

# Numerical Solution of Differential Equations by Direct Taylor Expansion

Piروز Mohazzabi, Jennifer L. Becker

Department of Mathematics and Physics, University of Wisconsin-Parkside, Kenosha, WI, USA

Email: mohazzab@uwp.edu

**How to cite this paper:** Mohazzabi, P. and Becker, J.L. (2017) Numerical Solution of Differential Equations by Direct Taylor Expansion. *Journal of Applied Mathematics and Physics*, 5, 623-630.

<https://doi.org/10.4236/jamp.2017.53053>

**Received:** January 30, 2017

**Accepted:** March 19, 2017

**Published:** March 22, 2017

Copyright © 2017 by authors and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## Abstract

A variation of the direct Taylor expansion algorithm is suggested and applied to several linear and nonlinear differential equations of interest in physics and engineering, and the results are compared with those obtained from other algorithms. It is shown that the suggested algorithm competes strongly with other existing algorithms, both in accuracy and ease of application, while demanding a shorter computation time.

## Keywords

Taylor Series, Expansion, Algorithm, Numerical Solution, Differential Equations

## 1. Introduction

With the advent of high speed personal computers and workstations and the decrease of the cost of computer resources in general, numerical methods and computer simulation have become an integral part of the scientific method and a third approach to the study of physical problems, in addition to theoretical and experimental methods.

The problem of solving differential equations numerically had long been of interest to mathematicians and scientists alike, long before the appearance of modern computers. One of the oldest and simplest algorithms is the Euler method, also known as the Euler-Cauchy method or the polygonal method [1]-[8]. This algorithm utilizes the definition of the derivative of a function,

$$\frac{dy}{dx} = \lim_{\Delta x \rightarrow 0} \frac{\Delta y}{\Delta x} \quad (1)$$

to solve first-order differential equations of the form [9]

$$\frac{dy}{dx} = f(x, y) \quad (2)$$

subject to initial condition  $y(x_0) = y_0$ . The method replaces the differential equation by a difference equation and, using a small enough step size  $\Delta x = h$ , advances the solution from  $x_n$  to  $x_{n+1} = x_n + h$  through

$$y_{n+1} = y_n + hf(x_n, y_n) \tag{3}$$

The Euler method can also be viewed as a Taylor expansion of the function about point  $x_n$  and retaining only the first two terms. Thus the remaining terms, or the error in the Euler algorithm, is given by

$$E = \frac{1}{2!} \left. \frac{d^2 y}{dx^2} \right|_{x=\xi} h^2 \tag{4}$$

where  $\xi$  is some value of  $x$  in the interval under consideration of width  $h$ . Consequently, the local error in the Euler method is of the order of  $h^2$ , resulting in a global error of the order of  $h$ . The Euler algorithm is, therefore, first order [10]. Achieving high accuracy with a first-order algorithm in some cases require very small intervals  $h$  (hence too many steps), increasing the total computation time and round-off error, resulting in instability of the algorithm.

An improved version of the Euler method is obtained by retaining three terms in the Taylor expansion of the function instead of two, yielding a second order algorithm [6]. Alternatively, one can advance the solution from  $x_n$  to  $x_{n+1}$  by using the value of the derivative at the midpoint of the interval rather than at the beginning,

$$y_{n+1} = y_n + hf \left[ x_n + \frac{h}{2}, y_n + \frac{h}{2} f(x_n, y_n) \right] \tag{5}$$

This algorithm is known as the modified Euler method, the midpoint method, or the second-order Runge-Kutta method [1] [2] [7]. It can be shown that this method is equivalent to the first three terms of the Taylor series described above.

Yet another way of improving the order of the Euler method is to use the average value of the derivative at the beginning and at the end of each interval,

$$y_{n+1} = y_n + h \frac{f(x_n, y_n) + f(x_{n+1}, y_{n+1})}{2} \tag{6}$$

where  $y_{n+1}$  on the right hand side is obtained from Equation (3). This algorithm is referred to as the Adams-Bashforth rule [6], and is second order.

The accuracy of the Euler method can be improved further by including higher terms of the Taylor expansion in the numerical calculations. Thus, by including the first five terms, one achieves a fourth-order algorithm. This approach, also referred to as the “Creeping up” process, has been mentioned in a limited number of references [7] [11], but has not received much attention. In particular, the discussion of this method in connection with the differential equations of order higher than one has been very limited in the literature [12]. This is mainly due to the general concern that in all but the simplest cases the necessary higher-order derivatives tend to become quite complicated, rendering the computations extraordinarily tedious [3] [8]. Thus, it is believed that the Taylor expansion method is generally not practical if derivatives of higher order

than the first are retained. It has also been stated that the method is not sufficiently accurate away from the initial point [6].

Currently the most widely used numerical algorithms for solving differential equations are the fourth-order Runge-Kutta (RK), fourth-order Adams-Bashforth-Moulton (ABM), and the fourth-order Milne methods. The RK algorithm [12] [13] is single-step, or self starting, and is the most commonly used method among the three. The ABM and the Milne algorithms are multi-steps and are not self starting [1] [14]. They are normally started by using a single-step algorithm, such as the RK algorithm.

The objective of this article is to discuss a variation of the direct Taylor series (DTS) algorithm for the solution of first- and higher-order differential equations. We show that not only this algorithm remains accurate away from the initial point, evaluation of the higher derivatives that are needed for accuracies comparable to the RK, ABM, and Milne methods are indeed quite simple. Finally, the accuracy and ease of application of the DTS method are explicitly demonstrated by considering several important second-order linear and nonlinear differential equations of mathematical physics and comparing their solutions using the fourth-order DTS, RK, ABM, and Milne methods.

## 2. A Variation of the Direct Taylor Series (DTS) Method

Consider a first-order differential equation given by (2). We expand the solution of this differential equation in a Taylor series about the initial point in each interval  $x_n$  to obtain its value at the end of that interval  $x_{n+1} = x_n + h$

$$y_{n+1} = y_n + \frac{y_n^{(1)}}{1!}h + \frac{y_n^{(2)}}{2!}h^2 + \frac{y_n^{(3)}}{3!}h^3 + \frac{y_n^{(4)}}{4!}h^4 + \dots \tag{7}$$

where  $y_n^{(1)}$ ,  $y_n^{(2)}$ ,  $y_n^{(3)}$ ,  $y_n^{(4)}$  are the first, second, third, and fourth derivatives of the function evaluated at  $x = x_n$ . Using the initial condition of the problem,  $y(x_0) = y_0$ , this expansion can be used iteratively to solve the differential equation up to some final value of the independent variable. The higher derivatives of the function, which are required in Equation (7), can be obtained by successive differentiation of the original differential Equation (2). Thus,

$$\left\{ \begin{array}{l} y^{(1)} = f(x, y) \\ y^{(2)} = \frac{\partial f}{\partial x} + \frac{\partial f}{\partial y}y^{(1)} \\ y^{(3)} = \frac{\partial^2 f}{\partial x^2} + 2\frac{\partial^2 f}{\partial x \partial y}y^{(1)} + \frac{\partial^2 f}{\partial y^2}(y^{(1)})^2 + \frac{\partial f}{\partial y}y^{(2)} \\ y^{(4)} = \frac{\partial^3 f}{\partial x^3} + 3\left(\frac{\partial^3 f}{\partial x^2 \partial y} + \frac{\partial^3 f}{\partial x \partial y^2}y^{(1)}\right)y^{(1)} + \frac{\partial^3 f}{\partial y^3}(y^{(1)})^3 \\ \quad + 3\left(\frac{\partial^2 f}{\partial x \partial y} + \frac{\partial^2 f}{\partial y^2}y^{(1)}\right)y^{(2)} + \frac{\partial f}{\partial y}y^{(3)} \\ \vdots \qquad \qquad \qquad \vdots \qquad \qquad \qquad \vdots \qquad \qquad \qquad \vdots \qquad \qquad \qquad \vdots \end{array} \right. \tag{8}$$

Although these equations look tedious, their evaluations in most cases are

quite straightforward and result in fairly simple expressions.

Using terms up to and including the  $k$ -th order in Equation (7) (*i.e.*, retaining  $k+1$  terms), results in a local error of

$$E = \frac{y^{(k+1)}(\xi)}{(k+1)!}h^{k+1} \tag{9}$$

and, thus, a global error of the order of  $h^k$ . Since the commonly used high-order numerical algorithms for solving differential equations are fourth order, we restrict our attention to Taylor expansions up to and including the fourth derivative, resulting in a fourth-order algorithm for comparison.

The DTS algorithm can be extended to numerically solve a differential equation of any order. To demonstrate this, consider a second-order differential equation given by

$$\frac{d^2y}{dx^2} = f(x, y, y^{(1)}) \tag{10}$$

subject to the initial conditions  $y(x_0) = y_0$  and  $y^{(1)}(x_0) = y_0^{(1)}$ . Of course, this equation can always be reduced to a system of two first-order differential equations. Alternatively, one can extend the Taylor algorithm as follows: From Equation (10) and its differentiation, the second and higher derivatives of the function are obtained and evaluated at  $x_0$ . Then Equation (7) is used to advance the solution from the initial point  $x_0$  to  $x_1$ . To advance the solution from  $x_1$  to  $x_2$ , however, various derivatives of the function at  $x_1$  are needed. These, in turn, can be obtained from the Taylor expansion of the derivatives themselves,

$$\left\{ \begin{array}{l} y_1^{(1)} = y_0^{(1)} + \frac{y_0^{(2)}}{1!}h + \frac{y_0^{(3)}}{2!}h^2 + \frac{y_0^{(4)}}{3!}h^3 + \dots \\ y_1^{(2)} = y_0^{(2)} + \frac{y_0^{(3)}}{1!}h + \frac{y_0^{(4)}}{2!}h^2 + \frac{y_0^{(5)}}{3!}h^3 + \dots \\ y_1^{(3)} = y_0^{(3)} + \frac{y_0^{(4)}}{1!}h + \frac{y_0^{(5)}}{2!}h^2 + \frac{y_0^{(6)}}{3!}h^3 + \dots \\ y_1^{(4)} = y_0^{(4)} + \frac{y_0^{(5)}}{1!}h + \frac{y_0^{(6)}}{2!}h^2 + \frac{y_0^{(7)}}{3!}h^3 + \dots \\ \vdots \qquad \qquad \qquad \vdots \qquad \qquad \qquad \vdots \end{array} \right. \tag{11}$$

and, using Equation (7), the solution is advanced from  $x_1$  to  $x_2$ . Iteration of these steps will eventually yield the value of the function at the desired final value of the variable. In Equation (11), the order of the highest derivative retained on the right side of each equation should be the same as the order of the numerical algorithm required. Thus, for a the numerical algorithm to be fourth-order, derivatives up to and including the fourth-order should be included in the expansions.

The suggested variation of the direct Taylor series, which we refer to as the DTS method, differs from the standard Taylor series method in the following way. In the standard method, the function is calculated at the required value of the variable  $x$  directly from the value of the function and its derivatives at the

initial value  $x_0$  [15],

$$y \cong y_0 + \frac{y_0^{(1)}}{1!}(x-x_0) + \frac{y_0^{(2)}}{2!}(x-x_0)^2 + \dots + \frac{y_0^{(n)}}{n!}(x-x_0)^n \tag{12}$$

In the suggested variation of the algorithm (the DTS method), on the other hand, the interval  $[x_0, x]$  is divided into many subintervals, each of width  $h$ . The function and its derivatives are then Taylor expanded and advanced from subinterval to subinterval until the function is evaluated at the required value of the variable  $x$ , thus resulting in a much greater accuracy.

### 3. Comparison with Other Algorithms

Applications of the DTS (direct Taylor series) method, as well as other common algorithms, to numerical solutions of first-order differential equations are straightforward and will not be discussed here. **Table 1** summarizes the results for several important second-order differential equations of mathematical physics. Each equation, subject to the given initial conditions, is numerically

**Table 1.** Several linear and nonlinear differential equations of interest. Each equation, subject to the initial conditions described, is numerically evaluated at  $x = x_f$ , using fourth-order Runge-Kutta (RK), Adams-Bashforth-Moulton (ABM), Milne, and the direct Taylor series (DTS) algorithms. In each cases, a step size of  $h = 0.1$  is used. The true values are given for comparison.

	Differential Equation	$x_f$	RK	ABM	Milne	DTS	True
(a)	$y'' + y' + y = \cos x$ $y(0) = 0, y'(0) = 1$	4	-0.8604	-0.8604	-0.8604	-0.8604	-0.8604
(b)	$y'' + \sin y = 0$ $y(0) = 1, y'(0) = 0$	3	-0.9488	-0.9488	-0.9488	-0.9487	-0.9488
(c)	$y'' + y^3 = 0$ $y(0) = 1, y'(0) = 0$	4	-0.9583	-0.9583	-0.9583	-0.9583	-0.9583
(d)	$y'' - (1 - y^2)y' + y = 0$ $y(0) = 1, y'(0) = 0$	3	-1.7280	-1.7279	-1.7280	-1.7280	-1.7280
(e)	$y'' + y' + y - \frac{y^3}{2} = \cos x$ $y(0) = 1, y'(0) = 0$	5	-0.7891	-0.7891	-0.7891	-0.7891	-0.7891
(f)	$y'' + y^{-7} - 2y^{-13} = 0$ $y(0) = 1.1, y'(0) = 0$	2	1.1485	1.1485	1.1485	1.1485	1.1485
(g)	$x^2 y'' + xy' + x^2 y = 0$ $y(1) = 1, y'(1) = 0$	2	0.6275	0.6275	0.6275	0.6275	0.6275
(h)	$(1 - x^2)y'' - 2xy' + 6y = 0$ $y(0) = 1, y'(0) = 0$	1	-2.0004	-2.0000	-2.0000	-2.0000	-2.0000
(i)	$xy'' + (1 - x)y' + 2y = 0$ $y(1) = 1, y'(1) = 0$	2	0.2338	0.2338	0.2338	0.2338	0.2338
(j)	$y'' - 2xy' + 4y = 0$ $y(0) = 1, y'(0) = 0$	1	-1.0000	-1.0000	-1.0000	-1.0000	-1.0000

solved and the function is evaluated at some final value of the variable  $x_f$  by the fourth-order RK (Runge-Kutta), ABM (Adams-Bashforth-Moulton), Milne, and the DTS algorithms, using a step size  $h = 0.1$  in each case.

In **Table 1**, Equations (a)-(f) describe different types of oscillating systems. In each equation  $x$  is time and  $y$  represent an oscillating physical quantity. The first equation corresponds to a sinusoidally driven damped harmonic oscillator. The second equation is the general equation of a pendulum (simple or compound), oscillating with some (not necessarily small) amplitude. The third is the differential equation of an intrinsically nonlinear oscillator, consisting of a mass tethered by two springs, oscillating in a direction perpendicular to the springs. Equation (d) describes the motion of a Van der Pol oscillator [16] [17]. Equation (e) is the Duffing's equation with damping, describing the oscillations of an externally driven hacksaw blade clamped at one end [18]. Finally, (f) is the differential equation of motion of a particle oscillating under a Lennard-Jones (6-12) potential energy function [19]. In each case, the final time  $x_f$  at which the function is evaluated numerically, is chosen such that the oscillator undergoes approximately one-half cycle. The last column gives the "true" value of the function at  $x_f$ , evaluated either by analytical solution of the differential equation when available, or by a very small step size ( $h = 0.001$ ), using the fourth-order RK algorithm.

Equations (g)-(j) are some of the common linear differential equations of mathematical physics, namely, Bessel, Legendre, Laguerre, and Hermite, respectively [20]. They show up in a variety of physical problems, ranging from heat conduction in solids, to wave equations, to quantum mechanical systems.

In all cases studied, the DTS algorithm generated a noticeably shorter CPU time than any other algorithm. In fact, the average CPU time for the DTS programs was found to be 21%, 58%, and 68% shorter than those for the corresponding RK, Milne and ABM programs, respectively.

#### 4. Discussion

Based on the results listed in **Table 1**, we see that in all cases studied the DTS (direct Taylor series) method provides accuracies that are essentially identical to the widely used algorithms of the same order while requiring a shorter computation time.

Evaluation of higher derivatives does not pose any serious difficulty. Indeed, higher derivatives become less problematic for differential equations of higher order. For example, with the fourth-order DTS algorithm and a second-order differential equation, only two higher derivatives should be computed from the differential equation; for a fourth-order equation, no higher-order derivatives are needed.

A striking feature of the DTS method is the ease with which its order of accuracy can be increased. For instance, to increase the order of the algorithm from four to six, one only needs to retain two additional terms in the Taylor expansion. Such an increase of the order of algorithm is not a trivial task in the

RK (Runge-Kutta), ABM (Adams-Bashforth-Moulton), or Milne method. Similarly, while generalizations of the latter methods to higher-order differential equations are not trivial, in the former case, it can be accomplished by simply incorporating Taylor expansions for higher derivatives, as we shall demonstrate in the following paragraph.

Third-order differential equations are not common in physics and engineering. Fourth-order equations, on the other hand, are occasionally encountered in some cases, such as bending of beams. As an example, we demonstrate the power of the DTS algorithm for the following simple fourth-order linear differential equation for which the analytical solution exists for comparison,

$$4 \frac{d^4 y}{dx^4} - 5 \frac{d^2 y}{dx^2} + y = 0 \quad (13)$$

subject to the initial conditions

$$y(0) = 1, \quad y^{(1)}(0) = \frac{1}{2}, \quad y^{(2)}(0) = 1, \quad y^{(3)}(0) = \frac{1}{8} \quad (14)$$

With a step size  $h = 0.1$ , the fourth-order DTS method yields  $y(1) = 2.0637$ . With the sixth-order algorithm and the same step size, we find  $y(1) = 2.0641757$ . These compare with the true value of 2.0641759, obtained from the analytical solution of the differential equation,

$$y = \cosh x + \sinh \frac{x}{2} \quad (15)$$

The discrepancies between the true and the numerical values in the two cases are, respectively, of the order  $h^4$  and  $h^6$ , as they should be.

## 5. Conclusion

In conclusion, we see that the direct Taylor series (DTS) algorithm is simple, easy to use, accurate, extendable to higher accuracies by simply retaining higher-order terms in Taylor expansions while requiring noticeably less computation time. Higher-order differential equations can be solved by trivial inclusion of Taylor expansions of higher derivatives. The other algorithms, such as Runge-Kutta, Adams-Bashforth-Moulton, and Milne methods are elegant but require complete construction of their working equations, which can be quite tedious depending on the order of the algorithm.

## Acknowledgements

This work was supported by a URAP grant from the University of Wisconsin-Parkside.

## References

- [1] Press, W.H., Teukolsky, S.A., Vetterling, W.T. and Flannery, B.P. (1992) Numerical Recipes in Fortran. 2nd Edition, Cambridge University Press, Cambridge, Chapter 16.
- [2] Brauer, F. and Nohel, J.A. (1967) Ordinary Differential Equations. Benjamin, New

York, Chapter 8.

- [3] Carnahan, B., Luther, H.A. and Wilkes, J.O. (1969) Applied Numerical Methods. Wiley, New York, Chapter 6.
- [4] Gould, H. and Tobochnik, J. (1996) An Introduction to Computer Simulation Methods. 2nd Edition, Addison-Wesley, Reading, 120-126.
- [5] Agnew, R.P. (1960) Differential Equations. 2nd Edition, McGraw-Hill, New York, Chapter 8.
- [6] Golomb, M. and Shanks, M. (1965) Elements of Ordinary Differential Equations. 2nd Edition, McGraw-Hill, New York, 18-33.
- [7] Tenenbaum, M. and Pollard, H. (1963) Ordinary Differential Equations. Dover, New York, Chapter 10.
- [8] Betz, H., Burcham, P.B. and Ewing, G.M. (1964) Differential Equations with Applications. 2nd Edition, Harper and Row, New York, 254-261.
- [9] Ordinary Differential Equations of Higher Order Can Always Be Reduced to a Set of First-Order Differential Equations (See, for Example, Reference [1]).
- [10] An Algorithm Is Called n-th Order if the Global Error Is of the Order of  $h^n$ .
- [11] Arfken, G. (1985) Mathematical Methods for Physicists. 3rd Edition, Academic Press, San Diego, 491-492.
- [12] Gear, C.W. (1971) Numerical Initial Value Problems in Ordinary Differential Equation. Prentice-Hall, Englewood Cliffs, Chapter 2.
- [13] Lapidus, L. and Seinfeld, J.H. (1971) Numerical Solution of Ordinary Differential Equations. Academic Press, New York, Chapter 2.
- [14] Boyce, W.E. and DiPrima, R.C. (1967) Elementary Differential Equations and Boundary Value Problems. Wiley, New York, 286-321.
- [15] Rainville, E.D. and Bedient, P.E. (1981) Elementary Differential Equations. 6th Edition, Macmillan Publishing Company, New York, 409-411.
- [16] Fowles, G.R. and Cassiday, G.L. (1999) Analytical Mechanics. 6th Edition, Saunders, New York, 132.
- [17] Baierlein, R. (1983) Newtonian Mechanics. McGraw-Hill, New York, 88-93.
- [18] Ibid, 81-82.
- [19] Kittel, C. (1976) Introduction to Solid State Physics. 5th Edition, Wiley, New York, 81.
- [20] Lebedev, N.N. (1972) Special Functions and Their Applications. Translated and Edited by Silverman R.A., Dover, New York.



**Submit or recommend next manuscript to SCIRP and we will provide best service for you:**

Accepting pre-submission inquiries through Email, Facebook, LinkedIn, Twitter, etc.

A wide selection of journals (inclusive of 9 subjects, more than 200 journals)

Providing 24-hour high-quality service

User-friendly online submission system

Fair and swift peer-review system

Efficient typesetting and proofreading procedure

Display of the result of downloads and visits, as well as the number of cited articles

Maximum dissemination of your research work

Submit your manuscript at: <http://papersubmission.scirp.org/>

Or contact [jamp@scirp.org](mailto:jamp@scirp.org)