Comparison New Algorithm Modified Euler in Ordinary Differential Equation Using Scilab Programming

N. M. M. Yusop, M. K. Hasan, and M. Rahmat

Abstract—Construction of algorithm is a significant aspect in solving a problem before being transferred to the programming language. Algorithm helped a user to do their job systematic and can be reduced working time. The real problem must be modeled into mathematical equation(s) before constructing the algorithm. Then, the algorithm be transferred into programming code using computer software. In this paper, a numerical method as a platform problem solving tool and researcher used Scilab 5.4.0 Programming to solve the mathematical model such as Ordinary Differential Equations (ODE). There are various methods that can be used in problem solving ODE. This research used to modify Euler's method because the method was simple and low computational. The main purposes of this research are to show the new algorithm for implementing the modify Euler's method and made comparisons between another modify Euler's and an exact value by integration solution. The comparison will be solved the ODE's using built-in functions available in Scilab programming.

Index Terms—Algorithm, modified euler, numerical method, Scilab.

I. INTRODUCTION

The development of numerical methods on a daily basis is to find the right solution techniques for solving problems in the field of applied science and pure science, such as semiconductor, wireless, weather forecasts, population, the spread of the disease, chemical reactions, physics, optics and others. Ordinary Differential Equations (ODE) acts reflect real-world problems in mathematical models [1]. Authors choose a numerical method to solve ODE problem. Euler method is an effective method in numerical methods are used in this study [2].

Euler's method is also called a tangent line method or one step method and is the simplest numerical method for solving Initial Value Problem (IVP) in ODE [3]. This method was developed by Leonhard Euler in 1768 and it is suitable for quick programming, simple implementation and low-cost computational [2]. However, the accuracy factor persuades scholar to use another complex method to replace Euler method [4], [5]. The primary aim of this investigation

is to discover a new algorithm as accurate as possible with the exact solution. We choose to solve the ODE's problem using modified Euler's method. We proposed the new algorithm using modify Euler's method that named as Harmonic Euler. Then the Harmonic Euler's be compared with exact solution and another modified Euler's method proposed by Chandio [6] and Qureshi [7].

Mathematical software and algorithm development is closely related to the problem represented by a mathematical model. Algorithm is a sequence of instructions to solve problems logically in simple language [8]. Algorithm also can illustrated as a step by step for solving the problems. Algorithm naturally is conceptual or abstract. Therefore, there are needs a way to delegate that can be communicated to humans or computers. Two popular way to convey the algorithm are pseudo code and flow chart. In this research, we choose pseudo code to transform the method and experiment.

Algorithms describe the elements involved clearly and then convert the algorithms into the program code more easily in a programming language. According to [9], construction process in mathematical software includes as follows,

- 1) The design and analysis algorithms
- 2) Algorithm coding
- 3) Details documentation
- 4) Distribution and maintenance of the software

Once the algorithm is developed, a computer program was implemented to test the effectiveness of the algorithm. The code program had been written using Scilab 5.4.0 Programming. At the final stage, all modified Euler's methods will compared with the exact solution.

II. BACKGROUND OF STUDY

A. Numerical Method as a Tool

The behavior of any physical or environment system can be described by one or more mathematical equation(s) [10]. If the mathematical equations are easy, the exact solution can be produced in closed-form. Even though closed-form solutions are desirable, for most engineering and applied science problems, the equations are relatively complex for which the exact solution cannot be found. In this situation, numerical method can be used to solve the mathematical equation using approximation solutions [11].

Together with the existing modern high speed digital computer technologies, the numerical methods have been effectively applied to study problems in mathematics, engineering and applied sciences. Numerical methods are great problem solving tools for handling equations, nonlinearities and complicated geometries that are often impractical to solve analytically.

Manuscript received April 29, 2014; revised July 28, 2014. This work was a part of first author Ph D research. This work was supported by Ministry of education under scholarship SLAB/SLAI.

N. M. M. Yusop and M. Rahmat are with Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia (UKM), 43600, UKM Bangi, Selangor, Malaysia, on leave from the National Universiti Defense, Malaysia (NDUM), Kem Sungai Besi, 57000, Kuala Lumpur, Malaysia (e-mail: n.moziyana@gmail.com, moziyana@upnm.edu.my, masura@ftsm.ukm.my).

M. K. Hasan was with Universiti Kebangsaan Malaysia. He is now with Faculty of Information Science and Technology, Malaysia (e-mail: mkh@ukm.edu.my).

B. Introduction of Scilab Programming

The intention of computer software is to provide a powerful computational tool. The writing of computer software requires a good understanding of mathematical model, numerical method and art of programming. A good computer software should provide some criteria of self-starting, accuracy and reliability, minimum numbers of levels, good documentation, ease of use and portability [11].

Scilab 5.4.0 Programming are selected as computer software in this experiment to improve the modified Euler's method. SCILAB is a tool for numerical, programming and powerful graphical environment. Scilab Programming is an amazingly useful, powerful and flexible for mathematics computer application using by engineers, researcher, scientist and students. It is developed for non profit by French government's world prominent "*Institut Nationale de Recherche en Informatique et en Automatique - INRIA* (National Institute for Informatics and Automation Research)". From that point, SCILAB can labeled as free software and no need to pay for licences [12]. The SCILAB console for the windows mode shows at Fig. 1.

Fig. 1. Overview of Scilab console.

Fig. 2. Sample overview Scilab text editor (SciNote).

Scilab is open source software work similar to numerical operation in the Matlab and other existing numerical or graphics environments. Scilab can use the execution of a wide range of operating system, such as UNIX, Windows, Linux, etc. [12]. Scilab programming can solve the problems related to the mathematical such as matrices, polynomial, linear equation, signal processing, differential equations and

statistic [3].

Fig. 2 shows the Scilab Text Editor called SciNotes. The SciNotes provided an editor to edit script easily. The editor allows managing several files at the same time.

C. Ordinary Differential Equation

To model real-world problems, especially for physics and engineering model, the exchange rate is a common problem in modeling. For example, the heat exchange rate against time and the environment as well as on the fluid flow rate. The problem will be translated into a mathematical model that would normally produce an equation [10].

For example, in the cooling process, Newton's laws insisted the temperature drop rate for a object of heat is proportional to the excess temperature than the surrounding temperature. A mathematical model that expresses the situation of cooling process is $\frac{d\theta}{dt} = -k(\theta - \theta_0)$ with θ , object temperature at the time t , and θ_0 temperatures around. This research focussed on solving IVP in ODE. IVP is the problem to find solutions of the equation of the n order ($n = 1, 2, 3, \dots$) that that fulfill n requirement [10]. The initial conditions must be at the same point. For example, $\frac{dy}{dx} + 3y = e^{-x}$, with initial condition, $y(0) = 1$.

D. Harmonic Euler as a Proposed Method

The authors examine the modified Euler method used by [6] and [7] in process to develop proposed method. The technique of improved the Euler Method called as modified Euler method. The modified Euler method tries to find a value of average slope of y between $x_n + h$ by averaging the slopes at x_n and at x_{n+1} [13]. Research of [6] uses the concept of the Heun method while the research of [7] using the concept of average. The concept average chosen by [7] is arithmetic mean which is called in this study as Arithmetic Euler.

The method proposed by the authors are used from Euler method same as in equation (1), that is

$$y_{n+1} = y_n + \Delta t f(t_0, y_0) \quad (1)$$

and modified by using concept of average. The proposed average is Harmonic mean of the two point function which is written as equation (2).

$$\frac{2[f(t_0, y_0) * f(t_1, y_1)]}{f(t_0, y_0) + f(t_1, y_1)} \quad (2)$$

Thus, the new coordinates of the point $R, y_0 + \Delta t/2$ and the slope are refer in equation (2), so the coordinate of R can written as an equation (3)

$$R = [t_0 + \frac{\Delta t}{2}, y_0 + \frac{\Delta t}{2} \frac{2[f(t_0, y_0) * f(t_1, y_1)]}{f(t_0, y_0) + f(t_1, y_1)}] \quad (3)$$

When equation (3) is included in the Euler method, this equation can be written as an equation (4) and the new equation is called the Harmonic Euler [14].

$$y_{n+1} = y_n + \Delta t f(t_0 + \frac{\Delta t}{2}, y_0 + \left(\frac{\Delta t}{2}\right) \left(\frac{2f(t_0, y_0) * f(t_1, y_1)}{f(t_0, y_0) + f(t_1, y_1)}\right)) \quad (4)$$

III. COMPARISON OF EULER METHOD

In this section, three algorithms of modified Euler are compared with the exact solution. The algorithms are proposed by [6], [7] and the authors. The purpose of this research to solve the ODE over the interval from $x = 0$ to 4 using a step size of 0.025, 0.1, 0.5 and 1 where $y(0) = 1$. Consider the equation $y' = 2y + 3e^t$ with exact solution which is $y = 3(e^{2t} - e^t)$. Time was recorded to compare each algorithm that gave close answer with exact solution. Equation (5) proposed by Chandio, equation(6) proposed by Qureshi and author proposed equation(7) [14]. The equations (5), (6) and (7) follow as below,

$$y_{n+1} = y_n + \Delta t f[t_n + 0.5\Delta t, y_n + (0.5\Delta t(f(t_n, y_n) + f(t_{n+1}, y_n + \Delta t f(t_n + \Delta t/2, y_n + \Delta t/2)))))] \quad (5)$$

$$y_{n+1} = y_n + \Delta t f[t_n + \frac{\Delta t}{2}, y_n + \frac{\Delta t}{2} (\frac{f(t_n, y_n) + f(t_{n+1}, y_{n+1})}{2})] \quad (6)$$

$$y_{n+1} = y_n + \Delta t f(t_0 + \frac{\Delta t}{2}, y_0 + (\frac{\Delta t}{2} (\frac{2(f(t_0, y_0) * f(t_1, y_1))}{f(t_0, y_0) + f(t_1, y_1)}))) \quad (7)$$

A. Chandio Algorithm

- 1) Start
- 2) Problem Equation $y' = 2y + 3e^x$
- 3) Set $x, y, h, y(n)$ and k value.
- 4) Start processing time.
- 5) Condition loop ($n \leq k$) for
 - Function $M \leftarrow (x_n + h/2, y_n + \Delta y/2)$
 - $b. y_{n+1} \leftarrow y_n + M$
- 6) End processing time.
- 7) Print processing time, y .
- 8) End

B. Arithmetic Euler Algorithm

- 1) Start
- 2) Problem Equation $y' = 2y + 3e^x$
- 3) Set $x, y, h, y(n)$ and k value.
- 4) Start processing time.
- 5) Condition loop ($n \leq k$) for
 - Set $A \leftarrow f(x_n, y_n) + f(x_{n+1}, y_{n+1})$
 - Set $B \leftarrow A/2$
 - Set $C \leftarrow f(x_n + h/2, y_n + (h/2 * B))$
 - $y_{n+1} \leftarrow y_n + h \times C$
- 6) End processing time.
- 7) Print processing time, y .
- 8) End.

C. Harmonic Euler Algorithm

- 1) Start
- 2) Problem Equation $y' = 2y + 3e^x$
- 3) Set $x, y, h, y(n)$ and k value.
- 4) Start processing time
- 5) Condition loop ($n \leq k$) for
 - Set $A \leftarrow f(x_{n+1}, y_{n+1})$

- Set $B \leftarrow f(x_n, y_n)$
- Set $C \leftarrow y_n + h/2 \times [2 * (A \times B) / A + B]$
- Set $D \leftarrow f(x_n + h/2, C)$
- $y_{n+1} \leftarrow y_n + h \times D$

- End for
- 6) End processing time.
- 7) Print processing time, y .
- 8) End

IV. MODIFIED EULER USES SCILAB PROGRAMMING

After the construction of the algorithms completed, we transferred into Scilab Programming to test which algorithm closed to the exact solution. The comparison of exact value and three modified Euler's shown in Table I. The error involved in this case is called relative error can be calculated as below [15], [16],

$$\text{Error} = \frac{|Ex - Ev|}{Ex}, \quad Ex = \text{Exact_value}, \quad Ev = \text{Euler's_modified_value}$$

TABLE I: COMPARISON VALUE OF MODIFIES EULER'S AND EXACT SOLUTION

Algorithm	$h = 0.025$	$h = 0.1$	$h = 0.5$	$h = 1$
Chandio	2.23E-01	1.15E+00	1.02E+01	1.35E+01
Aritmetic Euler	2.87E-02	1.54E-01	1.33E+00	2.20E+00
Harmonic Euler	2.84E-02	1.41E-01	5.37E-01	2.53E-01

Fig. 3. Overview graph generated using Scilab.

Table I, shown a comparison of the modified Euler's method proposed by Candio, Qureshi and authors. The Harmonic Euler algorithm suggested by the author approaches the exact solution compared with the algorithm proposed by others. When tested all algorithms in each step sizes, the result gave that Harmonic Euler method approaching an exact solution while in the largest step size.

According to Table II, time processing for each algorithm in second was recorded. The different step size which are 0.025, 0.1, 0.5 and 1 being used in each algorithm. The

purpose of comparing the different step size was to show that larger step size can reduce the complexity. For Harmonic Euler, when $h=0.025$, the time processing recorded 0.515 second while when $h=1$, the time processing is 0.489 second. When Chandio algorithm tested, using $h=0.025$ gave 0.468 second compared using $h=1$, gave 0.453 second prove that larger step size can reduce the complexity.

TABLE II: TIME PROCESSING EACH ALGORITHM

Algorithm	Time Processing (second)			
	$h = 0.025$	$h = 0.1$	$h=0.5$	$h=1$
Chandio	0.468	0.468	0.483	0.453
Aritmetic Euler	0.483	0.484	0.483	0.483
Harmonic Euler	0.515	0.499	0.499	0.489

Fig. 3 shown that overview graph for comparison modified Euler's with the exact solution. The large step size, $h=1$ were chosen to generate the graph. When $x=1$ and $x=2$, all modified Euler method has shown close to the exact solution. But when final stage, which is $x=4$, only Harmonic Euler approaches to the exact solution. Thus, we can conclude that Fig. 3 clearly shown that Harmonic Euler close to exact solution compare to another method.

V. CONCLUSION

The authors proposed a new algorithm using the modified Euler method that called Harmonic Euler as finding of this study. Subsequently, the Harmonic Euler was compared with the algorithm of Chandio and Arithmetic Euler. Each algorithm is tested by using Scilab 5.4.0 Programming and compared with the exact solution. Usually, the ordinary Euler method using a small step size gives the solution almost to the exact solution. However, the Harmonic Euler is also close to the exact solution while using a $h = 1$ as a step size. The benefits used larger step size will reduce complexity step and time processing. As a conclusion, the Harmonic Euler can be an alternative algorithm to the method proposed by Chandio and Qureshi.

REFERENCES

- [1] F. Yin, W. H. J. Song, and X. Cao, "Legendre wavelets-picard iteration method for solution of nonlinear initial value problems," *International Journal of Applied Physics and Mathematics*, vol. 3, pp. 127-131, Mar. 2013.
- [2] S. Fadugba, B. Ogurinde, and T. Okunlola, "Euler's method for solving initial value problems in ordinary differential equations," *The Pacific Journal of Science and Technology*, vol. 13, pp. 152-158, Nov. 2012.
- [3] Z. Salleh, "Ordinary Differential Equations(ODE) using Euler's technique and SCILAB programming," *Mathematical Models and Methods in Modern Science*, pp. 264-269, 2012.
- [4] P. Henrici, *Discrete Variable Methods in Ordinary Differential Equation*, John Wiley & Sons, 1962, pp. 187-190.

- [5] E. Kreyszig, *Advanced Engineering Mathematics*, 9th ed., John Wiley and Sons, 2006, pp. 19-20.
- [6] M. S. Chandio and A. G. Memon, "Improving the efficiency of Heun's metho," *Sindh University Resource Journal*, vol. 42, pp. 85-88, 2010.
- [7] S. Qureshi, Z. N. Memon, A. A. Shaikh, and M. S. Chandio, "On the construction and comparison of an explicit iterative algorithm withnonstandard finite difference schemes," *Mathematical Theory and Modeling*, vol. 3, pp. 78-87, 2013.
- [8] U. T. Malaysia, D. B. Pustaka, and K. P. Malaysia, *Ensiklopedia Teknologi MaklumatinEnsiklopedia Sains dan Teknologi*, Dewan Bahasa dan Pustaka, 2005, pp. 4-5.
- [9] A. R. Abdullah, *Pengiraan Berangka*, Dewan Bahasa dan Pustaka Kementerian Pendidikan Malaysia, pp. 26-34, 1990.
- [10] U. T. Malaysia, D. B. Pustaka, and K. P. Malaysia, *Ensiklopedia Matematikin Ensiklopedia Sains dan Teknologi*, Dewan Bahasa dan Pustaka, 2005, pp. 328-329, 346-348.
- [11] M. K. Jain, S. R. K. Iyengar, and R. K. Jain, *Numerical Methods for Scientific and Engineering Computation*, 3rd ed., Wiley Eastern Limited New Age International Limited, 1992, pp. 4-5.
- [12] G. E. Urroz, *Introduction to SCILAB*, 2007, p. 1-17.
- [13] C. F. Gerald and P. O. Wheatley, *Applied Numerical Analysis*, 7 ed., Pearson, 2004, pp. 400-402.
- [14] N. M. M. Yusop and M. K. Hasan, *Perbandingan Kaedah Euler dengan Tiga Kaedah Euler Terubahsuai pada Masalah Nilai Awal*, 2014, Faculty of Information Science and Technology, UKM, CAIT-IC-2014-1, pp. 1-13.
- [15] C. R. C. Teh, *Kaedah Berangka Untuk Sains Kejuruteraan menggunakan MAPLE*, 1st ed., Penerbit Universiti Teknologi Malaysia, 2005, pp. 208-214.
- [16] I. Mohd, *Penyelesaian Berangka Sistem Persamaan Pembezaan Biasa*, 1st ed., Dewan Bahasa dan Pustaka, 1993, pp. 42-48.

Nurhafizah Moziyana Mohd Yusop is a lecturer at the Faculty of Defence Science and Technology, National Defence University of Malaysia (NDUM). She graduated from Universiti Teknologi Malaysia for her B.Sc (computer science) and master in Information technology (industrial computing). Now she takes study leave as Ph.D candidate at Universiti Kebangsaan Malaysia (UKM). Her research studies are focused on numerical method and ordinary differential equation.

Mohammad Khatim Hasan is an associate professor at the Faculty of Information Science and Technology Universiti Kebangsaan Malaysia (UKM). He graduated from Universiti Kebangsaan Malaysia for his B.Sc (mathematics) and master in information technology (industrial computing) and Universiti Putra Malaysia (Ph. D). He had publish more than 200 in peer reviewed journals, proceedings, chapter in book and books.

Masura Rahmat is an IT teacher at the Faculty of Information Science and Technology Universiti Kebangsaan Malaysia. She graduated from Universiti Teknologi Malaysia in B.Sc in computer and master in information technology at the same University. Masura have more than 7 years experiance on teaching and learning especially in programming subject using variety of language such as JAVA , C and C++.