Review paper on the applications of Runge-Kutta method to solve the Differential Equations

Mahananda B.Bhopale.

General Engineering Department,D.K.T.E's ,Textile and Engineering Institute ,Ichalkaranji.416115

Abstract

The main objective of this paper is to explain Runge Kutta method to solve the ODE numerically and its applications in different fields of engineering. Here we have reviewed the applications given by distinguished faculties in different papers.

Keywords: *Numerical solution to Differential equations; Runge Kutta method; absolute relative true error..*

Introduction:

Differential equations are among the most important mathematical tools used in producing models in the physical sciences, biological sciences, and engineering. The differential equations are commonly obtained as mathematical representations of many real world problems. Then the solution of the underlying problem lies in the solution of differential equation. Finding solution of the differential equation is then critical to that real world problem. The solution of differential equation with desired accuracy can be achieved using classical Taylor series method at a specified point. This means for given h, one can go on adding more and more terms of the series till the desired accuracy is achieved. This requires the expressions for several higher order derivatives and its evaluation. It poses practical difficulties in the application of Taylor series method: Higher order derivatives may not be easily obtained Even if the expressions for derivatives are obtained, lot of computational effort may still be required in their numerical evaluation It is possible to develop one step algorithms which require evaluation of first derivative as in Euler method but yields accuracy of higher order as in Taylor series. These methods require functional evaluations of f(t,y(t)) at more than one point. On the interval [tk,tk+1]. The Category of methods are known as Runge-Kutta methods of order 2, 3 and more depending upon the order of accuracy. There are several reasons that Euler's method is not recommended for practical use as neither the method is very accurate when compared to other, fancier, methods run at the equivalent step size nor is it very stable. For many scientific users, fourth-order Runge-Kutta is not just the first word on ODE integrators, but the last word as well.

Method:

The classical fourth-order Runge-Kutta formula, which has a certain sleekness of organization about it:

$k_1 = hf\,(x_n, y_n)$

$k_2 = hf\,(x_n + h/2, y_n + k_1/2)$

$k_3 = hf\,(x_n + h/2, y_n + k_2/2)$

$k_4 = hf\,(x_n + h, y_n + k_3)$

$y_{n+1} = y_n + k_1/6 + k_2/3 + k_3/3 + k_4/6 + O(h_5)$

The components in the algorithm above.

The Runge-Kutta method iterates the *x*-values by simply adding a fixed step-size of *h* at each iteration.

The y-iteration formula is far more interesting. It is a weighted average of four values—k_1, k_2, k_3, and k_4. Visualize distributing the factor of 1/6 from the front of the sum. Doing this we see that k_1 and k_4 are given a weight of 1/6 in the weighted average, whereas k_2 and k_3 are weighted 1/3, or twice as heavily as k_1 and k_4. (As usual with a weighted average, the sum of the weights 1/6, 1/3, 1/3 and 1/6 is 1.) So these k_i values are being used in the weighted average are -

$\boldsymbol{k_1}$ This quantity, $h\, f(x_n, y_n)$, is simply Euler's prediction for can be called Δy—the vertical jump from the current point to the next Euler-predicted point along the numerical solution.

$\boldsymbol{k_2}$ The x-value at which it is evaluating the function f. $x_n + h/2$ lies halfway across the prediction interval. $y_n + k_1/2$ is the current y-value plus half of the Euler-predicted Δy that we just discussed as being the meaning of k_1. Recalling that the function f gives us the *slope* of the solution curve, $f(x_n + h/2, y_n + k_1/2)$, gives us an estimate of the slope of the solution curve at this halfway point. Multiplying this slope by h, just as with the Euler Method before, produces a prediction of the y-jump made by the actual solution across the whole width of the interval, only this time the predicted jump is not based on the slope of the solution at the left end of the interval, but on the estimated slope halfway to the Euler-predicted next point.

$\boldsymbol{k_3}$ has a formula which is quite similar to that of k_2, except that where k_1 used to be, there is now a k_2. Essentially, the f-value here is yet another estimate of the slope of the solution at the "midpoint" of the prediction interval. This time, however, the y-value of the midpoint is not based on Euler's prediction, but on the y-jump predicted already with k_2. Once again, this slope-estimate is multiplied by h, gives yet another estimate of the y-jump made by the actual solution across the whole width of the interval.

$\boldsymbol{k_4}$ evaluates f at $x_n + h$, which is at the extreme right of the prediction interval. The y-value coupled with this, $y_n + k_3$, is an estimate of the y-value at the right end of the interval, based on the y-jump just predicted by k_3. The f-value thus found is once again multiplied by h, just as with the three previous k_i, giving a final estimate of the y-jump made by the actual solution across the whole width of the interval.

Thus each of the k_i gives an estimate of the size of the y-jump made by the actual solution across the whole width of the interval. The first one uses Euler's Method, the next two use estimates of the slope of the solution at the midpoint, and the last one uses an estimate of the slope at the right end-point. Each k_i uses the earlier k_i as a basis for its prediction of the y-jump.

This means that the Runge-Kutta formula for y_{n+1}, namely:

$$y_{n+1} = y_n + (1/6)(k_1 + 2k_2 + 2k_3 + k_4)$$

is simply the y-value of the current point plus a weighted average of four different y-jump estimates for the interval, with the estimates based on the slope at the midpoint being weighted twice as heavily as those using the slope at the end-points.

Applications :

1. In [1] application of Runge-Kutta method for the solution of non-linear partial differential equations are explained.

Runge-Kutta method is a powerful tool for the solution of ordinary differential equations (ODE). In this study the solution of a class of non-linear partial differential equations (PDE) is obtained by using this method. A similar approach has been taken by Rushton' and Marino and Yeh' in the analysis of aquifer systems. A particular problem of this type describing the transient flow of a gas through porous media is investigated. The equation representing this phenomenon is non-linear in nature. A summary of previous work on the solution of PDEs is given by Sinovec and Madsen". This paper illustrates a method of lines" and presents its application to seven different problems. The present paper uses a Runge-Kutta scheme, and in addition, provides a detailed comparison of the solutions with those obtained analytically as well as those from various finite-difference methods. The study shows that the solution of non-linear PDE is feasible by the Runge-Kutta method; it yields more accurate results than that obtained by finite difference methods for the example considered here. The use of Runge-Kutta methods to solve problems of this type is a novel approach. It is anticipated that this technique can be utilized to solve other complex problems of a similar nature.

The Runge-Kutta method treats every step in a sequence of steps in identical manner. Prior behavior of a solution is not used in its propagation. This is mathematically proper, since any point along the trajectory of an ordinary differential equation can serve as an initial point. The fact that all steps are treated identically also makes it easy to incorporate Runge-Kutta into relatively simple “driver” schemes. We consider adaptive step size control, discussed in the next section, an essential for serious computing. Occasionally, however, you just want to tabulate a function at equally spaced intervals, and without particularly high accuracy. In the most common case, you want to produce a graph of the function. Then all you need may be a simple driver program that goes from an initial xs to a final xf in a specified number of steps. To check accuracy, double the number of steps, repeats the integration, and compare results. This approach surely does not minimize computer time, and it can fail for problems whose nature requires a variable step size, but it may well minimize user effort. On small problems, this may be the paramount consideration.

2. In [2] application of Runge-Kutta method for finding multiple numerical solutions to intuitionistic fuzzy differential equations are discussed.

Many real world problems can be solved by converting them into ordinary differential equations. But in most of the cases, the exact solution is not possible and hence it is necessary to study its numerical solutions. In general most of the real world problems do not contain crisp or precise data. Therefore, to accommodate the impreciseness, the concept of fuzzy set was introduced. However, there are situations in which even the fuzzy sets are also not enough. For, the refinement of fuzzy set, namely, intuitionistic fuzzy set was introduced. In the literature, it is found that a very few investigations on the study of numerical solutions of ordinary differential equations which are intuitionistic fuzzy in nature. An ordinary differential equation with intuitionistic fuzzy number as its initial value, was studied and solved numerically using Runge-Kutta method by Abbasbandy and Allahviranloo[3]. A study has been made on a n th-order intuitionistic fuzzy linear differential equation which is time dependent, by Lata and Kumar [4]. In recent times, strong and weak solutions of first order homogeneous intuitionistic fuzzy differential equation have been discussed by Mondal, et al. [5] and the authors have studied an application of a system of differential equations with triangular intuitionistic numbers as its initial value [6]. Nirmala and Chenthur Pandian [7] have used Euler method for the discussion of the numerical solution of intuitionistic fuzzy differential equation (IFDE) International Conference on Applied and Computational Mathematics IOP Conf. Series: Journal of Physics: Conf. Series 1139 (2018) 012012 IOP Publishing doi:10.1088/1742-6596/1139/1/012012 2 by making use of α –cut representation of intuitionistic fuzzy set. Nirmala et al. [8, 9] have discussed numerical solution of IFDE by Modified Euler method and by fourth order Runge-Kutta method, respectively under the concept of generalised differentiability. Again, the generalised differentiability concept has been used by Parimala et al. [10, 11] for the discussion of numerical solutions of IFDE by Milne’s and Adam’s predictor-corrector methods, respectively. Wang and Guo [12] have studied multiple solutions of intuitionistic fuzzy differential equations based on(α, β) -level depiction of an intuitionistic fuzzy set. Nirmala et al. [13] have discussed multiple numerical solutions of IFDE based on (α, β) -level depiction of an intuitionistic fuzzy set by Euler method.

3. In [14] application of runge-kutta numerical methods to solve the schrodiger equation for hydrogen and positronium atoms are explained.

One of the most important eigenvalues equations in physics is Schrödinger wave equation, and for atomic mass m in the potential V is

$$\underbrace{\frac{-\hbar^2}{2m}\nabla^2\Psi(r)}_{\text{Kinetic Energy}} + \underbrace{V(r)\Psi(r)}_{\text{Potential Energy}} = \underbrace{E\Psi(r)}_{\text{Total Energy}}$$

$\hbar$ is the reduced Planck constant,

m is the electron mass,

∇ is the Laplacian operator,

Ψ is the wave function,

V is the potential energy,

E is the energy eigenvalue,

(r) denotes the quantities are functions of spherical polar coordinates (r, θ, φ).

This equation has the answer for the few analytical potential functions and for many analytical potential it cannot be solved. So in quantum mechanics, numerical solution of Schrödinger's wave equation is very important and so far, for the special cases has been solved numerically [15, 16].

CONCLUSION:

This paper proposes an orderly approach for the solution of non-linear partial differential equations, for finding numerical solutions of intuitionistic fuzzy Cauchy problems when they are expressed in (α, β)-cut representation, to solve the schrodiger equation for hydrogen and positronium atoms .A real time problem is taken and is written as an ordinary differential equation in an intuitionistic fuzzy problem. It is solved by the methods of Euler, Modified Euler and fourth order Runge-Kutta. In all the four cases the numerical solutions by Runge-Kutta gives good results. Further studies, in future can be done on the multiple numerical solutions of IFDEs by higher order methods.Also this method can be used in analysis of quantum systems with different potentials.

REFERENCES:

[1] Ashok Kumar. Application of Runge-Kutta method for the solution of non-linear partial differential equations (Received 23 March 1976)

[2] V Nirmala*, V Parimala and P Rajarajeswari. Application of Runge-Kutta method for finding multiple numerical solutions to intuitionistic fuzzy differential equations, 1139:012012 · December 2018

[3] Abbasbandy S and Allahviranloo T 2002 Numerical solution of fuzzy differential equation by runge- kutta method and the intutionistic treatment Notes on Intuitionistic Fuzzy Sets 8 3 pp 45– 53

[4]Lata S and Kumar A 2012 A new method to solve time-dependent intuitionistic fuzzy differential equations and its application to analyze the intuitionistic fuzzy reliability of industrial systems Concurrent Engineering 20 3 pp 177–184.

[5]Mondal S P, Sankar P and Roy T K 2014 First order homogeneous ordinary differential equation with initial value as triangular intuitionistic fuzzy number Journal of Uncertainty in Mathematics Science 2014 pp 1–17.

[6]Mondal S P and Roy T K 2015 System of differential equation with initial value as triangular intuitionistic fuzzy number and its application International Journal of Applied and Computational Mathematics 1 pp 1–26.

[7]Nirmala V and Chenthur Pandian S 2015 Numerical approach for solving intuitionistic fuzzy differential equation Applied Mathematical Sciences 9 367 pp 3337–3346.

[8]Nirmala V, Parimala V, Rajarajeswari P 2015 Modified Euler method for finding numerical solution of intuitionistic fuzzy differential equation under generalised differentiability concept International Journal of Applied Engineering Research 10 pp 40-45.

[9]Nirmala V, Parimala V and Rajarajeswari P 2015 Intuitionistic fuzzy differential equation and numerical solution by fourth order Runge-Kutta method under generalised differentiability International Journal of Applied Engineering Research 10 pp 46-51.

[10]Parimala V, Nirmala V and Rajarajeswari P 2017 Numerical Solution of Intuitionistic Fuzzy Differential Equation by Milne's Predictor-Corrector Method Under Generalised Differentiability International Journal of Mathematics and its Applications 5 1-A pp 45-54.

[11]Parimala V, Nirmala V and Rajarajeswari P 2017 Numerical Solution of Intuitionistic Fuzzy differential equation by Adams's Predictor-Corrector method under generalized differentiability Mathematical Sciences international research journal 6 2 pp 156-164.

[12]Wang L and Guo S C 2016 New Results on Multiple Solutions for Intuitionistic Fuzzy Differential Equations Journal of Systems Science and Information 4 6 pp 560–573.

[13]Nirmala V, Parimala V and Rajarajeswari P 2018 Multiple Numerical Solutions for Intuitionistic Fuzzy Differential Equations International Journal of Advanced Research Trends in Engineering and Technology 5 12 pp 477-483.

[14] A. Binesh,A.A. Mowlavi,& H. Arabshahi : 2010 . Application of runge-kutta numerical methods to solve the schrodiger equation for hydrogen and positronium, 5 5 pp315-319.

[15]. T. E. Simos, A fourth algebraic order exponentially-fitted Runge–Kutta method for the numerical solution of the Schrödinger equation. Journal of Numerical Analysis 2001 21(4):919-931.

[16]. J. C. Tremblay, T. J. Carrington. Using preconditioned adaptive step size Runge-Kutta methods for solving the Time-dependent Schrödinger equation. J Chem. Phys. 2004, 121(23):11535-41.

[17] Kaw A, Keteltas M. Lagrange interpolation; 2009. Available:http://numericalmethods.eng.usf.edu