

Research Article

THE NUMERICAL SOLUTION OF ORDINARY DIFFERENTIAL EQUATIONS USING ADAM'S METHOD BY MATLAB

*Abdel Radi Abdel Rahman Abdel Gadir Abdel Rahman¹, SubhiAbdalazimAljily Osman²
Needal Hassan Elbadawy³, Arafat Abdelhameed Abdelrahman⁴

¹Department of Mathematics, Faculty of Education, Omdurman Islamic University, Omdurman, Sudan.

²Department of Mathematics , Faculty of Computer Science and Information Technology, University of ALBUTANA, Sudan.

³Department of Mathematics, College of Science, Tabuk University, Saudi Arabia.

⁴Department of Mathematics, Faculty of Sciences and Technology,Omdurman Islamic University, Omdurman, Sudan.

Received 29th December 2020; Accepted 30th January 2021; Published online 28th February 2021

ABSTRACT

MATLAB, which stands for Matrix Laboratory, is a software package developed by Math Works, Inc. to facilitate numerical computations as well as some symbolic manipulation .It strikes us as being slightly more difficult to begin working with it than such packages as Maple, Mathematical , and Macsyma, though once you get comfortable with it, it offers greater flexibility. The main point of using it is that it is currently the package you most likely found yourself working with if you get a job in engineering or industrial mathematics .So we found that the Matlab method in differential equations is very important and useful mathematical tools which help us to solve and plot differential equations .The aims of this paper is to solve differential equations numerically using Adams' method by MATLAB and compare between numerical and Matlab solution such that the Matlab solution is one of the most famous mathematical programs in solving mathematical problems. We followed the applied mathematical method using Matlab and we compared between the two solutions whence accuracy and speed .Also we explained that the solution of Matlab is more accuracy and speed than the numerical solution which proves the aptitude the usage of Matlab in different mathematical methods.

Keywords:Numerical Solution , Ordinary Differential Equations , Adam's Method , MATLAB

INTRODUCTION

The idea of extending the Euler method by allowing the approximate solution at a point to depend on the solution values and the derivative values at several previous step values was originally proposed by Bash forth and Adams . Not only was this special type of method, now known as the Adams–Bash forth method, introduced, but a further idea was suggested. This further idea was developed in detail by Moulton .[Butcher, 2008]. Adams methods compute a sequence $\{y_n\}$ where $y_n \approx y(t_n)$, and where the corresponding samples of the vector field are $y'_n = f(t_n, y_n)$. These are denoted y'_n instead of \dot{y}_n , since they are not derivatives of a function, but merely values of f obtained nearby the exact solution trajectory.[GustafSoderlind, 2017] On reaching t_n the basic methods are distinguished by whether or not they use previously computed quantities such as y_{n-1}, y_{n-2}, \dots . If they do, they are called methods with memory and otherwise, one–step methods.IVPs are categorized as non–stiff and stiff. It is hard to define stiffness, but its symptoms are easy to recognize. [Shampineet al., 2003]

Initial value problem

The general form of a first-order differential equation is

$$y' = f(x, y)$$

Where $y' = \frac{dy}{dx}$ and $f(x, y)$ is a given function. The solution of this equation contains an arbitrary constant (the constant of integration). To find this constant, we must know a point on the solution curve; that is, y must be specified at some value of x , say at $x = a$. We write

this auxiliary condition as

$$y(a) = \alpha$$

An ordinary differential equation of order n

$$y^{(n)} = f(x, y, y', \dots, y^{(n-1)})$$

can always be transformed in to n first-order equations. Using the notation

$$y_1 = y, \quad y'_1 = y_2, \quad y''_1 = y_3, \quad \dots, \quad y_n = y^{(n-1)}$$

the equivalent first- order equations are

$$y'_1 = y_2, \quad y'_2 = y_3, \quad y'_3 = y_4, \quad \dots, \quad y'_n = f(x, y, y_1, y_2, y_3, \dots, y_n)$$

The solution now requires the knowledge n auxiliary conditions. If these conditions are specified at the same value of x , the problem is said to be an initial value problem. Then the auxiliary conditions, called initial conditions, have the form

$$y_1(a) = \alpha_1, y_2(a) = \alpha_2, y_3(a) = \alpha_3, \dots, y_n(a) = \alpha_n$$

If y_i are specified at different values of x , the problem is called boundary value problem. [JaanKiusalaas, 2005]

Adams' Method

We once again seek the solution of the equation

$$y' = f(x, y) \quad (1)$$

On the interval $[x_0, b]$ which solution satisfies the initial condition $y = y_0$ when $x = x_0$ we introduce notation that will be needed later on . The approximate value of the solution at the points

$$x_0, x_1, x_2, \dots, x_n$$

Will be

$$y_0, y_1, y_2, \dots, y_n$$

The first differences or differences of the first order are

$$\Delta y_0 = y_1 - y_0, \quad \Delta y_1 = y_2 - y_1, \quad \dots, \quad \Delta y_n = y_n - y_{n-1}$$

The second differences or differences of the second order are

$$\Delta^2 y_0 = \Delta y_1 - \Delta y_0 = y_2 - 2y_1 + y_0$$

*Corresponding Author: Abdel Radi Abdel Rahman Abdel Gadir Abdel Rahman,
1Department of Mathematics, Faculty of Education, Omdurman Islamic University,
Omdurman, Sudan.

$$\Delta^2 y_1 = \Delta y_2 - \Delta y_1 = y_3 - 2y_2 + y_1$$

$$\Delta^2 y_{n-2} = \Delta y_{n-1} - \Delta y_{n-2} = y_n - 2y_{n-1} + y_{n-2}$$

Differences of the second differences are called differences of the third order, and so forth, we denoted by $y_0, y_1, y_2, \dots, y_n$ the approximate value of the derivative, and by $y_0'', y_1'', y_2'', \dots, y_n''$ the approximate value of the second derivative, etc. similarly we determine the first differences of the derivative :

$$\Delta y_0' = y_1' - y_0', \Delta y_1' = y_2' - y_1', \dots, \Delta y_n' = y_n' - y_{n-1}'$$

The second difference of the derivative:

$$\Delta^2 y_0' = \Delta y_1' - \Delta y_0', \Delta^2 y_1' = \Delta y_2' - \Delta y_1', \dots, \Delta^2 y_{n-2}' = \Delta y_{n-1}' - \Delta y_{n-2}'$$

And so on.

Write Taylor's formula for solving an equation in right neighbourhood of the point $x = x_0$:

$$y = y_0 + \frac{x - x_0}{1} y_0' + \frac{(x - x_0)^2}{1.2} y_0'' + \dots + \frac{(x - x_0)^m}{1.2 \dots m} y_0^{(m)} + R_m \tag{2}$$

In this formula y_0 is known and value of y_0', y_0'', \dots of the derivatives are found from equation (1) as follows. Putting the initial values x_0 and y_0 into the right side of equation (1). We find y_0' :

$$y_0' = f(x_0, y_0)$$

Differentiating the term of (1) with the respect to x , we get

$$y'' = \frac{\partial f}{\partial x} + \frac{\partial f}{\partial y} y' \tag{3}$$

Substituting into the right side the value x_0, y_0, y_0' we find

$$y_0'' = \left(\frac{\partial f}{\partial x} + \frac{\partial f}{\partial y} y' \right)_{x=x_0, y=y_0, y'=y_0'}$$

Once more differentiating (3) with respect to x and substituting the value x_0, y_0, y_0', y_0'' we find y_0''' continuing in this fashion. we can find the value of this derivatives of any order $x = x_0$. All terms are known except the remainder R_m on the right side of (2) thus neglecting the remainder we can obtain an Approximation of the solution for any value of x : their accuracy will depend upon the quantity $|x - x_0|$ and the number of the terms in the expansion. In the method given below we determine by formula (2) only the first few value of y when $|x - x_0|$ is small we determine the values y_1 and y_2 for $x_1 = x_0 + h$ and $x_2 = x_1 + h$ taking four terms of the expansion (y_0 is known from the initial data):

$$y_1 = y_0 + \frac{h}{1} y_0' + \frac{h^2}{1.2} y_0'' + \frac{h^3}{3!} y_0''' \tag{4}$$

$$y_2 = y_0 + \frac{2h}{1} y_0' + \frac{(2h)^2}{1.2} y_0'' + \frac{(2h)^3}{3!} y_0''' \tag{4}$$

We find

$$y_0' = f(x_0, y_0), \quad y_1' = f(x_1, y_1), \quad y_2' = f(x_2, y_2)$$

Knowing y_0, y_1, y_2 it is possible to determine $\Delta y_0', \Delta y_1', \Delta^2 y_0'$.

Let us determine the value of y_{k+1} from Taylor's formula, setting $a = x_k, x = x_{k+1} = x_k + h$:

$$y_{k+1} = y_k + \frac{h}{1} y_k' + \frac{h^2}{1.2} y_k'' + \frac{h^3}{1.2.3} y_k''' + \dots + \frac{(h)^m}{1.2 \dots m} y_k^{(m)} + R_m.$$

[N.PISKUNOV, 1969]

Theorem : The Adams method is convergent with the order, equal to the order of the consistency.[IstvánFaragó, 2013]

In this method, the step from y_n to y_{n+1} is performed by a formula expressed in terms of differences of $f(x, y)$. Adams' method uses the formula

$$y_{n+1} = y_n + h \left[f_n + \frac{1}{2} \Delta f_n + \frac{5}{12} \Delta^2 f_n + \frac{3}{8} \Delta^3 f_n + \dots \right] \tag{1}$$

$$h = x_{n+1} - x_n$$

$$f_n = f(x_n, y_n)$$

$$\Delta f_n = f_n - f_{n-1}$$

$$\Delta^2 f_n = \Delta f_n - \Delta f_{n-1}$$

and so on.

Obviously, to form a table of differences, it is necessary to have several (4 or more) approximate values of $y(x)$ in addition to the given initial condition $y(0)$. These values can be found by other methods such as the Taylor series or Runge-Kutta methods.

Example:

Given the differential equation

$$y' = 2y + x$$

with the initial condition $y(0) = 1$

compute the approximate values of y for $x = 0.1, 0.2, 0.3, 0.4,$ and 0.5 by the third-order Runge-Kutta method. Then, find the value of y corresponding to $x = 0.6$ correct to three decimal places using Adams' method.

Solution:

we computations of $y_1, y_2, y_3, y_4,$ and y_5 using the third-order Runge-Kutta method.

$$h = x_1 - x_0 = 0.1 - 0 = 0.1$$

$$L(1) = h * f(x_n, y_n) = h * f(x_0, y_0) = 0.1 * (0 + 2(1)) = 0.2$$

$$L(2) = h * f(x_n + 0.5 * h, y_n + 0.5 * L(1)) = h * f(x_0 + 0.5 * h, y_0 + 0.5 * L(1)) = h * [(0 + h) + 2 * (1 + 0.5 * L(1))] = 0.2250$$

$$L(3) = h * f(x_n + h, y_n + 2 * L(2) - L(1)) = h * [(0 + h) + 2 * (1 + 2 * L(2) - L(1))] = h * [(0 + h) + 2 * (1 + 2 * 0.2250 - 0.2)] = 0.2600$$

$$y_1 = y_0 + (L(1) + 4 * L(2) + L(3)) / 6 = 1 + (0.2 + 4 * 0.2250 + 0.2600) / 6 = 1.2267$$

$$h = x_2 - x_1 = 0.2 - 0.1 = 0.1$$

$$L(1) = h * f(x_n, y_n) = h * f(x_1, y_1) = 0.1 * (0.1 + 2(1.2267)) = 0.2553$$

$$L(2) = h * f(x_n + 0.5 * h, y_n + 0.5 * L(1)) = h * [(0 + 0.5 * h) + 2 * (1 + 0.5 * L(1))] = 0.2859$$

$$L(3) = h * f(x_n + h, y_n + 2 * L(2) - L(1)) = h * [(0 + h) + 2 * (1 + 2 * L(2) - L(1))] = 0.3286$$

$$y_2 = y_1 + (L(1) + 4 * L(2) + L(3)) / 6 = 1.2267 + (0.2553 + 4 * 0.2859 + 0.3286) / 6 = 1.5146$$

$$h = x_3 - x_2 = 0.3 - 0.2 = 0.1$$

$$L(1) = h * f(x_n, y_n) = h * f(x_2, y_2) = 0.1 * (0.2 + 2(1.5146)) = 0.3229$$

$$L(2) = h * f(x_n + 0.5 * h, y_n + 0.5 * L(1)) = h * [(0 + 0.5 * h) + 2 * (1 + 0.5 * L(1))] = 0.3602$$

$$L(3) = h * f(x_n + h, y_n + 2 * L(2) - L(1))$$

$$= h * [(0 + h) + 2 * (1 + 2 * L(2) - L(1))]$$

$$L(3) = 0.4124$$

$$y_3 = y_2 + (L(1) + 4 * L(2) + L(3))/6$$

$$y_3 = 1.5146 + (0.3229 + 4 * 0.3602 + 0.4124)/6$$

$$y_3 = 1.8773$$

$$h = x_4 - x_3$$

$$= 0.4 - 0.3 = 0.1$$

$$L(1) = h * f(x_n, y_n)$$

$$= h * f(x_3, y_3)$$

$$= 0.1 * (0.3 + 2(1.8773))$$

$$L(1) = 0.4055$$

$$L(2) = h * f(x_n + 0.5 * h, y_n + 0.5 * L(1))$$

$$= h * [(0 + 0.5 * h) + 2 * (1 + 0.5 * L(1))]$$

$$L(2) = 0.4510$$

$$L(3) = h * f(x_n + h, y_n + 2 * L(2) - L(1))$$

$$= h * [(0 + h) + 2 * (1 + 2 * L(2) - L(1))]$$

$$L(3) = 0.5148$$

$$y_4 = y_3 + (L(1) + 4 * L(2) + L(3))/6$$

$$y_4 = 1.8773 + (0.4055 + 4 * 0.4510 + 0.5148)/6$$

$$y_4 = 2.3313$$

$$h = x_5 - x_4$$

$$= 0.5 - 0.4 = 0.1$$

$$L(1) = h * f(x_n, y_n)$$

$$= h * f(x_4, y_4)$$

$$= 0.1 * (0.4 + 2(2.3313))$$

$$L(1) = 0.5063$$

$$L(2) = h * f(x_n + 0.5 * h, y_n + 0.5 * L(1))$$

$$= h * [(0 + 0.5 * h) + 2 * (1 + 0.5 * L(1))]$$

$$L(2) = 0.5619$$

$$L(3) = h * f(x_n + h, y_n + 2 * L(2) - L(1))$$

$$= h * [(0 + h) + 2 * (1 + 2 * L(2) - L(1))]$$

$$L(3) = 0.6398$$

$$y_5 = y_4 + (L(1) + 4 * L(2) + L(3))/6$$

$$y_5 = 2.3313 + (0.5063 + 4 * 0.5619 + 0.6398)/6$$

$$y_5 = 2.8969. [7]$$

Next we compute the following values to be used in Adams' formula of (1). These are shown below.

x_n	y_n	$f_n = x_n + 2y_n$	Δf_n	$\Delta^2 f_n$	$\Delta^3 f_n$
0.0	1.0000	2.0000			
			0.5534		
0.1	1.2267	2.5534		0.1224	
			0.6758	0.1496	0.0272
0.2	1.5146	3.2292		0.8254	0.0330
			1.0080	0.1826	0.0406
0.3	1.8773	4.0546		1.2312	
			1.0080	0.2232	
0.4	2.3313	5.0626			
			1.2312		
0.5	2.8969	6.2938			

$$\square_6 = 2.8969 \cdot 0.1 \left[6.2638 + \frac{1}{2}(1.2312) + \frac{5}{12}(0.2232) + \frac{3}{8}(0.0406) \right] = 3.599$$

Mat lab Solution:

Code:

```
m=inputdlg('Enter the 1st order ODE or slope function:');
s=m{:};
d=str2func(['@(x,y) s']);
f=str2func(['@(x,y) vectorize(s)']);
x=input('Select the initial x value:');
y=input('Select the initial y value:');
h=input('Select the increment value:');
order=0;
order2=0;
n=0;
term=0;
while order==0
order=input('\nSelect what order of Runge-Krutta to use (1 is min.
and 5 is max.):');
if order<1||order>5
fprintf('Wrong input! The value must be between 1 and 5 only.\n');
order=0;
end
end
if order==2
while order2==0
order2=input('Choose "1" for Heun method, "2" for Midpoint
method, and "3" for Ralston method:');
if order2<1||order2>3
fprintf('Wrong input! The value must be between 1 and 3
only.\n\n');
order2=0;
end
end
end
while (term<1)||(term>2)
term=input('\nPlease choose what termination criteria you
desire.\nPress "1" for a specified x value and "2" for nth
iteration:');
if term==1
val=input('The specified x value is:');
itr=(val-x)/h;
break;
end
end
if term==2
```

```

itr=input('The desired nth iteration (iterations will start from n=0)
is:');
break;
end
fprintf('Wrong input! Please try again.\n\n');
end
fprintf('\n');
%This is for labels
q='itrn';
w='x value';
b='current y';
t='solved y';
o='slope value';
s='f_n';
k='';

```

```

a=[q,k,w,k,b,k,t,k,o,s];
disp(a);

```

```

for i=1:itr+1

```

```

Y(i)=y;
X(i)=x;
l=x;
p=y;
f_n(i)=x+2*y;
x=x+(0.50*h);
k1=f(l,p);
y=p+(0.50*k1*h);

k2=f(x,y);
y=p-(k1*h)+(2*k2*h);
x=+h;
k3=f(x,y);
fr=(k1/6)+(4*k2/6)+(k3/6);
y=p+(h*fr);
C(i)=l;
V(i)=p;
fprintf('%2.0f %13.4f %13.4f %13.4f %17.4f %17.4f \n
',n,l,p,y,fr,f_n(i));
n=n+1;

```

```

end

```

```

for k=itr:-1:1
f_new(k)=f_n(k+1)-f_n(k);

```

```

end

```

```

for k=itr-1:-1:1
f_new1(k)=f_new(k+1)-f_new(k);

```

```

end

```

```

for k=itr-2:-1:1
f_new2(k)=f_new1(k+1)-f_new1(k);

```

```

end

```

```

Y=p+h*(f_n(itr+1)+0.5*f_new(itr)+(5/12)*f_new1(itr-1)+(3/8)*f_new2(itr-2));
plot(C,V, '+b-')
title('Numeric Solution of Differential Equation '),...
xlabel('x'), ylabel('y'), grid

```

RESULTS

```

Command Window
>> Adam_EE
Select the initial x value:0
Select the initial y value:1
Select the increment value:0.1

Select what order of Runge-Kutta to use (1 is min. and 5 is max.):3

Please choose what termination criteria you desire.
Press "1" for a specified x value and "2" for nth iterations!
The specified x value is:0.5

itrn  x value  current y  solved y  slope valuef_n  2.0000
0     0.0000  1.0000    1.2267    2.2667          2.0000
1     0.1000  1.2267    1.5146    2.8790          2.5533
2     0.2000  1.5146    1.8773    3.6269          3.2291
3     0.3000  1.8773    2.3313    4.5403          4.0545
4     0.4000  2.3313    2.8969    5.6559          5.0626
5     0.5000  2.8969    3.5987    7.0184          6.2938
A >> |

```

```

>> f_n'

ans =

    2.0000
    2.5533
    3.2291
    4.0545
    5.0626
    6.2938

>> f_new'

ans =

    0.5533
    0.6758
    0.8254
    1.0081
    1.2312

```

```

>> f_new1'

ans =

    0.1225
    0.1496
    0.1827
    0.2231

>> f_new2'

ans =

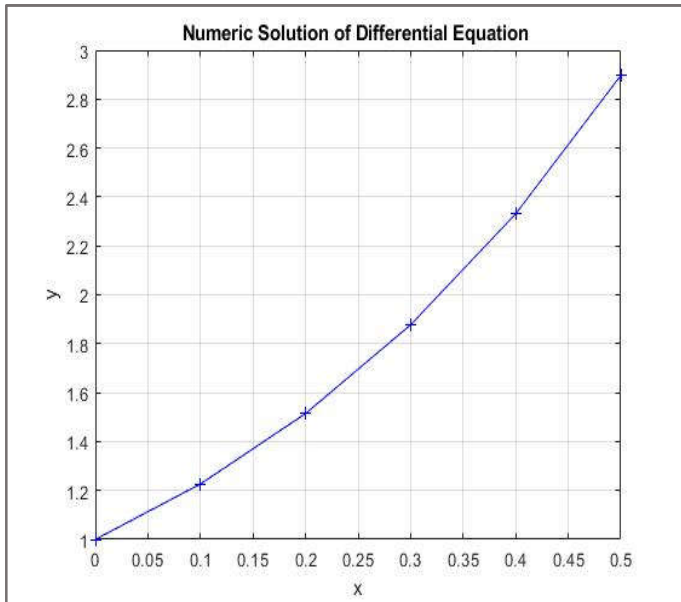
    0.0271
    0.0331
    0.0404

>> Y

Y =

    3.5986

```



REFERENCES

1. GustafSoderlind, Numerical Methods for Differential Equations, An Introduction to Scientific Computing, November 29, 2017, page 105
2. Istv'anFarag'o, Numerical Methods for Ordinary Differential Equations, Budapest, June 2013.
3. JaanKiusalaas, NUMERICAL METHODS IN ENGINEERING WITH MATLAB, JaanKiusalaas, 2005 page 251-252.
4. J. C. Butcher, Numerical Methods for Ordinary Differential Equations, Second Edition, ISBN: 978-0-470-72335-7, 2008, page 104
5. L.F. Shampine and I. Gladwell, S. Thompson, Solving ODEs with Matlab, January 2003, page 1
6. N.PISKUNOV , G. YANKOVSKY, H. C. HNCKYHOB, DIFFERENTIAL AND INTRGRAL, CALCULUS, MIR PUBERS, MOVEOS, 1969, page 584-586
7. Steven T.Karris, Numerical Analysis, Using MATLAB® and Spreadsheets, Second Edition, page 9. 13.
