

# Numerical methods for the solution of ordinary differential equations: initial value problems

Endorsed by  
The Royal Aeronautical Society

## ESDU DATA ITEMS

Data Items provide validated information for use in engineering design and analysis by, or under the supervision of, professionally qualified engineers. The data are founded on an evaluation of all the relevant information, both published and unpublished, and are invariably supported by original work of ESDU staff engineers or consultants. The whole process is subject to independent review for which crucial support is provided by industrial companies, government research laboratories, universities and others from around the world through the participation of some of their leading experts on ESDU Technical Committees. This process ensures that the results of much valuable work (theoretical, experimental and operational), which may not be widely available or in a readily usable form, can be communicated concisely and accurately to the engineering community.

We are constantly striving to develop new work and review data already issued. Any comments arising out of your use of our data, or any suggestions for new topics or information that might lead to improvements, will help us to provide a better service.

## THE PREPARATION OF THIS DATA ITEM

The work of the permanent professional staff of Engineering Sciences Data Unit on this particular Data Item was monitored and guided by the Dynamics Committee which has the following constitution:

### Chairman

Dr H.H.B.M. Thomas — Independent

### Members

|                          |                                               |
|--------------------------|-----------------------------------------------|
| Prof. G.T.S. Done        | — The City University, London                 |
| Prof. G.J. Hancock       | — Queen Mary College, London University       |
| Mr M.R. Heath            | — British Aerospace Dynamics Group, Stevenage |
| Dr M. Tobak <sup>†</sup> | — NASA, Ames Research Center, Calif., USA     |
| Dr M.A. Woodhead         | — University of Salford.                      |

<sup>†</sup> Corresponding Member.

The work on this Item was carried out in the Mechanical Motion and System Dynamics Group of the Engineering Sciences Data Unit. The member of staff who undertook the technical work involved in the initial assessment of the available information and the construction and subsequent development of the Item was

Mr C.J. Loughton — Group Head

The Dynamics Committee, which first met in January 1962, has the co-operation of many engineers and scientists in industry, research establishments and universities from whom much assistance and unpublished information is being received.

# NUMERICAL METHODS FOR THE SOLUTION OF ORDINARY DIFFERENTIAL EQUATIONS: INITIAL VALUE PROBLEMS

## CONTENTS

|                                                                                              | Page |
|----------------------------------------------------------------------------------------------|------|
| 1. INTRODUCTION                                                                              | 1    |
| 2. NOTATION                                                                                  | 2    |
| 3. MATHEMATICAL DESCRIPTION                                                                  | 4    |
| 3.1 First-order Methods                                                                      | 4    |
| 3.2 Second-order Methods                                                                     | 5    |
| 4. RUNGE-KUTTA METHODS                                                                       | 6    |
| 4.1 Classical Fourth-Order Runge-Kutta                                                       | 7    |
| 4.2 Runge-Kutta-Merson                                                                       | 8    |
| 4.3 Runge-Kutta-Scraton                                                                      | 9    |
| 4.4 Runge-Kutta-Hull                                                                         | 9    |
| 4.5 Runge-Kutta-England                                                                      | 10   |
| 4.6 Runge-Kutta-Gill                                                                         | 11   |
| 4.7 General Remarks                                                                          | 11   |
| 4.8 Adaptive Control of Step Size                                                            | 12   |
| 5. PREDICTOR-CORRECTOR METHODS                                                               | 14   |
| 5.1 Adam's Formulae                                                                          | 17   |
| 5.2 Milne's Formulae                                                                         | 18   |
| 5.3 Hamming's Method                                                                         | 19   |
| 5.4 General Remarks                                                                          | 20   |
| 6. GENERAL CONSIDERATIONS                                                                    | 21   |
| 7. SECOND-ORDER EQUATION METHODS                                                             | 22   |
| 7.1 Wilson- $\theta$ Linear Acceleration Method                                              | 22   |
| 7.2 Newmark Generalised Acceleration Method                                                  | 25   |
| 8. REFERENCES                                                                                | 28   |
| 9. TABLES                                                                                    | 30   |
| Table 9.1 Runge-Kutta formulae                                                               | 30   |
| Table 9.2 Adams predictor-corrector formulae                                                 | 33   |
| Table 9.3 Milne predictor-corrector formulae                                                 | 34   |
| APPENDIX A REDUCTION OF A HIGH-ORDER DIFFERENTIAL EQUATION TO A SET OF FIRST-ORDER EQUATIONS | 35   |
| A1. Derivation                                                                               | 35   |

## NUMERICAL METHODS FOR THE SOLUTION OF ORDINARY DIFFERENTIAL EQUATIONS: INITIAL VALUE PROBLEMS

### 1. INTRODUCTION

This Item considers methods for the numerical evaluation of initial value problems in ordinary differential equations. Some of the methods considered are applicable to a set of  $n$  first-order differential equations with an appropriate corresponding set of  $n$  initial conditions specified at a single value of the independent variable. It is also shown that a differential equation of higher-order than one, provided it is possible to isolate the highest derivative of the dependent variable, can be treated using the same methods. Other methods apply to a set of  $m$  second-order equations with an appropriate corresponding set of  $2 \times m$  initial conditions also specified at a single value of the independent variable.

Not included in the Item are those methods that are appropriate for boundary value problems, partial differential equations and methods that deal directly with differential equations that are higher than first-order.

Two broad classes of method are considered in this Item; single-step methods, specifically those that are referred to as Runge-Kutta and acceleration methods, and those multi-step methods that fall within the term predictor-corrector. Only those formulae that have found wide applicability are dealt with specifically in this Item but reference is made to other methods available in the general literature.

For the Runge-Kutta class of formulae fourth-order methods are described in detail since they are in common use in engineering applications. Also, since they are single-step formulae, accuracy improvement can be achieved by a simple change of step size. The schemes presented provide alternative ways of achieving reductions in either truncation error estimation or computation time. Adaptive control of the step size is also discussed and in this context a fifth-order Runge-Kutta method is presented.

A fixed-order predictor-corrector scheme, say fourth-order, can be used for differential equation solving with subsequent savings in the number of functional evaluations per step over the equivalent Runge-Kutta methods, but with an increased complexity of step changes. However the full advantages of these methods are obtained using a variable-order variable-step method and so general equations for evaluating the predictor-corrector formulae for any order are provided.

Acceleration methods which directly treat second-order differential equations have found increasing use in engineering problems. Two single-step methods, the Wilson- $\theta$  Linear Acceleration Method and the Newmark Generalised Acceleration Method are described in this Item.

The compromise between accuracy of results and effort expended is of considerable importance and so is discussed in this Item together with some general remarks concerning stability and "stiff" equations.

## 2. NOTATION

Any coherent system of units may be used.

|                 |                                                                                   |
|-----------------|-----------------------------------------------------------------------------------|
| $A$             | integral of interpolated polynomial (see Equation (5.4))                          |
| $A, B, C$       | coefficients associated with Runge-Kutta-Merson Method (see Equation (4.14))      |
| $a, b$          | general coefficients                                                              |
| $C(x)$          | corrector interpolating polynomial                                                |
| $D_n$           | associated with error truncation term (see Equation (4.7))                        |
| $E$             | truncation error term                                                             |
| $\underline{F}$ | matrix associated with second-order methods (see Equations (7.12) and (7.16))     |
| $f$             | general arbitrary function                                                        |
| $\underline{G}$ | matrix associated with second-order methods (see Equations (7.13) and (7.17))     |
| $h$             | step length                                                                       |
| $i, j$          | general integers                                                                  |
| $k$             | general coefficient                                                               |
| $M, m, N, n$    | general integers                                                                  |
| $O\{\dots\}$    | order of $\{\dots\}$                                                              |
| $P(x)$          | predictor interpolating polynomial                                                |
| $q$             | general parameter used in Runge-Kutta-Scruton and Runge-Kutta-Gill algorithms     |
| $r, s$          | general parameters used in Runge-Kutta-Scruton and predictor-corrector algorithms |
| $TOL$           | tolerance required                                                                |
| $T_n$           | period of mode with natural frequency of $\omega_n$                               |
| $t, \Delta t$   | time and time increment, respectively                                             |
| $u$             | applied load                                                                      |
| $x$             | independent variable                                                              |
| $Y$             | true value of $y$                                                                 |
| $y$             | dependent variable                                                                |
| $y', y''$       | first and second derivatives, respectively, of $y$ with respect to $x$            |
| $y^{(n)}$       | $n^{\text{th}}$ derivative of $y$ with respect to $x$                             |

|                         |                                                                                                                                                     |
|-------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|
| $\alpha, \beta, \gamma$ | general coefficients                                                                                                                                |
| $\beta$                 | variable used in Newmark Generalised Acceleration method                                                                                            |
| $\varepsilon$           | coefficient used in Wilson- $\theta$ Linear Acceleration and Newmark Generalised Acceleration methods (see Equation (7.9) and (7.18), respectively) |
| $\theta$                | variable used in Wilson- $\theta$ Linear Acceleration method                                                                                        |
| $\kappa$                | coefficient used in Wilson- $\theta$ Linear Acceleration and Newmark Generalised Acceleration method (see Equation (7.10) and (7.19), respectively) |
| $\lambda, \mu, \xi$     | general coefficients                                                                                                                                |
| $\zeta$                 | damping ratio                                                                                                                                       |
| $\tau$                  | increase in time                                                                                                                                    |
| $\phi$                  | variable used in Newmark Generalised Acceleration method                                                                                            |
| $\omega_n$              | natural frequencies                                                                                                                                 |
| $\nabla$                | backward difference operator, e.g. $\nabla y_m = y_m - y_{m-1}$                                                                                     |

## Subscripts

|     |                                                     |
|-----|-----------------------------------------------------|
| $q$ | general integer used in predictor-corrector methods |
| 0   | indicates initial condition                         |

## Superscripts

|     |           |
|-----|-----------|
| $p$ | predictor |
| $c$ | corrector |
| $m$ | modifier  |

One, two or three dots above a variable are used to indicate the first, second or third derivative, respectively, of that variable with respect to time.

### 3. MATHEMATICAL DESCRIPTION

This Item is concerned with methods for the numerical integration of either a set of first-order, or a set of second-order, ordinary differential equations of the form given in Sections 3.1 and 3.2 following.

#### 3.1 First-order Methods

A set of  $n$  first-order ordinary differential equations are defined as

$$\frac{dy_i(x)}{dx} \left( \equiv y_i'(x) \right) = f_i \left( x, y_1(x), y_2(x), \dots, y_n(x) \right), \quad i = 1, 2, \dots, n, \quad (3.1)$$

where  $x > x_0$ , subject to a known set of solutions at a particular point given by

$$y_i(x_0) = (y_0)_i, \quad i = 1, 2, \dots, n, \quad (3.2)$$

where  $x_0$  is the initial value of  $x$ .

The integration problem posed by Equation (3.1) together with the set of “particular” solutions of Equation (3.2) is known as an *initial value problem*.

The methods of numerical integration included in this Item can also be applied to differential equations of higher than first-order by reducing them to a set of first-order equations. This can always be accomplished provided that it is possible to isolate the highest derivative of the dependent variable. This process is illustrated in Appendix A.

It is sufficient, initially, to restrict attention to the case of a single first-order differential equation of the form

$$\frac{dy}{dx} \left( \equiv y'(x) \right) = f \left( x, y(x) \right), \quad (3.3)$$

subject to the initial condition

$$y(x_0) = y_0. \quad (3.4)$$

The solution of Equation (3.3) satisfying Equation (3.4) is given by

$$y(x) = y_0 + \int_{x_0}^x f \left( x, y(x) \right) dx. \quad (3.5)$$

When there is no analytical solution of Equation (3.5), numerical methods need to be used.

This Item considers two types of methods for the numerical integration of first-order differential equations, those methods that are commonly termed Runge-Kutta and those methods that fall within the term predictor-corrector. The range of formulae considered has been restricted to those that have found wide applicability.

## 3.2 Second-order Methods

A set of  $m$  second-order ordinary differential equations are defined as

$$\frac{d^2 y_i(x)}{dx^2} \left( \equiv y_i''(x) \right) = f_i \left( x, y_1(x), y_2(x), \dots, y_n(x), y_1'(x), y_2'(x), \dots, y_n'(x) \right), \quad i = 1, 2, \dots, m, \quad (3.6)$$

where  $x > x_0$  subject to a known set of solutions at a particular point given by

$$\left. \begin{aligned} y_i'(x_0) &= (y_0')_i, \quad i = 1, 2, \dots, m, \\ y_i(x_0) &= (y_0)_i, \quad i = 1, 2, \dots, m, \end{aligned} \right\} \quad (3.7)$$

where again  $x_0$  is the initial value of  $x$ .

The problem posed by Equation (3.6) with the set of “particular” solutions of Equation (3.7) is, like the first-order problem, known as an *initial value problem*.

The methods of direct numerical integration of second-order equations included in this Item can also be applied to differential equations of higher than second-order by reducing them to a set of second-order equations, but only if the order of the equation is even and it is possible to isolate the highest derivative of the dependent variable. The process of reduction is similar to that described in the process applied to the first-order given in Appendix A.

As with the methods for first-order equations it is sufficient, initially, to restrict attention to the case of a single second-order differential equation of the form

$$\frac{d^2 y(x)}{dx^2} \left( \equiv y''(x) \right) = f \left( x, y(x), y'(x) \right), \quad (3.8)$$

subject to the initial conditions

$$y(x_0) = y_0 \quad \text{and} \quad y'(x_0) = y_0'. \quad (3.9)$$

The solution of Equation (3.8) satisfying Equation (3.9) is given by

$$y(x) = y_0 + y_0'(x - x_0) + \int_{x_0}^x \left( \int_{x_0}^x f(x, y(x), y'(x)) dx \right) dx. \quad (3.10)$$

When there is no analytical solution of Equation (3.10), numerical methods need to be used.

This Item considers two methods for the direct numerical integration of second-order differential equations, methods known as the Wilson- $\theta$  Linear Acceleration Method and the Newmark Generalised Acceleration Method.

## 4. RUNGE-KUTTA METHODS

Runge-Kutta methods fall into the category of one-step methods since they allow extrapolation from  $y_m$  to  $y_{m+1}$  simply from information known at  $y_m$ .

These methods are designed to give the same values as a Taylor series truncated to the term in  $h^j$ , where  $h$  is the step size that is equal to  $x_{m+1} - x_m$  and  $j$  is the order of the method. The Taylor series expansion is of the form

$$y_{m+1} = y_m + h y'_m + \frac{h^2}{2!} y''_m + \dots + \frac{h^j}{j!} y_m^{(j)} + \dots \quad (4.1)$$

For the problem  $y' = f(x, y)$  with  $y(x_0) = y_0$  the truncated Taylor series expansion is replaced by a formulae, for  $y_{m+1}$ , of the form

$$y_{m+1} = y_m + h \{ \gamma_1 f(x_m, y_m) + \gamma_2 f(x_m + \mu_1 h, y_m + b_1 h) + \dots \\ \dots + \gamma_j f(x_m + \mu_{j-1} h, y_m + b_{j-1} h) + \dots \}. \quad (4.2)$$

In this expression the functions  $f$  can be expanded using Taylor's mean value theorem and the  $\gamma$ 's,  $\mu$ 's and  $b$ 's determined by equating terms of the expanded Equation (4.2) with terms of Equation (4.1). In order to simplify the derivation it is convenient to express each of the  $b$ 's in Equation (4.2) as a linear combination of the preceding values of  $f$ . The expression then takes the form

$$y_{m+1} = y_m + \gamma_1 k_1 + \gamma_2 k_2 + \dots + \gamma_j k_j, \quad (4.3)$$

which has been truncated to the  $j^{\text{th}}$  term,

$$\left. \begin{aligned} \text{and where } k_1 &= h f(x_m, y_m), \\ k_2 &= h f(x_m + \mu_1 h, y_m + \lambda_{11} k_1), \\ k_3 &= h f(x_m + \mu_2 h, y_m + \lambda_{21} k_1 + \lambda_{22} k_2), \\ &\vdots \\ k_j &= h f(x_m + \mu_{j-1} h, y_m + \lambda_{j-1,1} k_1 + \dots + \lambda_{j-1,j-1} k_{j-1}) \end{aligned} \right\} \quad (4.4)$$

and where the coefficients  $\gamma_1$  to  $\gamma_j$ ,  $\mu_1$  to  $\mu_{j-1}$  and  $\lambda_{11}$  to  $\lambda_{j-1,j-1}$  need to be determined.

The actual derivation of such formulae involve considerable algebraic manipulation and will not be included in this Item but References 12, 18 or 25 (although there are many other references) may be consulted for the full procedure.

Commonly only the 4th-order formulae are used since they adequately compromise error and effort required in functional evaluations. Only a selection of fourth-order Runge-Kutta formulae will be presented in this section but Table 9.1 does provide a more comprehensive list.

## 4.1 Classical Fourth-Order Runge-Kutta

For a first-order differential equation  $y' = f(x, y)$  with the initial condition  $y(x_0) = y_0$  the fourth-order Runge-Kutta algorithm is

$$y_{m+1} = y_m + \frac{1}{6} (k_1 + 2k_2 + 2k_3 + k_4), \quad (4.5)$$

where  $m = 0, 1, 2 \dots$  The coefficients  $k_1$  to  $k_4$  are given by:

$$\left. \begin{aligned} k_1 &= h f(x_m, y_m), \\ k_2 &= h f(x_m + h/2, y_m + k_1/2), \\ k_3 &= h f(x_m + h/2, y_m + k_2/2), \\ k_4 &= h f(x_m + h, y_m + k_3). \end{aligned} \right\} \quad (4.6)$$

The above method can be interpreted as the following geometric process. The slope  $k_1/h$  is evaluated at the point  $(x_m, y_m)$  and is used to extrapolate forward a half step to evaluate the slope there,  $k_2/h$ . This new slope is used to extrapolate forward from  $(x_m, y_m)$  again a half step and another slope,  $k_3/h$ , evaluated. Using this slope a full step extrapolation is made and the slope  $k_4/h$  evaluated. Finally the four slopes are averaged using the weights  $1/6, 1/3, 1/3, 1/6$  and this average slope is used to proceed to  $(x_{m+1}, y_{m+1})$ . It can be noted that if the function  $f$  is third-order or less this formula will be exact.

There is no simple expression for the truncation error for the preceding formula but an estimate of this error can be obtained from the following step-doubling procedure.

Let  $D_n h^5$  equal the truncation error associated with a formula of 4<sup>th</sup>-order accuracy in progressing from  $y_m$  to  $y_{m+1}$ , where  $D_n$  is assumed to be nearly independent of  $h$ , which is small and is assumed to vary slowly with  $n$ . Also let  $Y_{m+1}$  denote the true value of the dependent variable at  $x_{m+1}$ . If  $(2h)y_{m+1}$  and  $(h)y_{m+1}$  are the values of the dependent variable obtained by the approximation in progressing two steps and one double step respectively from  $x_{m-1}$  to  $x_{m+1}$ , then

$$\left. \begin{aligned} Y_{m+1} - (h)y_{m+1} &= 2 D_n h^5 \\ Y_{m+1} - (2h)y_{m+1} &= 2^5 D_n h^5 \end{aligned} \right\} \quad (4.7)$$

$$\text{Thus,} \quad Y_{m+1} = (h)y_{m+1} + \frac{(h)y_{m+1} - (2h)y_{m+1}}{15}. \quad (4.8)$$

The truncation error,  $E$ , is approximated by

$$E = \frac{(h)y_{m+1} - (2h)y_{m+1}}{15} \quad (4.9)$$

and can be evaluated periodically without too much added complication.

An empirical criterion suggested by Collatz (see Reference 8) is that if  $|(k_2 - k_3)/(k_1 - k_2)|$  is greater than a few hundredths then it is wise to decrease the step length,  $h$ .

For a set of  $n$  first-order differential equations of the form of Equation (3.1) given by

$$y'_i = f_i(x, y_1, y_2, y_3, \dots, y_n), \quad i = 1, 2, \dots, n, \quad (4.10)$$

with initial conditions given by Equation (3.2) of the form

$$y_i(x_0) = (y_0)_i, \quad i = 1, 2, \dots, n, \quad (4.11)$$

the modified form of Equation (4.5) is

$$y_{i, (m+1)} = y_{i, m} + \frac{1}{6} \left\{ k_{i,1} + 2k_{i,2} + 2k_{i,3} + k_{i,4} \right\}, \quad (4.12)$$

where  $i = 1, 2, \dots, n$  and  $m = 0, 1, 2, \dots$ .

The coefficients,  $k$ , are given by

$$\left. \begin{aligned} k_{i,1} &= h f_i(x_m, y_{1,m}, y_{2,m}, \dots, y_{n,m}), \\ k_{i,2} &= h f_i\left(x_m + \frac{h}{2}, y_{1,m} + \frac{k_{1,1}}{2}, y_{2,m} + \frac{k_{2,1}}{2}, \dots, y_{n,m} + \frac{k_{n,1}}{2}\right), \\ k_{i,3} &= h f_i\left(x_m + \frac{h}{2}, y_{1,m} + \frac{k_{1,2}}{2}, y_{2,m} + \frac{k_{2,2}}{2}, \dots, y_{n,m} + \frac{k_{n,2}}{2}\right), \\ k_{i,4} &= h f_i(x_m + h, y_{1,m} + k_{1,3}, y_{2,m} + k_{2,3}, \dots, y_{n,m} + k_{n,3}), \end{aligned} \right\} \quad (4.13)$$

where  $m = 0, 1, 2, \dots$

## 4.2 Runge-Kutta-Merson

A Runge-Kutta method that gives an approximation to the single step error is that due to Merson (see Reference 21). This method requires an additional functional evaluation and thus it will require 10 functional evaluations for a two-step extrapolation as against 12 functional evaluations needed for the step-doubling process of Equation (4.9).

The algorithm for this method is presented in Table 9.1 together with the approximate equation for the local truncation error. However the Merson process was derived specifically for use with, and is restricted to, sets of linear equations of the form

$$f_i(x, y_1, y_2, \dots, y_n) = Ax + B_1 y_1 + B_2 y_2 + \dots + B_n y_n + C, \quad (4.14)$$

where  $A, B$  and  $C$  are constants.

## 4.3 Runge-Kutta-Scruton

An alternative method that provides an error estimate at each step of the process is given by Scruton (see Reference 26). The method requires five functional evaluations per step, as does Merson's method, but has no restrictions on its validity, and so, unlike Merson's method, can be used for nonlinear equations.

The algorithm for this method has been included in Table 9.1.

When the bulk of the computation involved lies in the evaluation of the function,  $f(x, y)$ , then this method will require about the same amount of time as Merson's method.

## 4.4 Runge-Kutta-Hull

Another method that allows control of the truncation error but additionally links this error control to the automatic control of the size of step is due to Hull (see Reference 17). This error and step size control interaction is achieved by obtaining a truncation error from the difference between the classical fourth-order method and a "closely coupled" second-order method. This latter method is chosen to be as close to a third-order method as possible, to give a minimum truncation error and to use the same functional evaluations (as many as required) as the fourth-order method. The difference between these two methods then allows an estimate to be made of the truncation error of the second-order method which is used to find the size of the integration step to maintain a prescribed relative error.

The classical fourth-order method is given by Equations (4.5) and (4.6). The second-order Runge-Kutta equation is

$$y_{m+1} = y_m + \frac{1}{3}(k_2 + 2k_3), \quad (4.15)$$

where  $k_2$  and  $k_3$  are found from Equations (4.6). The truncation error,  $E$ , is then

$$E = \frac{1}{6}(k_1 - 2k_3 + k_4). \quad (4.16)$$

If a set of first-order differential equations is being solved then  $E$  is identified from the variable having the largest truncation error. The new step size,  $h_{new}$ , is then found using (see Reference 17) the approximate relationship

$$h_{new} = h \left( \frac{TOL}{E/x_m} \right)^{1/3}, \quad (4.17)$$

where TOL is the user-specified tolerance required by the integration. If  $h_{new} > h$  then the integration step  $h$  is accepted and  $h$  is used for the next integration step. When  $h_{new} < h$  the step  $h$  is rejected and the integration step is repeated using  $h_{new}$ .

The author of this method recommends that only a fraction (say 0.9) of the step predicted by Equation (4.17) be used. This minimises the number of rejected steps. This approach, on the basis of a limited set of results, has been shown to be more efficient than the method of step-doubling of Equation (4.9).