 - s_2) + 2s_1(r_1 - s_1) + 2s_2(r_0 - s_0) \end{cases} \quad (17)$$

And the general form of the above equations is represented in Equation (18)

$$\begin{cases} r_{i+1} = s_i - \sum_{j=0}^{i} r_j s_{i-j} \\ s_{i+1} = 2\sum_{j=0}^{i} s_j \left(r_{i-j} - s_{i-j}\right) \end{cases} \quad (18)$$

with the solution shown in Equation (19).

$$\begin{cases} u = \sum_{n=0}^{\infty} r_n \frac{t^n}{n!} \\ v = \sum_{n=0}^{\infty} s_n \frac{t^n}{n!} \end{cases} \quad (19)$$

Figure 1 depicts the solution for this problem with h=1.5. It is shown that at t =1000, the solution undergoes a very steep gradient. Table 3 compares the solution of the flame-propagation problem with three different methods: the Piecewise Adomian (PAP) solver, the exact solution obtained by Maple software with Lambert W-function and fourth-order Runge-Kutta method. The solution time domain listed in Table 3 is from 1000 to 1018 seconds, where the stiff region exists in this time domain. For $n=5$, it can be seen that the relative error is within acceptable computational limits and the solution quickly converges to a nearly exact solution with the Lambert W-function.

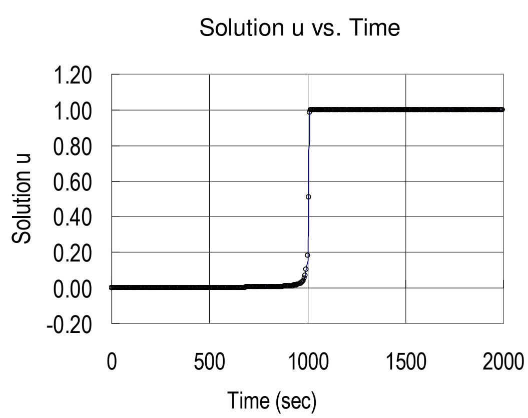

Figure 1. The Solution for the Flame-Propagation Problem with h=1.5

Conclusions

In this paper, a robust, accurate new technique was presented for the solution of stiff differential equations. The algorithm is based on the Adomian polynomials. It was shown that Δt's as much as 30 times higher than those of standard techniques of fourth-order Runge-Kutta can be used to obtain nearly exact solutions to highly stiff differential equations. One characteristic of the current algorithm is this ability to vary the order of polynomial to be used to increase the accuracy.

Table 3. Comparison of Relative Errors for Different Orders of Adomian Polynomials

Time (sec)	Exact With L-W Function	PAP (n=5)	RK4
1000	0.1840	0.1839	0.1845
1001	0.2164	0.2156	0.2164
1002	0.2592	0.2580	0.2592
1003	0.3181	0.3163	0.3180
1004	0.3999	0.3976	0.3997
1005	0.5118	0.5116	0.5115
1006	0.6519	0.6520	0.6515
1007	0.7945	0.7986	0.7940
1008	0.9006	0.9057	0.9001
1009	0.9583	0.9615	0.9579
1010	0.9838	0.9842	0.9834
1011	0.9940	0.9945	0.9937
1012	0.9978	0.9979	0.9976
1013	0.9992	0.9992	0.9991
1014	0.9997	0.9997	0.9997
1015	0.9998	0.9998	0.9999
1016	0.9999	0.9999	1.0000
1017	0.9999	0.9999	1.0000
1018	0.9999	0.9999	1.0000

References

[1] Kuo, K. K., (1986). *Principles of Combustion*. John Wiley.

[2] Moler, C. (n.d.). Stiff Differential Equations, http://www.mathworks.com/company/newsletters/articles/stiff-differential-equations.htm

[3] Biography for Stiff Differential Equations. (2003). Retrieved from http://math.fullerton.edu/mathews/n2003/stiffde/StiffDEsBib/Links/StiffDEsBib_lnk_3.htm

[4] Gear, C. W., & Petzold, L. R. (1984), ODE systems for the solution of differential algebraic systems, *SIAM Journal on Numerical Analysis*, *21*, 716-728.

[5] Ahmad, R., & Yaacob, N. (2005). Sin-Cos-Taylor-Like Method for Solving Stiff Differential Equation. *Journal of Fundamental Sciences, 1*, 34-43.

[6] Chapra, S. C., & Canale, R. P., (2010). *Numerical Methods for Engineers*. (6^{th} ed.). McGraw Hill.

[7] Adomian, G. (1989). *Nonlinear stochastic Systems Theory and Applications to Physics*. Kluwer Academic.

[8] Adomian, G. (1983). *Stochastic systems*. Academic Press, London.

[9] Adomian, G., (1984). A new approach to non-linear partial differential equations, *Journal of Mathematical Analysis and Apolications, 102*, 420-434.

[10] Adomian, G. (1991). A Review of the decomposition method and some recent results for nonlinear equations, *Computers & Mathematics with Applications*, *21*, 101-127.

[11] Adomian, G. (1994). *Solving Frontier Problems of Physics: The decomposition Method*. Kulvar Academic, Boston.

[12] Mobasher, A., Oviedo, R. R., & Vu, B. (2001). *Application of Method of Decomposition to Classical Fluid Dynamics Problems*. Fifth Mississippi State Conference on Differential Equations & Computational Simulations. Starkville, MS.

[13] Wang, L. (2004). A new algorithm for solving classical Blasius equation, *Applied Mathematics and Computation*, *157*, 1-9.

[14] Babolian, E., Biazar, J., & Vahidi, A. R. (2004). A new computational method for Laplace transforms by decomposition method. *Applied Mathematics and Computation*, *150*, 841-846.

[15] Dehghan, M. (2004). Application of Adomian Decomposition Method for two dimensional parabolic equation subject to non-standard boundary specifications, *Applied Mathematics and Computation*, *157*, 549-560.

[16] Luo, X., Wu, Q. & Zhang, B. (2006). Revisit on Partial Solutions in the Adomian Decomposition Method: Solving Heat and Wave Equations, *Journal of Math. Analysis and Applications*, *321*, 353-363.

Biographies

AMIR A. MOBASHER is an Associate Professor of Mechanical Engineering at Alabama A&M University. Dr. Mobasher may be reached at amir.mobasher@aamu.edu

ARASTOO POOR BIAZAR is a senior research scientist of the Department of Atmospheric Sciences at University of Alabama in Huntsville. Dr. Biazar may be reached at biazar@nsstc.uah.edu

ZHENGTAO DENG is a professor of Mechanical Engineering at Alabama A&M University. Dr. Deng may be reached at zhengtao.deng@aamu.edu