(NASA-CR-124416) A STUDY OF NUMERICAL METHODS OF SOLUTION OF THE EQUATIONS OF MOTION OF A CONTROLLED SATELLITE UNDER THE INFLUENCE OF GRAVITY (Mississippi State Univ.) ~~186~~ p HC $11.50 CSCL 22C G3/31

N73-32762

Unclas 15593

ENGINEERING & INDUSTRIAL RESEARCH STATION

AEROPHYSICS & AEROSPACE ENGINEERING / MISSISSIPPI STATE UNIVERSITY

A STUDY OF NUMERICAL METHODS OF SOLUTION OF THE EQUATIONS OF MOTION OF A CONTROLLED SATELLITE UNDER THE INFLUENCE OF GRAVITY GRADIENT TORQUE

BY

JOE F. THOMPSON

JOHN C. MCWHORTER

SHAHID A. SIDDIQI

SAMUEL P. SHANKS

EIRS-ASE-74-1

COLLEGE ADMINISTRATION

HARRY C. SIMRALL, M.S.
DEAN, COLLEGE OF ENGINEERING

WILLIE L. MCDANIEL, JR., PH.D.
ASSOCIATE DEAN

WALTER R. CARNES, PH.D.
ASSOCIATE DEAN

LAWRENCE J. HILL, M.S.
DIRECTOR, ENGINEERING EXTENSION

CHARLES B. CLIETT, M.S.
AEROPHYSICS & AEROSPACE ENGINEERING

WILLIAM R. FOX, PH.D.
AGRICULTURAL & BIOLOGICAL ENGINEERING

JOHN L. WEEKS, JR., PH.D.
CHEMICAL ENGINEERING

ROBERT M. SCHOLTES, PH.D.
CIVIL ENGINEERING

B. J. BALL, PH.D.
ELECTRICAL ENGINEERING

W. H. EUBANKS, M.ED.
ENGINEERING GRAPHICS

J. E. THOMAS, M.S.
INSTITUTE OF ENGINEERING TECHNOLOGY

FRANK E. COTTON, JR., PH.D.
INDUSTRIAL ENGINEERING

ALLAN G. WEHR, PH.D.
MATERIALS ENGINEERING

C. T. CARLEY, PH.D.
MECHANICAL ENGINEERING

JOHN I. PAULK, PH.D.
NUCLEAR ENGINEERING

WILLIAM D. MCCAIN, JR., PH.D.
PETROLEUM ENGINEERING

For additional copies or information
address correspondence to:

ENGINEERING AND INDUSTRIAL RESEARCH STATION
DRAWER DE
MISSISSIPPI STATE UNIVERSITY
MISSISSIPPI STATE, MISSISSIPPI 39762

TELEPHONE (601)325-2266

A STUDY OF NUMERICAL METHODS OF SOLUTION OF THE EQUATIONS OF MOTION OF A CONTROLLED SATELLITE UNDER THE INFLUENCE OF GRAVITY GRADIENT TORQUE

by

Joe F. Thompson
John C. McWhorter
Shahid A. Siddiqi
Samuel P. Shanks

Report Number EIRS-ASE-74-1

Prepared by

Mississippi State University
Engineering and Industrial Research Station
Department of Aerophysics and Aerospace Engineering
Mississippi State, Mississippi 39762

Under Contract NAS8-28833

for

NATIONAL AERONAUTICS and SPACE ADMINISTRATION
George C. Marshall Space Flight Center
Marshall Space Flight Center, Alabama 35812

July 1973

/

TABLE OF CONTENTS

ABSTRACT

Numerical methods of integration of the equations of motion of a controlled satellite under the influence of gravity-gradient torque are considered. The results of computer experimentation using a number of Runge-Kutta, multi-step, and extrapolation methods for the numerical integration of this differential system are presented, and particularly efficient methods are noted. A large bibliography of numerical methods for initial value problems for ordinary differential equations is presented, and a compilation of Runge-Kutta and multi-step formulas is given. Less common numerical integration techniques from the literature are noted for further consideration.

This report was prepared by Department of Aerophysics and Aerospace Engineering, Mississippi State University under Contract NAS8-28833 for the George C. Marshall Space Flight Center of the National Aeronautics and Space Administration.

INTRODUCTION

The integration of the equations of motion describing the dynamics of a body in orbit, as affected by various perturbing forces and the corrective action of control systems, normally requires the use of hybrid computers because of the difficulty and time involved in integration of high frequency components by present numerical techniques.

It is desirable, however, to be able to describe the vehicular dynamics and the vehicle response to control action numerically in order to take advantage of the advanced development and ease of use of digital computers, particularly on board the vehicle. This can only be achieved by improved mathematical analysis of the numerical integration of the initial value problem with simultaneous ordinary differential equations. While the numerical techniques that have become standard are adequate for the integration of such systems when only low frequency modes are involved, the efficient analysis of high frequency modes requires the development of new approaches.

It is therefore necessary to develop further the mathematical analysis of the numerical integration of systems of simultaneous ordinary differential equations as an initial value problem with specific application to the equations describing the vehicular dynamics of a controlled body in orbit. The ultimate goal of the present project is to develop particularly efficient numerical integration schemes for the equations of motion of a flexible orbiting body rotating under the influence of gravity gradient and control torques.

In the present investigation an extensive bibliography of papers dealing with the numerical solution of systems of ordinary differential equations has been compiled and is included in this report. Many different methods of solution have been obtained therefrom to be compared. In the present effort comparisons of a large number of Runge-Kutta, multi-step, hybrid, and extrapolation methods have been made using the equations of motion of a rigid satellite in circular orbit rotating under the influence of gravity gradient and control torques to a fixed attitude, and the results of these comparisons are reported herein. Further effort is required to extend the comparison to other types of methods, and to compare all methods in regard to the equations of motion of non-rigid satellites.

CHAPTER I

EQUATIONS OF MOTION

For a rigid space vehicle in orbit, six ordinary differential equations are required to specify its spatial motion--three equations for rotations and three for displacements. If the vehicle is considered non-rigid, more equations (partial differential equations) are required to specify the deformations of the body.

In this study the body is considered rigid. The equations of motion are formulated in terms of quaternions, these being functions of the direction cosines of the body axes. The introduction of these quaternions results in seven first order ordinary differential equations which specify the motion of the body.

The vehicle is in orbit and the motion is assumed to be affected only by earth's gravitational field. The space vehicle is to be kept attitude fixed, that is, its direction should be invariant for all time. Any deviations from this fixed direction are to be corrected by an on-board control system. Deviations from the fixed attitude occur because of the gravity gradient and imperfections in the launch process. These deviations are indicated by the quaternions.

The seven first-order ordinary differential equations, in matrix form are:

$$\underline{I}\cdot\underline{\dot{w}} + \underline{w} \times \underline{I}\cdot\underline{w} = \frac{3GM}{|\underline{r}|^5}\,\underline{r} \times \underline{I}\cdot\underline{r} + \underline{T_c} \qquad 1.1 \text{ (3 equations)}$$

$$\underline{\dot{q}} = -\frac{1}{2}\,\underline{\Omega}(\underline{w})\underline{q} \qquad 1.2 \text{ (4 equations)}$$

where

$$\underline{w} = \begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix}$$ is the angular velocity vector,

$\dot{\underline{w}}$ is the time rate of change of $\underline{w}$,

$$\underline{q} = \begin{pmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{pmatrix} = \begin{pmatrix} e_1 \sin \psi/2 \\ e_2 \sin \psi/2 \\ e_3 \sin \psi/2 \\ \cos \psi/2 \end{pmatrix}$$ is the quaternion vector,

$\dot{\underline{q}}$ is the time rate of change of $\underline{q}$,

ψ is the angle of rotation of the body's rotation vector $\underline{e}$ from a fixed attitude,

$\underline{T_c}$ is the control torque vector, a function of $\underline{w}$ and $\underline{q}$,

$$\underline{I} = \begin{pmatrix} i_{11} & i_{12} & i_{13} \\ i_{21} & i_{22} & i_{23} \\ i_{31} & i_{32} & i_{33} \end{pmatrix}$$ is the moment of inertia matrix,

$\underline{r}$ is the radius vector of the orbit referred to the vehicle fixed axes and is related to the inertial axes, fixed to the center of the earth, by a transformation matrix $\underline{D}$,

G is the universal gravitational constant,

M is the mass of the earth,

$\underline{\Omega}(\underline{w})$ is an asymmetric matrix function of $\underline{w}$ as defined below:

$$\underline{\Omega}(\underline{w}) = \begin{pmatrix} 0 & -w_3 & w_2 & w_1 \\ w_3 & 0 & -w_1 & w_2 \\ -w_2 & w_1 & 0 & w_3 \\ -w_1 & -w_2 & -w_3 & 0 \end{pmatrix}$$

Since $\underline{I}$ is non-singular, we may write (1.1) as

$$\dot{\underline{w}} = \underline{I}^{-1} \left[\frac{3GM}{|\underline{r}|^5} \underline{r} \times \underline{I} \cdot \underline{r} - \underline{w} \times \underline{I} \cdot \underline{w} + \underline{T_c} \right] \qquad 1.3$$

Evaluating the matrix cross products and defining an asymmetric matrix function $\underline{F}(\underline{\alpha})$, where $\underline{\alpha}$ is any vector, by

$$\underline{F}(\underline{\alpha}) = \begin{pmatrix} 0 & -\alpha_3 & \alpha_2 \\ \alpha_3 & 0 & -\alpha_1 \\ -\alpha_2 & \alpha_1 & 0 \end{pmatrix}$$

we have

$$\dot{\underline{w}} = \underline{I}^{-1} \left[\frac{3GM}{|\underline{r}|^5} \underline{F}(\underline{r}) \underline{I}\ \underline{r} - \underline{F}(\underline{w}) \underline{I}\ \underline{w} + \underline{T_c} \right] \qquad 1.4$$

$$\dot{\underline{q}} = -\frac{1}{2} \underline{\Omega}\ (\underline{w})\ \underline{q} \qquad 1.5$$

The transformation of $\underline{r}_I$ (inertial axes) to $\underline{r}$ (vehicle axes) is given by

$$\underline{r}_I = R_0 \begin{pmatrix} \cos w_0 t \\ \sin w_0 t \\ 0 \end{pmatrix} \qquad 1.6$$

$$\underline{r} = \underline{D}\ \underline{r}_I \tag{1.7}$$

where

$$\begin{aligned}
D_{11} &= q_1^2 - q_2^2 - q_3^2 + q_4^2 \\
D_{12} &= 2(q_1q_2 + q_3q_4) \\
D_{13} &= 2(q_1q_3 - q_2q_4) \\
D_{21} &= 2(q_1q_2 - q_3q_4) \\
D_{22} &= -\ q_1^2 + q_2^2 - q_3^2 + q_4^2 \\
D_{23} &= 2(q_2q_3 + q_1q_4) \\
D_{31} &= 2(q_1q_3 + q_2q_4) \\
D_{32} &= 2(q_2q_3 - q_1q_4) \\
D_{33} &= -\ q_1^2 - q_2^2 + q_3^2 + q_4^2
\end{aligned} \tag{1.8}$$

The control torque vector is given by

$$\underline{T_c} = \underline{A}\ \underline{q} + \underline{b}\ \underline{w} \tag{1.9}$$

where

$\underline{A}$ is a control matrix (with dimensions of torque), taken in the present comparisons as

$$\underline{A} = 2 \begin{pmatrix} 4250 & 0 & 0 \\ 0 & 39{,}950 & 0 \\ 0 & 0 & 39{,}950 \end{pmatrix} \qquad \text{nt - m}$$

$\underline{b}$ is a control matrix (with dimensions of angular momentum):

$[\underline{B} = w_0\underline{b}]$

$$\underline{b} = \begin{pmatrix} -3400 & 0 & 0 \\ 0 & -8800 & 0 \\ 0 & 0 & -8800 \end{pmatrix} \qquad \text{kg - m}^2/\text{sec}$$

For the spacecraft under consideration in these comparisons, the moments of inertia in MKS units are:

$$\underline{I} = \begin{pmatrix} 5{,}320 & 0 & 0 \\ 0 & 88{,}950 & 0 \\ 0 & 0 & 89{,}120 \end{pmatrix} \qquad \text{kg-m}^2$$

with: $GM = 3.98602 \times 10^{14}\ \text{m}^3/\text{sec}^2$

$R_0 = 6.6525535 \times 10^7$ m for a 90 minute circular orbital period.

Equations 1.4 and 1.5 can be expressed in vector form as an initial value problem as follows:

$$\frac{d\underline{Y}}{dt} = \underline{F}(\underline{Y}, t) \qquad\qquad 1.10$$

$$\underline{Y}(t_0) = \underline{Y}_0$$

where:

$$\underline{Y} = \begin{pmatrix} w_1 \\ w_2 \\ w_3 \\ q_1 \\ q_2 \\ q_3 \\ q_4 \end{pmatrix}$$

t_0 is the initial time and

$$\underline{Y} = \begin{pmatrix} w_1(t_0) \\ w_2(t_0) \\ w_3(t_0) \\ q_1(t_0) \\ q_2(t_0) \\ q_3(t_0) \\ q_4(t_0) \end{pmatrix}$$

CHAPTER II

NUMERICAL EXPERIMENTATION PROCEDURE

The complexities of the system preclude an analytical solution, and hence as the true solution is unknown, the various numerical integration methods could only be compared with each other. The basis of comparison was the estimated truncation error as detailed below. No account of round-off error was made because this error depends on the computer used and the number of computations made in each method.

We assume that the local truncation error for a p-order single-step method has the form

$$T(t, h) = g\ h^{p+1} \tag{2.1}$$

where g is the principal error function (assumed essentially constant). Then proceeding from (t_{n-1}, Y_{n-1}) to (t_{n+1}, Y_{n+1}), using two steps, each of size h, the truncation error is approximately

$$T(t_{n+1}, h) = g\ 2(h^{p+1}) \tag{2.2}$$

so that

$$Y(t_{n+1}) - Y_{n+1}^{(h)} = 2\ g\ h^{p+1} \tag{2.3}$$

With the same method, going from (t_{n-1}, Y_{n-1}) to (t_{n+1}, Y_{n+1}) in one step of size 2h, the truncation error is

$$T(t_{n+1}, h) = g(2h)^{p+1} \tag{2.4}$$

$$Y(t_{n+1}) - Y_{n+1}^{(2h)} = 2^{p+1}\ g\ h^{p+1} \tag{2.5}$$

Subtracting 2.3 from 2.5 gives

$$Y_{n+1}^{(h)} - Y_{n+1}^{(2h)} = g\ h^{p+1}[2^{p+1} - 2] \tag{2.6}$$

Then approximately

$$T(t_{n+1}, h) = \frac{Y_{n+1}^{(h)} - Y_{n+1}^{(2h)}}{2^{p+1} - 2} \qquad 2.7$$

Equation 2.7 was used to compare different methods. This is not an exact expression for the truncation error because g is not completely constant. Still this expression is a useful way to compare methods because it gives some measure of the truncation error involved. This estimate applies strictly to single-step methods, but was used for all methods since it still is some measure of the error and should tend to zero for any method as the step size decreases to zero even though it is not strictly to be interpreted as truncation error for multi-step methods.

Each method tested was run for the maximum step size, and then a number of runs were made with the step size halved successively. Equation 2.7 was used to estimate the error between the solution using step size h, and the solution using h/2. This error was calculated at each time step and a root mean square of these errors was calculated. This RMS value was used to judge different methods.

The initial conditions used for the comparison were

$$\underline{w}(t_0) = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \qquad \underline{q}(t_0) = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

The period of orbit was 90 minutes. The maximum step size used was $h = 0.001$.*

---

*All times and time steps without units are fractions of the orbit period.

Figure 1 illustrates a typical solution up to t = 0.25 (quarter of an orbit).

With the initial conditions used, w_1, w_2, q_1, q_2, q_4 always remain unchanged, while w_3 and q_3 oscillate. This is to be expected as the gravity differential on the spacecraft is the only disturbing moment, and hence only w_3 and q_3 are affected. q_3 oscillates at twice the orbital period, and has no transient phase. w_3, however, has a transient phase, and dictates the step size, h, necessary for stability. For this reason, only the truncation error committed in w_3 was used to judge the methods tested. The error in w_3 by far outweighs the error in q_3.

Most methods were unstable for h = 0.001. With each method six more h's were run, with h halved for each run. Figure 1 shows that the fast ocsillations of w_3 die down at about t = 0.02. The comparison scheme discussed above was used to calculate the RMS truncation error up to t = 0.04. This interval includes a period of fast oscillation, up to t = 0.02, and a period of slow periodic oscillation, up to t = 0.04. The h = 0.001 runs will obviously have the maximum truncation error, and each successive halving of h will reduce this error because as $h \to 0$ the truncation error $\to 0$.

All runs were made in single precision arithmetic on a UNIVAC 1106 computer.

CHAPTER III

RESULTS OF COMPARISONS

Runge-Kutta Methods*

In Table I Runge-Kutta methods are compared by two parameters: RMS truncation error and computational time. The computational time is calculated as the number of function evaluations required by the particular method, i.e., the number of stages, V, for Runge-Kutta methods. Thus computation time is in units of number of function evaluations. Each method compared is identified by its name and its equation number in Appendix I.*

An X under a step size h, indicates that the method was unstable for that h. A - under a step size h, indicates that this step size was not run with this method. A 0 will occur for the smallest step size run because no truncation error can be calculated for the smallest step size.

Figure 2 shows examples of the errors in w_3 and q_3 for various step sizes plotted against time t. Figure 3 shows examples of the RMS T plotted against step size for w_3. Both these figures are for the fourth-order Runge-Kutta-Ralston method.

Selecting a best method out of the ones tested is a judgment problem and depends on the users' requirements. Most users require both speed and accuracy. Hence the methods of Table I were judged on the basis of an error-time parameter, ET. This is a judgment

---

*Shahid Ahmed Siddiqi, "A Comparison of Various Order, Single-Step Explicit Runge-Kutta Methods Used to Solve the Equations of Motion of a Rigid Spacecraft in Circular Orbit" (unpublished M.S. thesis, Mississippi State University, 1973).

parameter which gives an equal weight to both the error and the computational time of a particular method:

$$ET = \left(\frac{RMS}{2^{p+1} - 2} \frac{T}{}\right)\left(\frac{V}{h}\right)$$

Based on this parameter, ET, the following observations are immediately made from Table I:

Best fourth-order RKE: Ralston RKE(4,4)*

Best fifth-order RKE: Butcher RKE(5,6)

Best sixth-order RKE: Shanks RKE(6,6)

Best seventh-order RKE: Sarafyan RKE(7,10)
(except for h = 0.000125)

Best eighth-order RKE: Shanks RKE(8,10)
(except for h = 0.000125)

Now from among these five methods, the best choice, based on the ET parameter, is Shanks RKE(8,10) with h = 0.00025.

The results can be interpreted in another way: it is better, i.e., smaller ET, to use the Butcher RKE(5,6) at h = 0.000125 than to use the Shanks RKE(8,10) at h = 0.0005. Another observation, evident from Table I, is that the sixth, seventh, and eighth-order methods reach minimum ET's, but not at the smallest step size.

It is again emphasized that the judgment parameter ET gives equal weight to error and time.

<u>Multi-Step Methods</u>

Although a complete investigation of the multi-step methods is desirable, the available time permitted only a limited investigation.

---

*RKE(q,r) refers to an explicit Runge-Kutta method of q-order with r stages.

With the time factor in mind, three of the most promising types of predictors were chosen - Adams-Bashforth (AB), Krough, and Craine-Klopfenstein (CK), and three types of correctors were chosen - Adams-Moulton (AM), Rodabough-Wesson (RW) and Wesson. Also, a Butcher fifth order hybrid method was chosen to compare with the multi-step methods.

The predictor and corrector equations were used to solve the initial value problem in various P-C combinations in both the PEC and PECE modes. Also the $PE(CE)^s$ mode was run with an accelerated Jacobi scheme; however, the acceleration parameter was found to be zero for this solution. Thus, the $PE(CE)^s$ mode was discarded. The result of the experimentation is presented in Table II.

From Table II, the best of the three types of predictors was the CK predictor, and the best of the three types of correctors was the AM corrector. To verify these conclusions the following observations were made:

1) The three predictors were used in P-C combinations with the sixth-order AM corrector. The fourth-order CK was found to give a truncation error 9.4 times less than that of the fourth-order Krogh predictor for a time step of .000125. Also, the fourth-order CK was found to give a truncation error 16.5 times less than that of the fourth-order AB predictor at the same time step.
2) The fifth-order Butcher hybrid gave 3 times less truncation error than the eighth-order AB-AM combination for a time step of .000125. However, the sixth-order CK-AM combination gave 2 times less truncation error than the fifth-order Butcher hybrid method.

3) The fourth-order CK-sixth-order AM combination gave 6.5 times less truncation error than the eighth-order AB-AM combination.

4) The RW correctors gave solutions that did not closely compare with the other Runge-Kutta and multi-step solutions. Thus, the RW correctors were discarded.

5) The fourth-order CK-eighth-order AM combination had slightly less truncation error than the fourth order CK-eighth-order Wesson combination for all timesteps considered.

Based on the merit factor, ET, introduced above, the fourth-order CK-eighth-order AM combination is the best of those considered.

The best multi-step methods considered are generally less effective than the best Runge-Kutta methods when judged by this merit factor, and stability limitations preclude the use of the larger step sizes allowed with the Runge-Kutta methods with the higher-order methods. It thus appears that the best Runge-Kutta methods are to be preferred for this system of differential equations.

Extrapolation Methods

Six extrapolation methods were investigated. These methods used rational function and polynomial function extrapolation with the Euler and modified midpoint algorithms. The Euler algorithm was used with polynomial function extrapolation with the basic time step being subdivided according to each of the sequences

$$h_k = h_0/2^k$$

and

$$h_k = \{h_0,\ h_0/2,\ h_0/3,\ h_0/4,\ \cdots\} \quad .$$

The modified midpoint rule was used with both rational function and polynomial function extrapolation with each of the above sequences.

All methods used local extrapolation. The A-stable trapezoidal rule was not used since it requires global extrapolation. Further details on the Euler-Romberg and Bulirsch-Stoer modified midpoint method can be found in Reference [556].

Since the extrapolation methods subdivide each time step repeatedly until a desired tolerance between successive extrapolations is obtained, comparisons with other methods is not directly possible. However, some conclusions can be drawn concerning the best of the extrapolation methods with regard to accuracy and number of function evaluations.

Extrapolation methods have as variable parameters the basic step size and the tolerance between successive extrapolations. In this work there is no automatic step size correction. Results for comparison purposes were obtained by making a number of computer runs at different basic step sizes with constant tolerance between successive extrapolations and then repeating the runs with successively smaller tolerances between successive extrapolations. These results indicate an optimum value of tolerance and step size for each method. As the basic step size increases, the number of extrapolations (function evaluations) needed to obtain a given tolerance decreases. However, for large step sizes the accuracy of the extrapolated values tends to decrease. The optimum values must be determined for each method experimentally. If the tolerance between successive extrapolations is too low, accuracy is poor, while if too high, instability can occur.

All midpoint rule methods became unstable at a basic step size of .00037 (fraction of one orbit), some at a smaller step size. So apparently extrapolation cannot totally overcome the instability characteristic of the midpoint rules. When using a halving sequence, polynomial function extrapolation required more function evaluations than the rational function extrapolation with little difference in accuracy. If a reciprocal sequence was used the polynomial function extrapolation required fewer function evaluations than the rational function extrapolation method but gave less accurate results and went unstable at a lower step size. Thus there is little advantage to either rational function or polynomial function extrapolation, and the reciprocal and halving sequences produced no real savings in function evaluations when used with the midpoint rule.

Of the two Euler methods which both used polynomial function extrapolation, the solution obtained with the reciprocal sequence was very slightly more accurate than the solution obtained by successively halving the time step. The number of function evaluations was from 2 to 8 times fewer for the solution with the reciprocal sequence of step size subdivision, depending upon the smallness of the tolerance between successive extrapolations.

Partial results are presented in Figure 4 . The best two methods are shown, these being the Euler algorithm with polynomial function extrapolation utilizing a reciprocal sequence for step size subdivision (ERPESR) and the modified midpoint rule with rational function extrapolation utilizing a halving sequence (BSRESH).

The accuracy of both methods is seen to be a function of step size and tolerance between successive extrapolations. BSRESH approaches the accuracy of ERPESR but requires about 40 times as many function evaluations at a step size of .000185 (fraction of one orbit). At a step size of .00037 the order of the function evaluations is the same but BSRESH goes unstable at higher values of time. Extrapolation to higher tolerances increases accuracy, particularly at lower step sizes, at the expense of increased function evaluations for both BSRESH and ERPESH. However, there is a limit to the tolerances for both methods below which the solution becomes unstable. And, of course, if the tolerance is too large accuracy decreases.

ERPESH had constant accuracy over a large range of tolerances (10^{-4} - 10^{-10}) for step sizes between .0000231 and .000185. However, at larger step sizes the accuracy of the solution decreased at lower tolerances (10^{-4}).

Thus the best of the extrapolation methods is ERPESR, Euler algorithm with polynomial function extrapolation using a reciprocal sequence to subdivide the basic step size until a tolerance between successive extrapolation of 10^{-4} to 10^{-10} is achieved. This method gives better accuracy than the Runge-Kutta and predictor corrector methods and takes about the same number of function evaluations. A value of angular velocity obtained by the best of the predictor corrector methods is shown on Figure 5 at a step size of .000185. This value is 1.4% lower than the value obtained by ERPESR and required 40 function evaluations as compared to 41 function evaluations for ERPESR when the extrapolation was carried to a tolerance of 10^{-4}, or 71 function evaluation for a tolerance of 10^{-10}. Thus ERPESR appears to require the same order of function evaluation as Runge-Kutta and predictor corrector methods at a better accuracy.

CHAPTER IV

SURVEY OF TECHNIQUES FOR FURTHER CONSIDERATION

The numerical solution of the initial value problem

$$\underline{y}' = \underline{f}(t,\underline{y}) \quad , \quad \underline{y}(0) \equiv \underline{y} \ . \qquad 4.1$$

has been approached in various ways as outlined below. As is noted, the various classes cited are not exclusive, but overlap one another.

Quadrature Methods

One may write formally

$$\underline{y}(t_r) = \underline{y}(t_q) + \int_{t_q}^{t_r} \underline{f}\ (t,\ \underline{y}(t))\ dt \qquad 4.2$$

In the quadrature methods, the integral in 4.2 is approximated by a numerical quadrature expression. For instance, suppose it is desired to calculate the numerical approximation, $\underline{y}_n$, at a series of times, t_n, $n=0, 1, 2, \cdots$, called the nodes hereafter for convenience of notation, these nodes not necessarily being equally spaced. Then we may approximate the integral of 4.2 by a numerical quadrature consisting of a linear combination of nodal values and possibly also some intermediate values at points interspersed among the nodes. The coefficients in the linear combination and the locations of the nodes and/or the inter-nodal points are determined by the particular type of numerical quadrature employed. These factors are obtained directly from quadrature expressions for some types of methods, but by only indirect reference to numerical quadrature in others, the former type being that strictly referred to as "quadrature methods" in many works.

If only previously calculated nodal values, and possibly the value at the node presently under consideration, are included in the quadrature formula, then no additional equations are required, and the value at the node presently under consideration is determined either explicitly or implicitly through the solution of a nonlinear equation, depending on whether or not the value at the node under consideration is included in the quadrature expression. Methods of this type are also a sub-class of the linear multi-step methods discussed below.

> Nesterchuk [378] gives such an implicit method involving all previous nodal points.

If nodal points beyond the node presently under consideration are included in the quadrature formula then an equation must be written for each of these points, and a system of equations must then be solved simultaneously for all the unknown nodal values involved. This type method is also a sub-class of the block linear multi-step methods discussed below.

> Day [116] gives a method of this type using Lobatto quadrature and successively higher moments of the differential equation to supply the needed equations for all the unknown nodal values included in the block. For linear equations this method achieves n+1 order with n function evaluations and is reported to be superior in accuracy to some Runge-Kutta methods.

If inter-nodal points are involved, additional equations for the values at these points must be included. If these inter-nodal values are determined in turn by quadrature formulas, we have the standard Runge-Kutta methods. Within this class we have explicit methods if the expressions for the inter-nodal values involve only values at previous inter-nodal points, semi-implicit methods if these expressions involve the presently considered inter-nodal value as well but none beyond, and implicit methods if inter-nodal values beyond that presently considered are included. The latter types require, respectively, the solution of one nonlinear equation at each inter-nodal point or the simultaneous solution of a system of equations, one for each internodal point.

The method of Hulme [253] is a generalization of this implicit Runge-Kutta form constructed using Gaussian quadrature to approximate the integrals involved in a Galerkin approximation with the solution represented by a polynomial within each step. This produces a piecewise continuous solution with essentially the same effort required by conventional implicit Runge-Kutta methods and the same nodal values as produced thereby. Axelsson [13], in a general discussion of quadrature methods, presents implicit Runge-Kutta methods based on Radau and Lobatto quadratures. Implicit Runge-Kutta methods are also discussed in Gourlay [194], where it is noted that certain methods stable for linear equations, i.e., with constant Jacobian matrices, may not be stable when the Jacobian varies as is the case with nonlinear equations. The common trapezoidal method is a case in point, and a stable modification of the same order is given therefor.

Explicit Runge-Kutta methods with provision for intrinsic truncation error estimation are given in Warten [536], Sarafyan [448], and Zonneveld [553]. Haines [220] gives a semi-implicit Runge-Kutta method with the coefficients chosen to increase stability and with the Jacobian matrix, required in such methods, evaluated by finite differences. Semi-implicit Runge-Kutta methods are also given by Allen [3].

Sarafyan [452] fits a Hermite polynomial to the inter-nodal values of an explicit embedded Runge-Kutta method to produce a continuous approximation of order only one less than that of the discrete approximation. Merson [356] gives Runge-Kutta methods for which each inter-nodal value is required to be a good approximation of the solution at the corresponding point. Treanor [519] uses local linearization before the quadrature to develop a modification of the Runge-Kutta type. The algebraically difficult derivation of the coefficients involved in Runge-Kutta methods is relegated to the computer via a program of Sarafyan and Brown [447]. Many other Runge-Kutta methods are discussed in Appendix I.

The more general inclusion of inter-nodal points along with nodal points in the quadrature expression for the integral of 4.2 results in a combination of the above types, variously referred to as multi-step Runge-Kutta methods or hybrid methods.

> Rosen [430] gives an explicit multi-step Runge-Kutta method using at each node all the inter-nodal values used at the previous node, with a resultant decrease in the number of evaluations per step required for a given order.

Finally, in the manner already discussed above, the inclusion of nodal and/or inter-nodal points beyond the node presently under consideration produces block methods of the Runge-Kutta or hybrid type, requiring simultaneous solution for all the unknown nodal and/or inter-nodal values involved.

> Block Runge-Kutta methods in which the iteration at each node is intentionally not carried to convergence are discussed by Rosser [433]. These methods require fewer evaluations per step for a given order than the usual point Runge-Kutta methods.

<u>Multi-value Methods</u>

At each nodal point, one may write, more generally, the value of the dependent variable, any of its derivatives, and any functions or combinations thereof as linear combinations of these quantities at each node and/or at inter-nodal points, and so produce the almost all-inclusive class of multi-value methods.

> Methods of this type are discussed in general in Gear [177]. Methods with variable step size are discussed in regard to stability in Tu [520].

The entire class of quadrature methods discussed above is a subclass of the multi-value methods for which the value of the dependent variable, $\underline{y}$, at only one previous nodal point is involved, and the linear combination includes only the first derivative, $\underline{y}' = \underline{f}\,(t, \underline{y})$.

The use of values of only the dependent variable and its first derivative in the linear combination produces the linear multi-step methods. These methods are explicit if only values of the derivative at previous nodes are included, semi-implicit (commonly called implicit) if the derivative at the presently-considered node is included as well but none beyond, and implicit (commonly called "block") if nodal values beyond that presently under consideration are included. The latter types require, respectively, the solution of a nonlinear equation at each node or the simultaneous solution of a system of equations for all the unknown nodal values involved. Again inter-nodal points may be included as well, and the so-called hybrid methods developed as discussed above. The more common types of predictor-corrector methods, in which values are "predicted" by explicit multi-value forms and then "corrected" using implicit multi-value forms involving the predicted value as well as previously calculated values are in this class. Additional applications of the corrector may follow, using the most recently corrected values.

Hull and Creemer [244], in a comparison of various predictor-corrector forms, state that error, stability, and the order required for a given error at least computation are less sensitive to order with two corrector applications than with one, with no significant improvement thereafter. Lambert [308] shows the equivalence of any predictor-corrector method using a finite number of corrector applications to some explicit multi-step method. This limits the expectations for stability of predictor-corrector methods. Donelson and Hansen [129] use a set of correctors applied cyclically over a set of time steps and thus achieve higher-order stable methods than are possible with the same corrector used at all steps. Order 2k-1 is thereby achieved for stable k-step methods, as opposed to the maximum order, k-1, possible with stability for methods using only a fixed corrector. The use of variable step size in predictor-corrector methods has been considered by Van Wyk [532], and in a single-step method of the multi-step form by Richards, Lanning, and Torrey [419]. Other predictor-corrector methods are given in Appendix II.

Many multi-step methods have been developed in which some coefficients are chosen to increase stability as in Krough [291], with a comparison of stability plots for various methods; Crane and Lambert [96]; Nigro [383]; Rahme [406], with an attempt to minimize error while maintaining stability; and Lomax [332], considering Runge-Kutta and hybrid methods as well. A multi-step method with the coefficients chosen to fit the characteristic roots to some of the characteristic exponentials of the differential system is given by Miranker [366] for nonlinear systems with local linearization, the method being implicit, and for linear systems by Liniger and Willoughby [323], an explicit method. Osborne [398] gives a single-step method with the nodes chosen to give desired characteristic roots. Timlake [518] increases the stability of multi-step methods by averaging over previous steps, determining the weights from the Jacobian eigenvalues. The stability of multistep methods has been analyzed particularly in Karim [258,256], sufficient conditions for instability being given in the former and the effect of the predictor as well being included in the latter; Dahlquist [100-106], a series of classic papers; and Hafner [217], giving stability charts for a large number of Runge-Kutta and hybrid methods as well.

Implicit multi-step methods particularly for stiff systems have been given by Ratliff [411], giving a comparison of several methods; Jain and Srivastava [246,247]; Dill [126], a systematic search for such methods; Gelinas [182]; Dill and Gear [125], again a search; and Gear [172]. Such methods achieve a particular shape of stability boundary suitable for stiff systems by always including the derivative at the present node with perhaps the derivatives at a few previous nodes.

Tyson [521] gives an implicit multi-step method for nonlinear systems using local linearization about a predicted value, rather than the previous nodal value, that achieves order two greater than that of the predictor used. (Local linearization is used in many methods to render nonlinear systems tractable by methods restricted to linear systems, but such linearization about the previous nodal value restricts the order to two regardless of the method used.) Boggs [30] used Broyden iteration (involving direct approximation of the Jacobian inverse rather than inversion of the Jacobian) with implicit multi-step methods. Block methods are given by Daniel [109,108] and by Shampine and Watts [469]. Methods for determining the necessary set of starting values (a problem equivalent to the development of block multi-step methods) have been given in Reimer [412], Rakitskii [407], and Alonzo [4].

The addition of inter-nodal points (hybrid methods) has been considered by Papian and Ball [399]; Lomax [332], who gives a method for choosing the coefficients in explicit methods of this type to improve stability; Gragg and Stetter

[195]; discussing the coefficients therein for maximum order; Gear [175]; and Hafner [217], who gives a large number of stability charts. Additional consideration of multi-step methods has been given by Spijker [492], considering the question of direct application to higher-order equations or a split to a system of first-order equations, and by Zverskina [555], using derivatives only at previous nodes separated by some distance from the present node.

Various other quantities, such as higher derivatives and divided differences, have been included in the linear combination, and methods using one type of quantities may be derived by transformation of methods using another type as in Osborne [395] and Kohfeld and Thompson [283]. Again the above-mentioned ramifications involving inter-nodal points and/or nodal points beyond that presently under consideration may be developed. A further modification is that values at previously passed nodes may be subsequently changed. It is possible to derive methods of this type that use higher derivatives or higher-order divided differences, all of which are produced at each node from linear combinations of the same, in lieu of values at previous nodes. In this manner more efficient step size change can be accomplished.

In Grobner, Kohnert, Reitberger and Wanner [207] higher derivatives are included in generalizations of both implicit Runge-Kutta and multi-step type methods. Multi-value methods involving higher-order derivatives are discussed in Gear [177,178]; Sloan [484]; Lewis and Stovall [321]; and Nordsieck [387], a single step form with ease of step-size change; and Osborne [395]. Kohfeld and Thompson [283] give a method using higher derivatives and inter-nodal points. The single-step implicit method of Urabe [522] uses second derivatives. Makinson [348] gives a single-step implicit method using higher derivatives, the form being specially constructed to preserve the original sparseness in the matrix solution of the implicit equations. Lomax [334] presents a single-step implicit method using second derivatives with coefficients chosen to increase stability. Ehle [132] and Davison [110] also give single-step methods using higher derivatives, and Reimer [414] analyses multi-step methods using higher derivatives.

Non-Polynomial Interpolates

In the great majority of methods the linear combinations involved in the multi-value methods, and thus in the quadrature methods, are required to be exact when the dependent variable is a polynomial of some degree. This restriction may be relaxed, and an even broader spectrum of methods may be developed by requiring these linear combinations to be exact for some other function, in particular for perhaps some class of functions especially adapted to the particular differential equation being considered.

> Loscalzo and Talbot [339] use a spline function with the higher derivatives matched at the nodes, which, while producing the same nodal values developed by a multi-step method, is a single-step method and gives a piecewise continuous approximation. It does, however, require differentiation. Byrne and Chi [45] use a spline approximation of the intergrand in a quadrature type method requiring no differentiatian. Callender [71] uses a spline approximation spanning several nodes to develop a block multi-step method.
>
> Blue and Gummel [27] and Lambert and Shaw [306] use rational function approximation of matrix exponentials, the former using coefficients chosen to increase stability and producing thereby Pade approximates of the exponential. The latter method matches higher derivatives and thus requires differentiation. Roe [84] gives a multi-step method using a combination of a polynomial and an exponential as the interpolate. Other non-polynomial interpolates have been used by Shaw [475], Lambert and Shaw [303], and Pope [403], the first two being primarily of use when singularies are involved in the solution.

Only a relatively small number of methods of this type have been considered, since the intention of most developments has been to develop methods that may be applied at least fairly well to all types of differential equations, hence the use of polynomial interpolation. In this connection the remarks of Lomax [334] are pertinent:

> "It is unlikely that 'new' combinations of linear equations that connect a function, u, and its derivative, u', at a series of reference points, equispaced or not,

> will improve existing methods for the numerical integration of general sets of coupled ordinary differential equations..." (p. 2). "What appears to be needed are studies of special methods designed for special methods designed for special classes of equations..." (p. 6).

Iterative Methods

Methods of this class start with some approximate solution, such as the solution of a related but simpler differential equation or the result of any numerical method, and improve the approximation by an iterative procedure. One such method is the Lie series method. The extrapolation methods by which solutions with successively smaller truncation errors are obtained by extrapolating results with successively smaller step-sizes may also be considered iterative in the sense stated.

> In reference [207] Runge-Kutta methods having error expansions in even powers of the step size are developed to serve as the basis for these extrapolation methods. Lie Series methods have been given by Knapp and Wanner [279,280].

Transformation Methods

In methods of this type the dependent and/or independent variable is transformed before numerical solution in order to obtain a form with better numerical properties. In this manner stiff systems may be transformed into systems with smaller ranges of eigenvalues.

> Lawson [313] gives a transformation of the dependent variable which reduces the stiffness of the system, but the method requires the use of matrix exponentials. A similar transformation of the dependent variable is used by Jain [259]. Other transformations of the dependent variable are given by Decell, Guseman and Lea [117], requiring evaluation of the Jacobian inverse, and by Calahan [67].

Higher-Order Equations

As is well known, any higher-order differential equation can be broken into a set of simultaneous first-order equations to which the preceding methods are applicable. Generalizations of the classes cited follow readily, however, so that application may be made directly to the higher-order equation if such is thought to be preferable. This is still something of an open question.

Osborne [397] gives a quadrature method for linear higher-order equations. Day [112,113] has implicit Runge-Kutta type methods using Gaussian and Lobatto quadrature, respectively, for second-order equations with no first derivative. Cooper [91,93] gives a generalization of the implicit Runge-Kutta methods to equations of any order and states that the split into a set of first-order equations decreases the local accuracy and increases the computation required, with little effect on the global accuracy.

The multi-value methods using higher derivatives are directly applicable to higher-order equations, and Gear [177] states also that direct solution of the higher-order equations may be faster than that of the split system of first-order equations. Methods of this type are also given in Gear [178, 484] and Allen [2]. Adrianova [1] found less error without splitting to the first-order system. However, Spijker [503] states that round-off error is reduced by splitting into first-order equations.

Conclusions and Directions

Implicit methods in general are much more stable than explicit methods, but may still be less efficient because of the difficulty of the solution of the system of simultaneous equations that is required at each time step. The key to the use of the more stable implicit schemes is thus the iterative solution of this system, and effort should be directed toward the development of faster and more appropriate iterative schemes for use in these methods. There are, of course, a great many general iterative schemes that have never been applied in implicit numerical solutions of ordinary differential equations.

Block methods with a set number of iterations, such as that of Rosser [432], can achieve a given order with fewer evaluations per step than for point methods and are thus attractive. The number of evaluations required may also be decreased by the retention of previously calculated inter-nodal values as in the method of Rosen [430].

The many possible forms of the multi-value methods of Gear [177], in particular those that update previously calculated values, may offer more efficient methods than the conventional multi-step forms. Stable methods of higher order than possible for multi-step methods are attainable in this class. Methods of this type may be developed from the existing multi-step methods by linear transformations, and yet may be more efficient in computation.

The local linearization of nonlinear systems about a predicted value, Tyson [521], rather than the value at the previous step, allows linear methods to be applied without limiting the resultant overall order for the nonlinear system to two. This procedure has received little attention but could widen considerably the scope of application of many methods applicable only to linear systems.

The use of a cyclic set of correctors, Donelson and Hansen [129], rather than a fixed corrector, over a series of time steps yields stable methods of higher order than possible with a fixed corrector and this should be pursued.

Methods using non-polynomial interpolates specialized for particular systems of differential equations should be considered. Here speed may quite possibly be gained from the loss of generality. In the same respect, preliminary transformations of the dependent variable adapted to the particular differential system should also be considered.

TABLE I

Explicit Runge–Kutta RKE(p,V)	EQ. IN APPENDIX -1	V TIME	RMS TE $\times 10^{-10}$ / ET = (RMS TE/2^{p+1}−2)(V/h) $\times 10^{x}$						
			h =						
			.001	.0005	.00025	.000125	.0000625	.00003125	.000015625
Classic(4,4)	A.26	4	X	1.578×10^{4} 4.206-4	1.686×10^{3} 8.99-5	1.05×10^{2} 1.10-5	6.325 1.349-6	2.683 1.145-6	0
Gill(4,4)	A.25	4	X	1.578×10^{4} 4.206-4	1.686×10^{3} 8.99-5	1.05×10^{2} 1.12-5	6.325 1.349-6	3.578 1.527-6	0
Ralston(4,4)	A.21	4	X	1.565×10^{4} 4.173-4	1.58×10^{3} 8.43-5	1.04×10^{2} 1.11-5	6.325 1.349-6	2.951 1.259-6	0
Nystrom(5,6)	A.34	6	X	1.006×10^{4} 1.95×10-4	3.95×10^{2} 1.53-5	2.24×10^{1} 1.73-6	0	-	-
Butcher(5,6) Newton-Cotes Quadrature	A.38	6	X	3.107×10^{3} 6.013-5	5.34×10^{1} 2.07-6	1.789 1.39-7	0	-	-
Lawson(5,6) Newton-Cotes Quadrature	A.42	6	2.973×10^{8} 2.876	4.49×10^{3} 8.689-5	1.18×10^{2} 4.58-6	3.578 2.77-7	0	-	-
Fehlberg(5,6)	A.49	6	X	4.64×10^{3} 8.98-5	1.03×10^{2} 4.0-6	3.130 2.423-7	0	-	-
Luther(5,6) Newton-Cotes Quadrature	A.35	6	X	3.65×10^{4} 7.05-4	9.04×10^{2} 3.50-5	2.683×10^{1} 2.08-6	0		
Luther(5,6) Gauss Quadrature	A.43	6		2.88×10^{4} 5.32-4	6.90×10^{2} 2.67-5	2.06×10^{1} 1.59-6	0		
Luther(5,6) Radau Quadrature	A.44	6		1.02×10^{3} 1.976-4	2.70×10^{2} 1.05-5	8.05 6.234-7	0		
Luther(5,6) Lobatto Quadrature	A.45	6		1.02×10^{3} 1.976-4	2.70×10^{2} 1.05-5	8.05 6.234-7	0	-	-
Shanks(5,5) N=5	A.47	5	9.898×10^{11} 7.982+3	1.057×10^{3} 1.64-4	2.72×10^{2} 8.78-6	1.297×10^{1} 8.37-7	0	-	-
Shanks(5,5) N=100	A.47	5	9.692×10^{11} 7.817+3	1.375×10^{4} 2.22-4	1.151×10^{3} 3.71-5	1.127×10^{2} 7.27-6	0	-	-
Modified Shanks(5,4) N=100	A.48	4	X	2.64×10^{4} 3.41-4	4.20×10^{3} 1.08-4	4.03×10^{2} 2.08-5	0	-	-
Butcher(6,7) Newton-Cotes Quadrature	A.57	7	X	1.046×10^{4} 1.16-4	1.19×10^{2} 2.63-6	1.789 7.947-8	1.265 1.124-7	1.789 3.18-7	0
Shanks(6,6)	A.67	6	1.233×10^{12} 5.872+3	2.52×10^{3} 2.4-5	3.54×10^{1} 6.75-7	.894 3.407-8	.6325 4.819-8	2.683 4.04-7	0
Sarafyan(7,10) Newton-Cotes Quadrature	A.68	10	4.575×10^{3} 1.80-5	3.64+1 2.87-7	.949 1.49-8	1.789 5.635-8	0	-	-
Shanks(7,7)	A.73	7	X	5.18×10^{2} 2.85-6	4.427 4.88-8	.894 1.971-8	0	-	-
Curtis(8,11)	A.75	11	3.108×10^{4} 6.704-5	9.41×10^{1} 4.06-7	.949 8.18-9	1.342 2.315-8	2.530 8.728-8	5.367 3.704-7	0
Shanks(8,10)	A.77	10	2.949×10^{4} 5.781-5	9.21×10^{1} 3.62-7	.949 7.44-9	1.789 2.806-8	2.530 7.931-8	5.367 3.368-7	0

The signed number following each entry indicates multiplication by the corresponding power of 10.

TABLE II

MULTISTEP METHOD	EQ. IN AP-PEN-DIX	V TIME	RMS TE 10^{-10} = / ET = (RMS TE/2^{p+1}-2)(V/h) 10^x / h = .0005	.00025	.000125	.0000625	.00003125
AB(3,3)	31	2	X	1.54×10^4	1.955×10^3	0	
AM(3,2)(PECE)	55			8.80-6	2.23-6		
AB(3,3)	31	2	X	6.75×10^3	1.185×10^3	0	
AM(4,3)(PECE)	56			1.80-6	6.32-7		
AB(3,3)	31	1	X	7.77×10^3	1.07×10^3	0	
AM(4,3)(PEC)	56			1.035-6	2.85-7		
CK(4,4)	4	2	X	-	4.24×10^2	40.6	0
AM(4,3)(PECE)	56				2.26-7	4.3-8	
Butcher Hybrid	1	3	X	1.71×10^3	57.2	0	
fifth order				3.30-7	2.22-8		
CK(4,4)	4	2	X	4.12×10^3	1.55×10^2	31.7	0
Wesson(5,5)(PECE)	73			5.32-7	4.0-8	1.5-8	
AB(4,4)	32	2	X	2.57×10^3	6.09×10^2	0	
AM(6,5)(PECE)	58			5.16-7	1.73-7		
Krogh(4,4)	25	2	X	2.61×10^3	3.46×10^2	0	
AM(6,5)(PECE)	58			5.23-7	9.85-8		
AB(4,4)	32	1	X	1.40×10^4	2.46×10^3	0	
AM(6,5)(PEC)	58			1.40-6	3.53-7		
Krogh(4,4)	25	1	X	1.40×10^4	2.46×10^3	0	
AM(6,5)(PEC)	58			1.40-6	3.53-7		
CK(4,4)	4	2	X	1.537×10^3	82.6	0	
AM(6,5)(PECE)	58			9.77-8	1.05-8		
CK(4,4)	4	1	X	4.81×10^3	3.13×10^2	0	
AM(6,5)(PEC)	58			1.55-7	1.99-8		
AB(8,8)	36	2	X	X	2.1×10^3	7.9×10^2	0
AM(8,7)(PECE)	60				6.58-8	4.97-8	
CK(4,4)	4	2	X	-	85.	22.8	0
AM(8,7)(PECE)	60				2.0-9	1.43-10	
CK(4,4)	4	2	X	-	96.5	24.	0
Wesson(8,8)(PECE)	74				3.3-9	1.50-10	

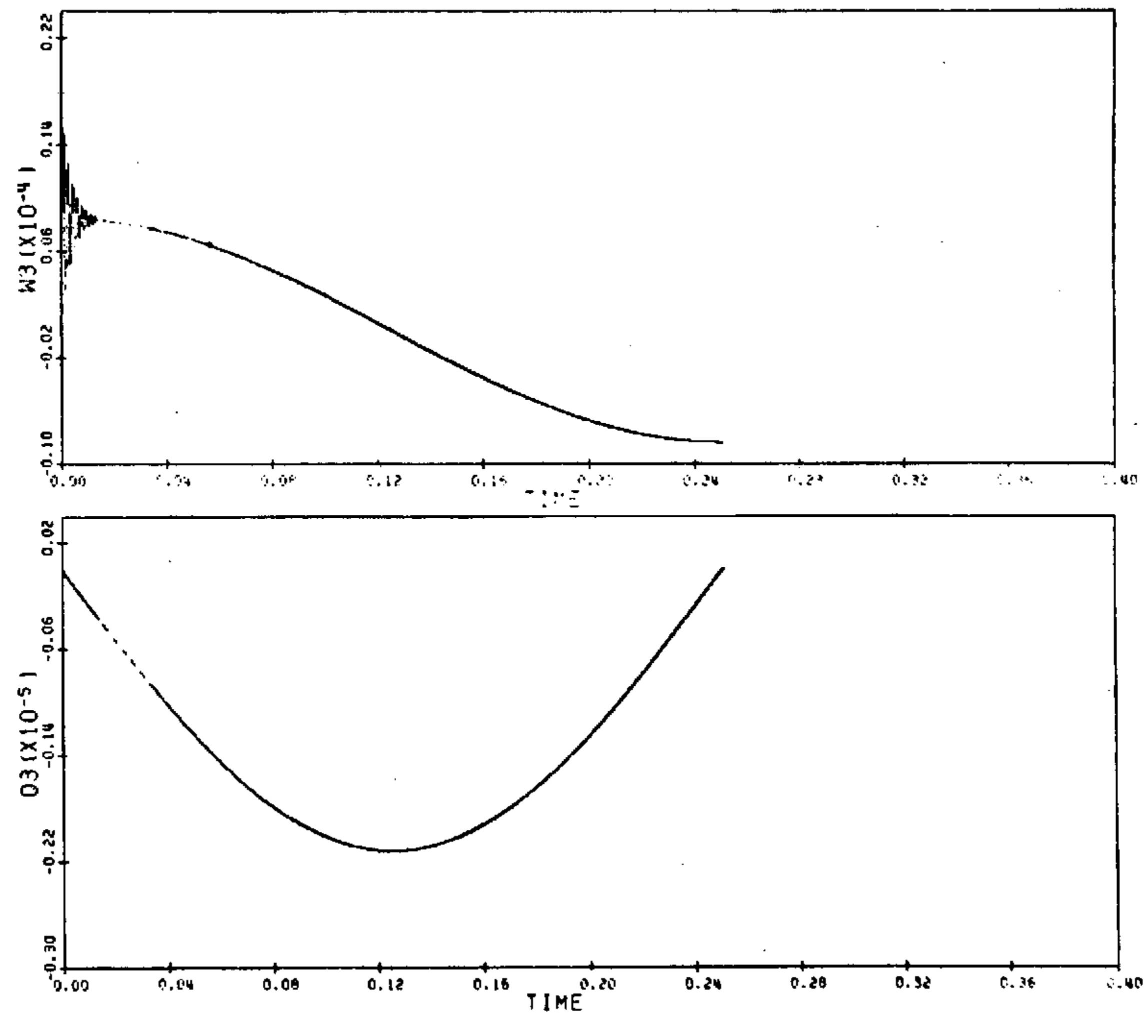


Figure 1. W_3 and q_3 up to T = .25 (quarter of an orbit).

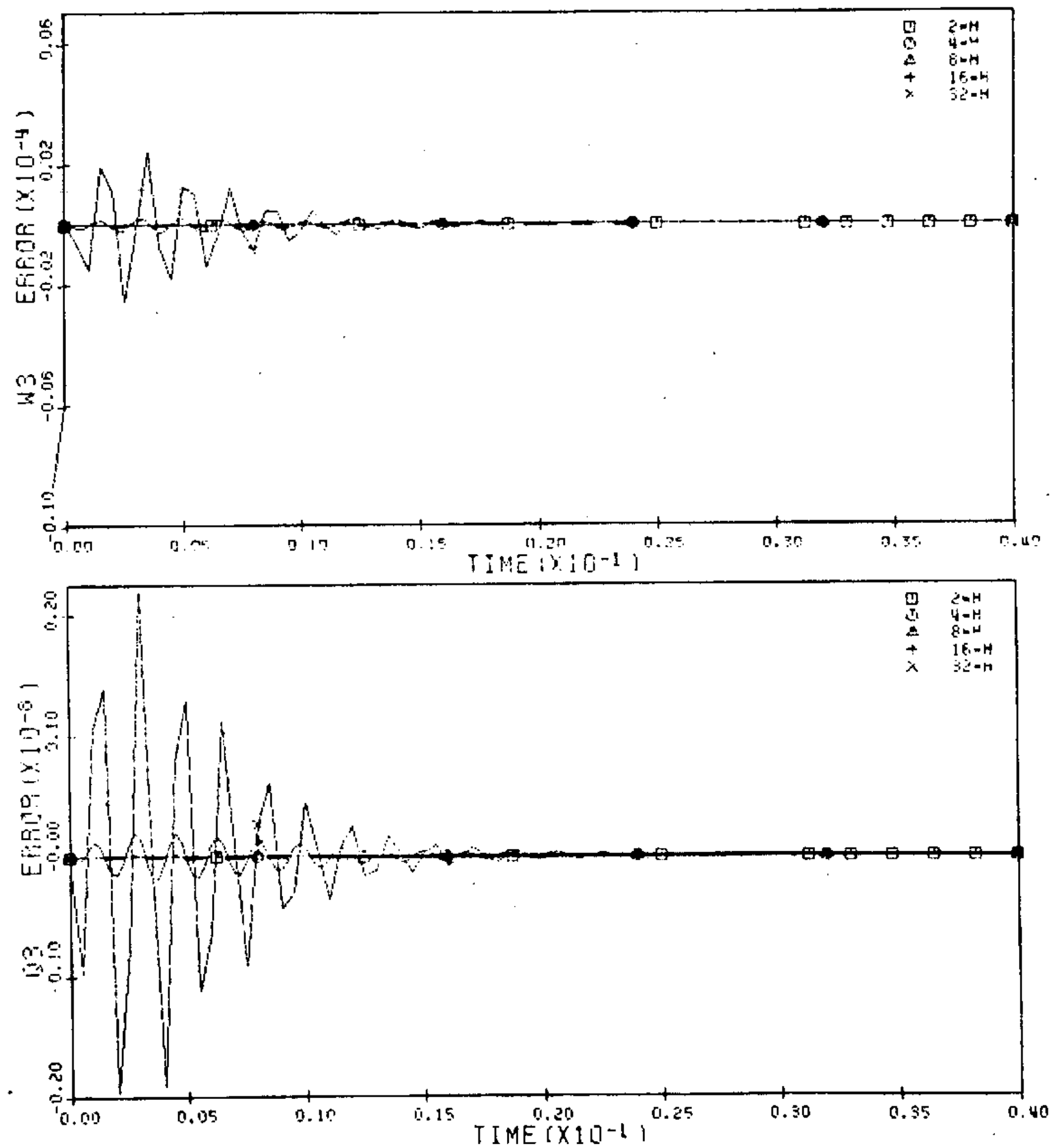
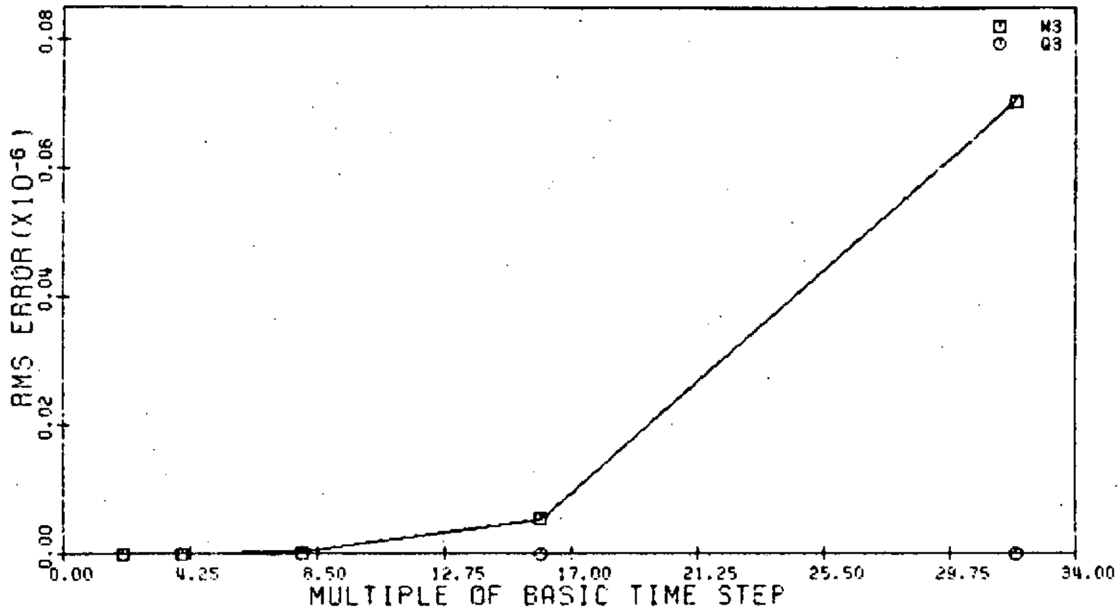


Figure 2. Errors in W_3 and q_3 versus time, up to T = .04.



RK-RALSTON(4,4) OPTIMISED FOR T.E. (CLAIMED TO BE BETTER THAN GILL(4,4))

COMPARISION FOR TRUNCATION ERROR (BASED ON SMALLEST
STEP-SIZE AS THE ACCURATE SOLUTION)

6 RUNS MADE FROM H=.000015625 TO H=.0005 , DOUBLING THE H EACH TIME
TILL THE METHOD GOES UNSTABLE. THE LAST RUN IS THE LAST STABLE H USABLE
COMPARISION RUN TO T=.04. THIS INTERVAL INCLUDES A TIME OF HIGH
OSCILLATION AND A TIME OF STRAIGHT DECAY. OSCILLATIONS STOP AT ABOUT
T=.02. T=.04 CORRESPONDS TO NSTEPS=2560 FOR H=.000015625

Figure 3. RMS error in W_3 versus step size.

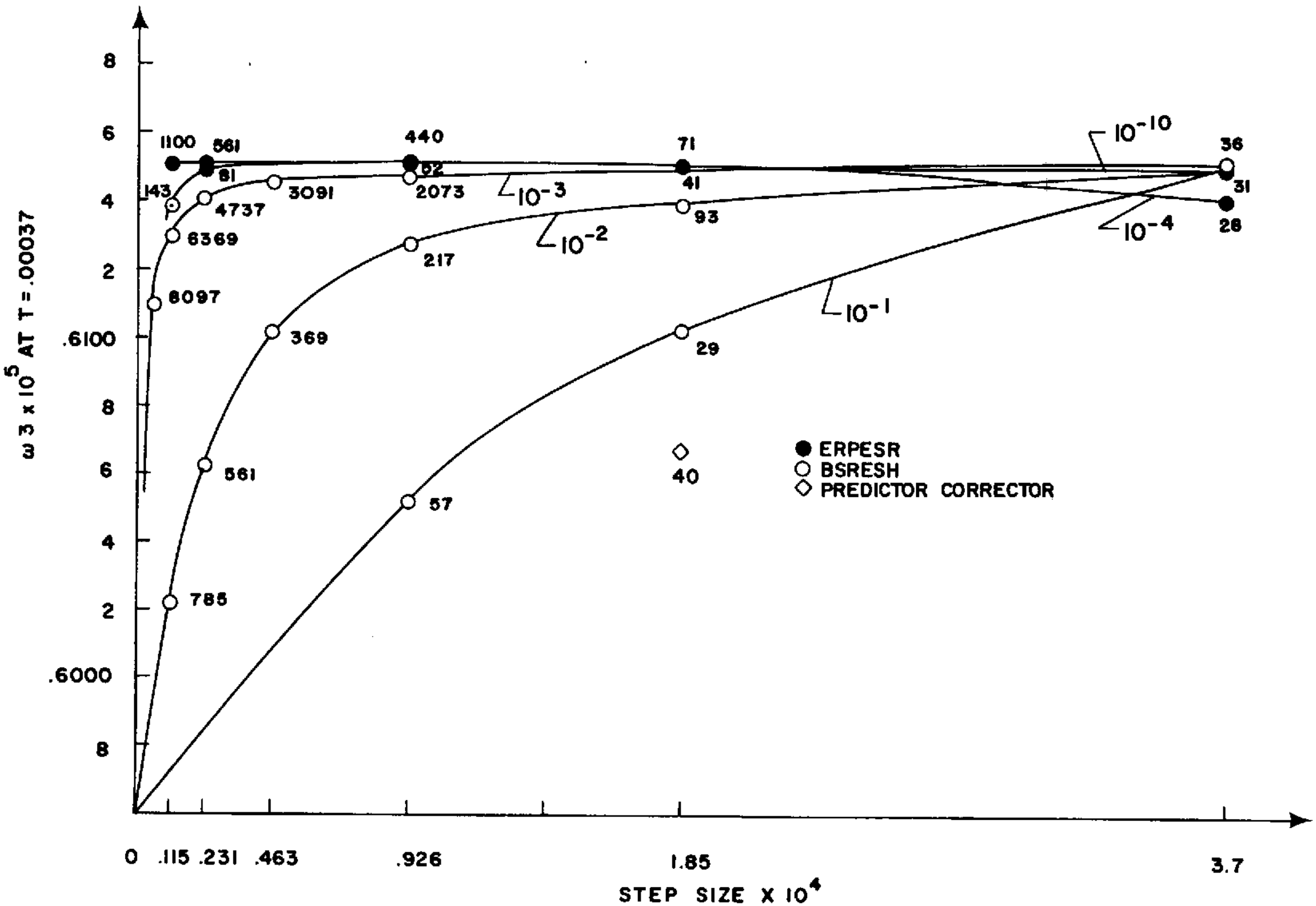


Figure 4. Comparison of ERPESR and BSRESH Extrapolation Methods - Time Step Effects.

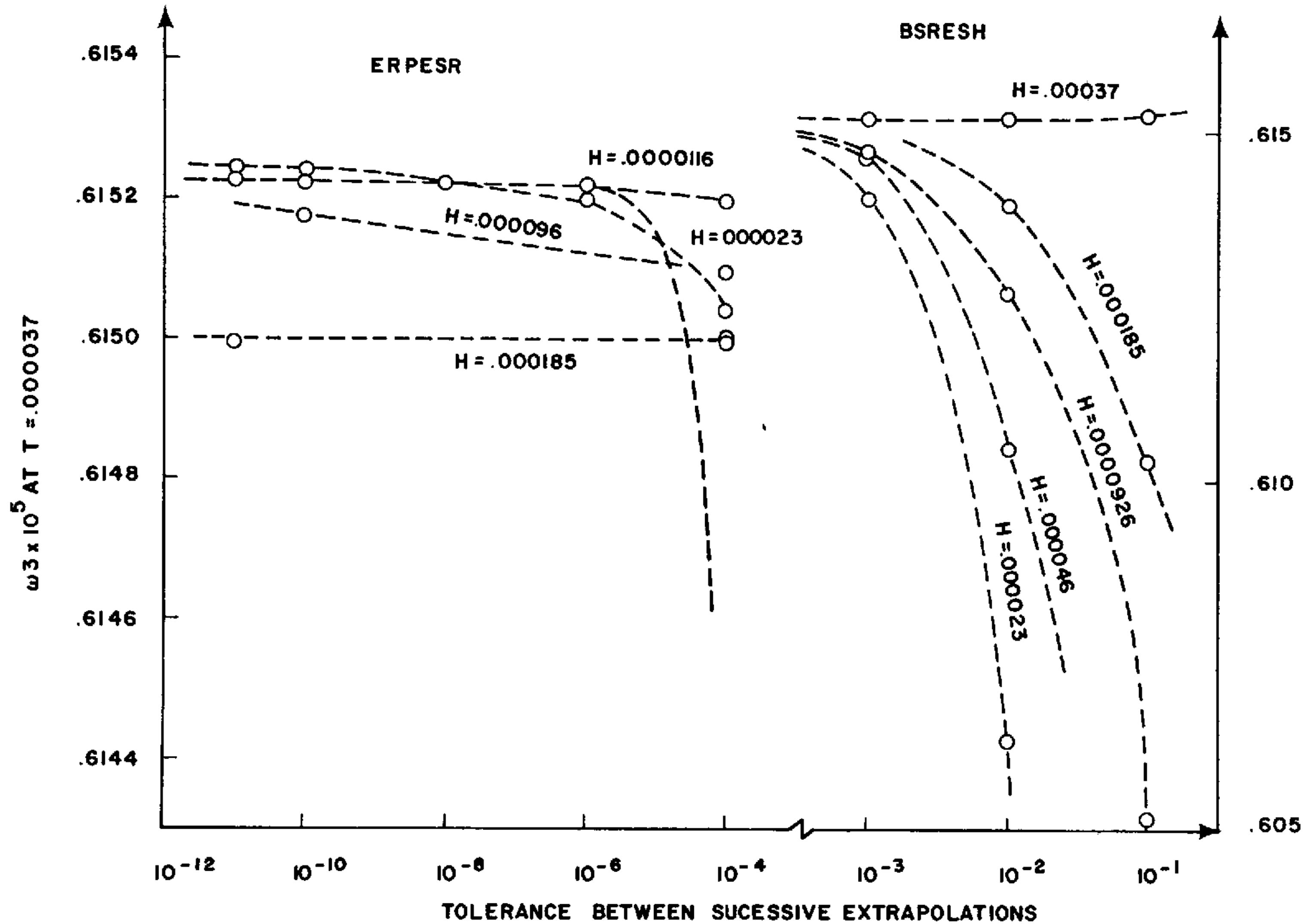


Figure 5. Comparison of ERPESR and BSRESH Extrapolation Methods - Tolerance Effects.

BIBLIOGRAPHY

1. Adrianova, L., "Comparison of Precision of Numerical Integration of Second Order Differential Equations with Their Corresponding Systems of First Order Differential Equations," Diff. Eq. 3, 788 (1967).

2. Allen, R., "A New Method of Solving Second-Order Differential Equations When the First Derivative Is Present," Comp. J. 8, 392 (1965-66).

3. Allen, R., "Numerically Stable Explicit Integration Techniques Using a Linearized Runge-Kutta Extension," Info. Sci. Rept. 39, Boeing Scientific Research Lab. (1969).

4. Alonzo, R., "A Starting Method for the Three-Point Adams Predictor-Corrector Method," J. Assoc. Comp. Mach. 7 (1960).

5. Anderson, N., Ball, R., & Voss, J., "A Numerical Method for Solving Control Differential Equations on Digital Computers," J. Assoc. Comp. Mach. 7, 61 (1960).

6. Anderson, W., "The Solution of Simultaneous Ordinary Differential Equations Using a General Purpose Digital Computer," Comm. ACM 3, 357 (1960).

7. Andrus, J., "Integration of Control Equations & the Problem of Small Time Constants," NASA Tech. Note D-3907 (1967).

8. Ansorge, R., "Zur Struktur gewisser Konvergenzkriterien bei der numerischen Losung von Anfangswertaufgaben," Num. Math. 6, 224 (1964).

9. Ansorge, R., "Konvergenz von Mehrschrittverfahren zur Losung Halblinearer Anfangswertaufgaben," Num. Math. 10, 209 (1967).

10. Ansorge, R., & Tornig, W., "Zur Stabilitat des Nystromschen Verfahrens," ZAMM 40, 568 (1960).

11. Antosiewicz, H. & Gautschi, W., "Numerical Methods in Ordinary Differential Equations," A Survey of Numerical Analysis, Editor, J. Todd, McGraw-Hill, New York 314 (1962).

12. Aronson, D., "The Stability of Finite Difference Approximation to Second Order Linear Parabolic Differential Equations," Duke Math. J. 30, 117 (1963).

13. Axelesson, O., "A Class of A-Stable Methods," BIT 9, 185 (1969).

14. Axelsson, O., "Global Integration of Differential Equations Through Lobatto Quadrature," BIT 4, 69 (1964).

15. Ballester, C. & Pereyrn, V., "On the Construction of Discrete Approximation to Linear Differential Expressions," Math. Comp. 21, 297 (1967).

16. Bard, A., Ceschino, F., Kuntzmann, J., Launrent, P., "Formules de Base de le Methode de Runge-Kutta," Chiffres 4, 31 (1961).

17. Baron, W., "Stabilitatsgebiete fur Predictor-Corrector-Vergahren," Master's Thesis TH Wien (1968).

18. Baxter, D., "The Digital Simulation of Transfer Functions," Nat. Res. Council of Canada Mech. Eng. Report MK-13 (1964).

19. Baxter, D., "Digital Simulation Using Approximate Methods," Nat. Res. Council of Canada Mech. Eng. Report MK-15 (1965).

20. Baxter, D., "The Step Response of Digital Simulation," Nat. Res. Council of Canada Mech. Eng. Report MK-14 (1965).

21. Bellman, R. & Richardson, J., "On Some Questions Arising in the Approximate Solution of Nonlinear Differential Equations," Quar. Appl. Math. 20, 333 (1963).

22. Benyon, P., "Comments on Giese's Paper, State Variable Difference Methods for Digital Simulation," Sim. 8, 270 (1967).

23. Benyon, P., "A Review of Numerical Methods for Digital Simulation," Sim. 11, 219 (1968).

24. Birkhoff, G., "Numerical Integration of Reactor Dynamics Equations," Symposium on the Numerical Solution of Nonlinear Differential Equations, Math. Res. Center, Univ. of Wisconsin, Madison (1966).

25. Bjurel, G., "Preliminary Report on Modified Linear Multistep Methods for a Class of Stiff Ordinary Differential Equations," Dept. of Info. Processing, The Royal Institute of Technology, Stockholm, Report #NA 69.02 (1969).

26. Bjurel, G., Dahlquist, B., Lindberg, B., Linde, S., & Oden, L., "Survey of Stiff Ordinary Differential Equations," Dept. of Information Processing, Royal Institute of Technology, Stockholm, Report #NA 70.11 (1970).

27. Blue, J., & Gummel, H., "Rational Approximations to Matrix Exponential for Systems of Stiff Differential Equations," J. Comp. Phy. 5, 70 (1970).

28. Blum, E., "A Formal System for Differentiation," J. Assoc. Comp. Mach. 13, 495 (1966).

29. Blum, E., "A Modification of the Runge-Kutta Fourth-Order Method," Math. Comp. 16, 176 (1962).

30. Boggs, P., "The Solution of Nonlinear Systems of Equations by A-Stable Integration Techniques," SIAM J. Num. Anal. 8, 767 (1971).

31. Borshch, Yu., "On a Matrix Algorithm for Solving Linear Differential Equations and sets of Such Equations," U.S.S.R. Comp. Math. & Math. Phy. 7, 231 (1967).

32. Bramble, J., "Error Estimates for Difference Methods in Forced Vibration Problems," SIAM J. Num. Anal. 3, 1 (1966).

33. Brandin, E., "The Mathematics of Continuous System Simulation," AFIPS Conference Proceedings, 33 (1966).

34. Braun, J. & Moore, R., "Solution of Differential Equations," Math. Res. Center, Report #901 (1968).

35. Brayton, R., "Necessary & Sufficient Conditions for Bounded Global Stability of Certain Nonlinear Systems," Quar. Appl. Math. 6, 237 (1971).

36. Brayton, R., Gustavson, F., & Liniger, W., "A Numerical Analysis of the Transient Behavior of a Transistor Circuit," IBM J. Res. & Dev. 10, 292 (1966).

37. Brayton, R., Gustavson, F., Hacktel, R., "A New Efficient Algorithm for Solving Differential Algebraic Systems Using Implicit Background Difference Formulas," Proc. IEEE 60, 98 (1972).

38. Brennan, R. & Linbarger, R., "A Survey of Digital Simulation; Digital Analog Simulator Programs," Sim. 3, 22 (1964).

39. Brock, P. & Murrary, F., "The Use of Exponential Sums in Step by Step Integration," MTAC 6, 63 (1952).

40. Brooks, J. & Pope, D., "Asymptotic Error Estimates & The Numerical Solution of the Equations of Orbital Motion," SIAM J. Num. Anal. 4, 446 (1967).

41. Brown, R., Riley, J., & Bennett, M., "Stability Properties of Adams-Moulton Type Methods," Math. Comp. 19, 90 (1965).

42. Brunner, H., "Marginal Stability & Stabilization in the Numerical Integration of Ordinary Differential Equations," Math. Comp. 24, 635 (1970).

43. Brunner, H., "Stabilization of Optimal Difference Operations," ZAMP 18, 438 (1967).

44. Brush, D., _et al._, "Solution of Ordinary Differential Equations Using Two 'Off-Step' Points," _J. Assoc. Comp. Mach._ 14, 769 (1967).

45. Bryne, G. & Chi, D., "Linear Multistep Based on G-Splines," _SIAM J. Num. Anal._ 9, 316 (1972).

46. Bulirsch, R. & Stoer, J., "Fehlerabschatzungen und Extrapolation mit Rationalen Funktionen bei Verfahren von Richardson-Typus," _Num. Math._ 6, 413 (1964).

47. Bulirsch, R. & Stoer, J., "Numerical Treatment of Ordinary Differential Equations by Extrapolation Methods," _Num. Math._ 8, 1 (1966).

48. Bulirsch, R. & Stoer, J., "Asympotic Upper and Lower Bounds for Results of Extrapolation Methods," _Num. Math._ 8, 93 (1966).

49. Butcher, J., "An Algebraic Theory of Integration Methods," _Math. Comp._ 26, 79 (1972).

50. Butcher, J., "A Convergence Criterion for a Class of Integration Methods," _Math. Comp._ 26, 107 (1972).

51. Butcher, J., "A Multistep Generalization of Runge-Kutta Methods with Four or Five Stages," _J. Assoc. Comp. Mach._ 14, 84 (1967).

52. Butcher, J., "A Modified Multistep Method for the Numerical Integration of Ordinary Differential Equations," _J. Assoc. Comp. Mach._ 12, 124 (1965).

53. Butcher, J., "Coefficients for the Study of Runge-Kutta Integration Processes," _J. Aus. Math. Soc._ 3, 185 (1963).

54. Butcher, J., "Implicit Runge-Kutta Processes," _Math. Comp._ 18, 50 (1964).

55. Butcher, J., "Integration Processes Based on Radau Quadrature Formulas," _Math. Comp._ 18, 233 (1964).

56. Butcher, J., "On Attainable Order of Runge-Kutta Methods," _Math. Comp._ 19, 408 (1965).

57. Butcher, J., "On the Convergence of Numerical Solutions to Ordinary Differential Equations," _Math. Comp._ 20, 1 (1966).

58. Butcher, J., "On the Integration Processes of A. Huta," _J. Aus. Math. Soc._ 3, 203 (1963).

59. Butcher, J., "On Runge-Kutta Processes of High Order," _J. Aus. Math. Soc._ 4, 179 (1964).

60. Butcher, J., "The Effective Order of Runge-Kutta Methods," Conference on the Numerical Solutions of Differential Equations, Lecture Notes in Math. #109, Springer-Verlag, Berlin, 133 (1969).

61. Butcher, J., "On the Convergence of Numerical Solutions to Ordinary Differential Equations," Math. Comp. 20, 1 (1966).

62. Byrne, G. & Lambert, R., "Pseudo-Runge-Kutta Methods Involving Two Points," J. Assoc. Comp. Mach. 13, 114 (1966).

63. Byrne, G., "Pseudo-Runge-Kutta Involving Two Points," Ph.D. Thesis, Iowa State University (1963).

64. Byrne, G., "Parameters for Pseudo-Runge-Kutta Methods," Comm. ACM 10, 102 (1967).

65. Calahan, D., "Efficient Numerical Analysis of Nonlinear Circuits," Proc. 6th Annual Allerton Conference on Circuit & System Theory, Univ. of Illinois, Urbana 321 (1968).

66. Calahan, D. & Gear, C., "An Ill-Conditioning Problem with Implicit Integration," Proc. IEEE 57, 1775 (1969).

67. Calahan, D., "Numerical Consideration in the Transient Analysis and Optimal Design of Nonlinear Circuits," Digest Record of Joint Conference on Mathematical and Computer Aids to Design, ACM/SIAM/IEEE, Anaheim, California 129 (1969).

68. Calahan, D., "Numerical Solutions of Linear Systems With Widely Separated Time Constants," Proc. IEEE 55, 2016 (1967).

69. Calahan, D. & Abbott, N., "Stability Analysis of Numerical Integration," Proc. 10th Midwest Symp. on Circuit Theory, Purdue Univ., Lafayette, Indiana (1967).

70. Calahan, D., "A Stable Accurate Method of Numerical Integration for Nonlinear Systems," Proc. IEEE 56, 744 (1968).

71. Callender, E., "Single Step Methods and Low Order Splines for Solutions of Ordinary Differential Equations, SIAM J. Num. Anal. 8, 61 (1971).

72. Carr, J., "Error Bounds for the Runge-Kutta Single-Step Integration Process," J. Assoc. Comp. Mach. 5, 39 (1958).

73. Case, J., "A Note on the Stability of Predictor-Corrector Techniques," Math. Comp. 23, 741 (1969).

74. Casity, C., "Solutions of the Fifth Order Runge-Kutta Equations," SIAM J. Num. Anal. 3, 598 (1966).

75. Cassity, C., "The Complete Solution of the Fifth Order Runge-Kutta Equations," SIAM J. Num. Anal. 6, 432 (1969).

76. Certaine, J., "The Solution of Ordinary Differential Equations with Large Time Constants," Mathematical Methods for Digital Computers, Ralston & Wilf, ed., Wiley, New York 128 (1960).

77. Ceschino, F., "Modifications de la Longueur du pas dan L'ingration Numerique par les Methods a pas Lies," Chiffres 2, 101 (1961).

78. Ceschino, F., "Une Methode de Mise en Oeucre des Formules d'Obrechkoff pour L'integration des Equations Differentielles," Chiffres 2, 49 (1961B).

79. Chai, A., "Error Estimate of a Fourth-Order Runge-Kutta Method with Only One Initial Derivative Evaluation," AFIPS Conf. Proc. 32, 467 (1968).

80. Chai, A., "A Modified Runge-Kutta Method," Sim. 10, 221 (1968).

81. Chi, D., "Linear Multistep Based on G-Splines," Doctorial Thesis University of Pittsburgh (1970).

82. Chase, P., "Stability Properties of Predictor-Corrector Methods for Ordinary Differential Equations," J. Assoc. Comp. Mach. 9, 457 (1962).

83. Christiansen, J., "Numerical Solution of Ordinary Simultaneous Differential Equations of the First Order Using a Method for Automatic Step Change," Num. Math. 14, 317 (1970).

84. Clancey, J. & Fineberg, M.,"Digital Simulation Languages, A Critique and a Guide," AFIPS Conf. Proc. 27 (1965).

85. Clark, N., "Program Description for Library Subroutine ANL D250 DIFSUB," Argonne National Lab., Argonne, Ill. (1966).

86. Clark, N., "A Study of Some Numerical Methods for the Integration of Systems of First-Order Ordinary Differential Equations," Argonne National Lab. Report, ANL-7428 (1968).

87. Clenshaw, C. & Curtiss, A., "A Method for Numerical Integration on an Automatic Computer," Num. Math. 2, 197 (1960).

88. Clenshaw, C., "The Numerical Solution of Linear Differential Equations in Chebyshev Series," PICC Symposium, Rome, on Differential & Integral Equations, Birkhauser, Basel 222 (1960).

89. Cohen, C. & Hubbard, E., "An Algorithm Applicable to Numerical Integration of Orbits in Multiple Revolution Steps," Astron. J. 65, 454 (1960).

90. Cooke, C., "On Stiffly Stable Implicit Linear Multistep Methods," SIAM J. Num. Anal. 9, 29 (1972).

91. Cooper, G., "A Class of Single-Step Methods for Systems of Non-linear Differential Equations," Math. Comp. 21, 597 (1967).

92. Cooper, G., "Error Bounds for Numerical Solutions of Ordinary Differential Equations," Num. Math. 18, 162 (1971).

93. Cooper, G., "Interpolation & Quadrature Methods for Ordinary Differential Equations," Math. Comp. 22, 69 (1968).

94. Cooper, G. & Gal, E., "Single-Step Methods for Linear Differential Equations," Num. Math. 10, 307 (1967).

95. Crane, R. & Klopfenstein, R., "A Predictor-Corrector Algorithm with an Increased Range of Absolute Stability," J. Assoc. Comp. Mach. 12, 227 (1965).

96. Crane, R. & Lambert, R., "Stability of a Generalized Corrector Formula," J. Assoc. Comp. Mach. 9, 104 (1962).

97. Cullum, J., "Numerical Differentiation and Regularization," SIAM J. Num. Anal. 8, 254 (1971).

98. Curtis, A., "An Eight Order Runge-Kutta Process with Eleven Function Evaluations per Step," Num. Math. 16, 268 (1970).

99. Curtiss, C. & Hirachfelder, J., "Integration of Stiff Equations," Proc. Nat. Acad. Sci. U.S.A. 38, 235 (1952).

100. Dahlquist, G., "Convergence and Stability in the Numerical Integration of Ordinary Differential Equations," Math. Scand. 4, 33 (1956).

101. Dahlquist, G., "A Numerical Method for Some Ordinary Differential Equations with Large Lipschitz Constants," Proc. IFIP Congress, Edinburgh (1968).

102. Dahlquist, G., "On Rigorous Error Bounds in the Numerical Solution of Ordinary Differential Equations," The Numerical Solution of Nonlinear Differential Equations, ed. D. Greenspan, Wiley, New York 89 (1966).

103. Dahlquist, G., "Stability and Error Bounds in the Numerical Integration of Ordinary Differential Equations," Kungl. Tekniska Hogskolans Handlingar 130 (1959).

104. Dahlquist, G., "Stability Questions for Some Numerical Methods for Ordinary Differential Equations," Experimental Arithmetic High Speed Computing & Mathematics, Proc. of Symposium on Applied Math. XV, Amer. Math. Soc., Providence (1963).

105. Dahlquist, G., "Stability Questions for Some Numerical Methods for Ordinary Differential Equations," AMA Proc., Symposia Appl. Math. 15, 147 (1963).

106. Dahlquist, G., "A Special Stability Problem for Linear Multistep Methods," BIT 3, 27 (1963).

107. Danchick, R., "Further Results on Generalized Predictor-Corrector Methods," J. Comp. & Sys. Sci. 2, 203 (1968).

108. Daniel, J., Pereyra, W. & Shumaker, L., "Iterated Deferred Corrections for Initial Value Problems," Math. Res. Cntr. Report # 808 (1967).

109. Daniel, J., "Nonlinear Equations Arising in Deferred Correction of Initial Value Problems," Math. Res. Cntr. Report #818 (1967).

110. Davison, E., "A High Order Crank-Nicjolson Technique for Solving Differential Equations," Comp. J. 10, 195 (1967).

111. Davison, E., "The Numerical Solution of Large Systems of Linear Differential Equations," AIChE J. 14, 46 (1968).

112. Day, J., "A One-Step Method for the Numerical Integration of the Differential Equations $y'' = f(x)\,y + g(x)$," Comp. J. 7, 314 (1965).

113. Day, J., "A One-Step Method for the Numerical Solution of Second Order Linear Ordinary Differential Equations," Math. Comp. 18, 664 (1964).

114. Day, J., "A Runge-Lutta Method for the Numerical Integration for the Differential Equation $y'' = f(x,y)$," ZAMM 5, 354 (1965).

115. Day, J., "A Runge-Kutta Method for the Numerical Solution of the Goursat Problem in Hyperbolic Partial Differential Equations," Comp. J. 9, 81 (1966).

116. Day, J., "Quadrature Methods for Arbitrary Order for Solving Linear Ordinary Differential Equations," BIT 6, 181 (1966).

117. Decell, H., et al., "Concerning the Numerical Solution of Differential Equations," Math. of Comp. 20, 431 (1966).

118. Dejon, B., "Numerical Stability of Difference Equations with Matrix Coefficients," SIAM J. Num. Anal. 4, 119 (1967).

119. Dejon, B., "Stronger Than Uniform Convergence of Multistep Difference Methods," Num. Math. 8, 29 (1966).

120. Dejon, B., "Addendum to Stronger Than Uniform Convergence Multistep Difference Methods," Num. Math. 9, 268 (1966).

121. Dennis, S., "The Numerical Integration of Ordinary Differential Equations Possesing Exponential Type Solutions," Proc. Cambridge Philos. Soc. 56, 240 (1960).

122. Dennis, S., "Step By Step Integration of Ordinary Differential Equations," Applied Math. Quart. 20, 359 (1962).

123. Descloux, J., "A Note on a Paper by A. Nordsieck," Dept. of Comp. Sci. Report 131, Ill. (1963).

124. Devyotko, V., "On a 2-Sided Approximation for the Numerical Integration of Ordinary Differential Equations," U.S.S.R. Comp. Math. & Math. Phys. 3, 336 (1963).

125. Dill, C., & Gear, C., "A Graphical Search for Stiffly Stable Methods for Ordinary Differential Equations," J. Assoc. Comp. Mach. 18, 75 (1971).

126. Dill, C., "A Computer Graphic Technique for Finding Numerical Methods for Ordinary Differential Equations," Dept. of Computer Sci. Rept. 295, University of Illinois (1969).

127. Distefano, G., "Stability of Numerical Integration Techniques," AIChE J. 11, 946 (1968).

128. Dolgopolova, T., & Ivanov, V., "On Numerical Differentation," U.S.S.R. Comp. Math. & Math. Phys. 4, 174 (1966).

129. Donelson, J., & Hansen, E., "Cyclic Composite Multistep Predoctor-Corrector Methods," SIAM J. Num. Anal. 8, 137 (1971).

130. Durham, H., et al., "Study of Methods for Numerical Solution of Ordinary Differential Equations," NASA CR 57430 (1964).

131. Dyer, J., "Generalized Multistep Methods in Satellite Orbit Computation," J. Assoc. Comp. Mach. 15, 712 (1968).

132. Ehle, B., "High Order A-Stable Methods for the Numerical Solution of Systems of Differential Equations," BIT 8, 276 (1968).

133. Ehle, B., "On Pade Approximations to the Exponential Functions and A-Stable Methods for the Numerical Solution of Initial Value Problems," Res. Rept. CSRR 2010, University of Waterloo (1969).

134. Eisenpress, H., & Bomberault, A., "Efficient Symbolic Differentiation Using PL/1-FORMAC," IBM New York Scientific Center Report 320-29561 (1968).

135. El-Sherif, H., "Implicit Implementation of the Weighted Backward Euler Formula," IBM J. Res. Develop. 6, 336 (1968).

136. El-Sherif, H., "Solution of Electromechanical Systems Using A Weighted Backward Euler Formula," Proc. First Annual ASILOMAR Conference on Circuits & Systems, 11 (1967).

137. Emanuel, G., "The Wilf Stability Criterion for Numerical Integration," J. Assoc. Comp. Mach. 10, 557 (1963).

138. Emanuel, G., "Numerical Analysis of Stiff Equations," SSD-TDR-63-380, Aerospace Corp., El Segundo, California (1964).

139. Emanuel, G., "Problems Underlying the Numerical Integration for the Chemical & Vibrational Rate Equations in a Near-Equilibrium Flow," AEDC-TDR-63-82, Aerospace Corp., El Segundo, California (1963).

140. Engeli, M., "Achievements & Problems in Formula Manipulation," Information Processing 68, ed. A. J. H. Morrell, North Holland Publishing Co., Amsterdam 24 (1969).

141. England, R., "Error Estimates for Runge-Kutta Type Solutions to Systems Of Ordinary Differential Equations," Comp. J. 12, 166 (1969).

142. England, R., "Automatic Methods for Solving Systems of Ordinary Differential Equations," Thesis, University of Liverpool (1967).

143. Falbi, P. & Groome, G., "Stability of Difference Approximations to Differential Equations," J. Diff. Eq. 13, 48 (1973).

144. Fedenko, N., "A Remark on the Instability of Certain Numerical Solution Schemes for the Cauchy Problem," Instute of Math. AS BSST 11, 172 (1964).

145. Fehlberg, E., "Classical Fifth-, Sixth-, Seventh-, & Eight-Order Runge-Kutta Formulas With Stepsize Control," NASA Technical Report, NASA TR R-287 (1968).

146. Fehlberg, E., "Eine Methode zur Fehlerverkleinerung beim Runge-Kutta Verfahren," ZAMM 38, 421 (1958).

147. Fehleberg, E., "Klassische Runge-Kutta-Formeln vierter und niedreigerer Ordung Mit Schrittwein-Kontrolle und ihre Anwendung auf Warmeleitungsprobleme," Computing (Arch. Elektron, Rechnen), 6, 61 (1970).

148. Fehlberg, E., "Low-Order Classical Runge-Kutta Formulas With Stepsize Control," NASA Technical Report, M-256 (1969).

149. Fehlberg, E., "Neue Genauere Runge-Kutta Formeln fur Differentialgleichungen n-ter Ordung," ZAMM 40, 449 (1960).

150. Fehlberg, E., "New High-Order Runge-Kutta Formulas with Step Size Control for Systems of First- and Second-Order Differential Equations," Presented by S. Filippi at the Meeting of the GAMM in Giessen, Germany April (1964).

151. Fehlberg, E., "New High-Order Runge-Kutta Formulas with an Arbitrarily Small Truncation Error," ZAMM 45 (1965).

152. Fehlberg, E., "Numerically Stable Interpolation Formulas With Favorable Error Propagation for First & Second Order Differential Equations," NASA TN D-599 (1961).

153. Fehlberg, E., "Runge-Kutta Type Formulas of High-Order Accuracy & Their Application to the Numerical Integration of the Restricted Problem of Three Bodies," Presented at the Colloque Interantianal des Techniques de Calcul Analogique et Numerique in Aeronaitique in Liege, Belgium, Sept. (1963).

154. Feldstein, M., & Stetter, H., "Simplified Predictor-Corrector Methods," Assoc. Comput. Mach. National Conference (1963).

155. Ferguson, R., & Orlow, T., "FNOL 3, A Computer Program to Solve Ordinary Differential Equations," Naval Ordance Lab, White Oak, Silver Spring, Maryland, March (1971).

156. Filippi, S., "Contribution To The Implicit Runge-Kutta Process," Elektronische Datenverarbeitung 10, 113 (1968).

157. Filippi, S., & Glasmacher, W., "New Results On Applying The Runge-Kutta Processes With The Aid of Nonnumerical Programs," Elektronische Datenverarbeitung 10, 16 (1968).

158. Forrington, C., "Extensions of the Predictor-Corrector Method for the Solution of Systems of Ordinary Differential Equations." Comp. J. 4, 80 (1961-1962).

159. Forrington, C., "The Numerical Solution of System of Differential Equations Using a Different Step Length for Each Equation," Royal Aircraft Estab. Tech. Note Mem. Spare 69 (1960).

160. Fowler, M., "A New Numerical Method for Simulation," Sim. 5, 324 (1965).

161. Fowler, M., & Warten, R., "A Numerical Integration Technique for Ordinary Differential Equations with Widely Separated Eigenvalues," IBM J. Res. Dev. 11, 537 (1967).

162. Frey, T., "Solution of Equations By Stepwise Perturbation," Studia Scientiarum Mathematicarum Nungarica, 3, 93 (1968).

163. Froese, C., "An Evaluation of Runge-Kutta Type Methods for Higher Order Differential Equations," J. Assoc. Comp. Mach. 8, 637 (1961).

164. Fryer, W. & Schultz, W., "A Survey of Methods for Digital Simulation of Control Systems." CAL Report No. ZA-1681-E-1 (1964).

165. Fyfe, D., "Economical Evaluation of Runge-Kutta Formula," Math. Comp. 20, 392 (1966).

166. Gabel, G., "Predictor-Corrector Methods Using Divided Differences," Master's Thesis, University of Tornoto (1968).

167. Gaskill, R., Harris, J., & Knight, A., "DAS--A Digital Analog Simulator," Proc. of 1963 Spring Joint Computer Conference AFIPS 23, 83 (1963).

168. Gaskill, R., "Fact & Fallacy in Digital Simulation," Sim. 5, 309 (1965).

169. Gates, L., "Numerical Solution of Differential Equations by Repeated Quadratures," SIAM Review 6, 134 (1964).

170. Gautschi, W., "Numerical Integration of Ordinary Differential Equations Based on Trigonometric Polynominals," Num. Math. 3, 381 (1961).

171. Gear, C., "The Automatic Integration of Large Systems of Ordinary Differential Equations," Digest Record of 1969 Joint Conference on Mathematical & Computational Aids to Design, Anaheim, CA, 27 (1969).

172. Gear, C., "The Automatic Integration of Stiff Ordinary Differential Equations," Information Processing 68, ed. A. J. H. Morrel, North Holland, Amsterdam, 187 (1969).

173. Gear, C., "The Automatic Integration of Ordinary Differential Equations," Comm. ACM 14, 176 (1971).

174. Gear, C., "The Numerical Integration of Ordinary Differential Equations," Math. Comp. 21, 146 (1967).

175. Gear, C., "Hybrids Methods for Initial Value Problems in Ordinary Differential Equations," SIAM J. Num. Anal. 2, 69 (1965).

176. Gear, C., "Numerical Integration of Stiff Ordinary Differential Equations," Report #221, University of Illinois, Dept. of Computer Science (1967).

177. Gear, C., "The Numerical Integration of Ordinary Differential Equations," Math. Comp. 21, 146 (1967).

178. Gear, C., "The Numerical Integration of Ordinary Differential Equations of Various Orders," Argonne National Lab. Report #ANL 7126 (1966).

179. Gear, C., "Rational Approximations by Implicit Runge-Kutta Schemes," BIT 10, 20 (1970).

180. Gear, C., "The Simulaneous Numerical Solution of Differential Algebraic Equations," IEEE CT-18 (1967).

181. Gekeler, E., "On the Solution of System of Equations by the Epsilon Algorithm of Wynn," Math. Comp. 118, 427 (1972).

182. Gelina, R., "Stiff Systems of Kinetic Equations--A Practitioner'a View," J. Comp. Phys. 9, 222 (1972).

183. Giese, C., "State Variable Difference Methods for Digital Simulation," Sim. 8, 263 (1967).

184. Gibbons, A., "A Program for the Automatic Integration of Differential Equations Using the Method of Taylor-Series," Comp. J. 3, 108 (1960).

185. Gilbert, E., et al., "Discussion Following Fowler's Paper, A New Numerical Method for Simulation," Sim. 6, 90 (1966).

186. Gilbert, E., "Dynamic-Error Analysis of Digital & Combined Analog-Digital Computer Systems," Sim. 6, 24 (1966).

187. Gilbert, E., "Some Critical Remarks on a New Numerical Method of Simulation of Dynamical Systems," Sim. 6, 90 (1966).

188. Gill, S., "A Process for the Step-By-Step Integration of Differential Equations in an Automatic Digital Computing Machine," Proc. of the Cambridge Philosophical Society 47, 96 (1951).

189. Giloi, W. & Grebe, H., "Construction of Multistep Integration Formulas for Simulation Purposes," IEEE Trans. on Computers C-17, 1121 (1968).

190. Godunov, S. & Prokopov, G., "The Solution of Differential Equations by the Use of Circualinear Difference Networks," U.S.S.R. Comp. Math. & Math. Phys. 8 (1968).

191. Goncharova, I. & Martynov, A., "On A Practical Method for the Automatic Selection of Scaling Factors in the Solution of Systems of Ordinary Differential Equations," U.S.S.R. Comp. Math. & Math. Phys. 2, 1074 (1962).

192. Gorbunov, A. & Shakhov, Y., "On the Approximate Solution of Cauchy's of Correct Figures," U.S.S.R. Comp. Math. & Math. Phys. 3, 239 (1963).

193. Gordon, P., "Nonsymmetric Differences Equation," SIAM J. 13, 667 (1965).

194. Gourlay, A., "A Note on Trapezoidal Methods for the Solution of Intial Value Problems," Math. Comp. 24, 629 (1970).

195. Gragg, W. & Stetter, H., "Generalized Multistep Predictor-Corrector Method," J. Assoc. Comp. Mach. 11, 188 (1964).

196. Gragg, W., "On Extrapolation Algorithms for Ordinary Initial Value Problems," SIAM J. Num. Anal. 2, 384 (1965).

197. Gragg, W., "Repeated Extrapolation to the Limit in the Numerical Solution of Ordinary Differential Equations," Ph.D. Thesis, UCLA (1963).

198. Gragg, W., "Two Algorithms of Romberg Type for Ordinary Initial Value Problems," Information Processing 65, ed. W. A. Kalemich, MacMillian, London, 564 (1965).

199. Gray, H., "Digital Computer Solution of Differential Equations," Proc. of the Western Joint Computer Conference AFIPS 13, 87 (1958).

200. Gray, H., "Numerical Methods in Digital Real-Time Simulation," Quart. App. Math. XII, 133 (1954).

201. Gray, H., "Propagation of Truncation Errors in the Numerical Solutions of Ordinary Differential Equations by Repeated Closures," J. Assoc. Comp. Mach. 2, 5 (1955).

202. Greenspan, H., et al., "On Varying Stepsize in Numerical Integration of First Order Differential Equations," Num. Math. 7, 286 (1965).

203. Greenspan, D., "Numerical Studies of the 3-Body Problem," SIAM J. Appl. Math. 20, 67 (1971).

204. Greenspan, D., "Approximate Solution of Initial Value Problems for Ordinary Differential Equations by Boundary Value Techniques," Math. Res. Cntr. Report #752 (1967).

205. Grigorieff, R., "Uber die Koerzitivitat Gewohnlicher Differenzenoperatoren und die Konvergenz von Mehrschrittverfahren," Num. Math. 15, 196 (1970).

206. Grigorieff, R., "Uber die Knovergenz von Differenzenverfahren die Linearen geivohnlichen Differentialgleichungen," ZAMM 48, T72 (1968).

207. Grobner, W., Kuhnert, K., Reitberger, H., & Wanner, G., "Development of New Methods for the Solution of Nonlinear Differential Equations by the Method of Lie Series Extension to New Fields," Final Technical Report, European Research Office (1970).

208. Grobner, W., "Die Lie-Reihen und Ihre Anwendungen," Deutscher Verlag der Wissenschaften, Berlin (1960).

209. Grobner, W., Fuchs, F., Hairer, E., Kuhnert, K., Kostlunger, K., Mitter, P., & Wanner, G., "The Solution of Differential Equations by the Method of Lie Series & Its Generalization," Final Report, European Research Office, Contract DA JA37-71-C-2956, University of Innsbruck (1972).

210. Grosswald, E., "Transformations Useful in Numerical Integration Methods," J. Soc. Indust. & Appl. Math. 7, 76 (1959).

211. Gruttke, W., "Pseudo-Runge-Kutta Methods of the Fifth Order," J. Assoc. Comp. Mach. 17, 613 (1970).

212. Guderley, K. & Hsu, C., "A Predictor-Corrector Method for a Certain Class of Stiff Differential Equations," Math. Comp. 26, 51 (1972).

213. Gudovich, N., "On Formulae for Numerical Differentiation," U.S.S.R. Comp. Math. & Math. Phys. 6, 207 (1966).

214. Gurk, H. & Rubinoff, M., "Numerical Solution of Differential Equations," Proc. of the Eastern Joint Computer Conference, AFIPS 6, 58 (1954).

215. Gurk, H., "The Use of Stability Charts in the Synthesis of Numerical Quadrature Formulae," Quart. Appl. Math. XIII, 73 (1955).

216. Haber, S., "Numerical Evaluation of Multistep Integrals," SIAM Review 12, 481 (1970).

217. Hafner, P., "Stability Charts of Various Numerical Methods for Solving Systems of Ordinary Differential Equations," Weapons Research Establishment Technical Note WSD 112, Salisbury, South Australia (1969).

218. Hain, K. & Hertweck, F., "Numerical Integration of Ordinary Differential Equations by Difference Methods with Automatic Determination of Steplength," PICC Symposium, Rome, Differential & Integral Equations, Birkhauser, Basel (1960).

219. Haines, C., "Implicit Processes for the Numerical Integration of Systems of First Order Differential Equations," M. Sc. Thesis, University of Manchester Institute of Science & Technology (1968).

220. Haines, C., "Implicit Integration Processes With Error Estimate for the Numerical Solution of Differential Equations," Comp. J. 12, 183 (1969/70).

221. Hall, G., "The Stability of Predictor-Corrector Methods," Comp. J. 9, 410 (1967).

222. Hammer, P. & Hollingsworth, T., "Trapezoidal Methods of Approximating Solutions of Differential Equations," Math. Tables & Other Aids to Computation 9, 92 (1955).

223. Hamming, R., "Stable Predictor-Corrector Methods for Ordinary Differential Equations," J. Assoc. Comp. Mach. 6, 37 (1959).

224. Hansen, E., "Cyclic Composite Multistep Predictor-Corrector Methods," Proc. ACM 24th Nat. Conf. 135 (1969) ACM, New York.

225. Hansen, K., Koen, B., & Little, W., "Stable Numerical Solutions of the Reactor Kinetics Equations," Nuc. Sci. Eng. 25, 183 (1966).

226. Harnett, R., Sanson, F., & Warshawsky, L., "MIDAS--An Analog Approach to Digital Computation," Sim. 3 (1964).

227. Henry, M., "A Best Approximating Solutions of Certain Nonlinear Differential Equations," SIAM J. Num. Anal. 6, 143 (1969).

228. Henry, M., "A Sequence of Best Approximating Solutions of y" F (x, y, y')," SIAM J. Num. Anal. 7, 129 (1970).

229. Henrici, P., "The Propagation of Roundoff Error In the Numerical Solution of Initial Value Problems Involing Ordinary Differential Equations," Proc. IFIP Congress 1968, Booklet A, North-Holland Amsterdam 91 (1968).

230. Heinrich, K., "Runge-Kutta Formulas with Multistep Nodes," Universitat Philosophische Fakultat, Dr. Dissertation, 67 (1969).

231. Heun, K., "Neue Methode zur Approximativen Integration der Differentialgleichunge einer Unabhagigen Veranderlichen," Z. Math. U. Phys. 45, 23 (1960).

232. Ho, Y. & Brentani, P., "On Computing Optimal Control With Inequality Constraints," SIAM J. Control Ser. A, 1, 319 (1963).

233. Houwen, P., "Explicit & Semi-Implicit Runge-Kutta Formulas for the Integration of Stiff Equations," Report TW 132/72, Mathematisch Centrum, Amsterdam, 20 (1972).

234. Houwen, P., "Explicit Runge-Kutta Formulas with Increased Stability Boundaries," Num. Math. 20, 149 (1972).

235. Houwen, P., "One-Step Methods for Linear Initial Value Problems," ZAMM 5k, T58 (1971).

236. Houwen, P., "One-Step Methods for Linear Initial Value Problems I Polynomial Methods," Report TW 119/70, Mathematisch Centrum, Amsterdam, 36 (1970).

237. Houwen, P., "One-Step Methods for Linear Initial Value Problems II, Applications to Stiff Equations," Report TW 122/70. Mathematisch Centrum, 42 (1970).

238. Houwen, P., "Stabilized Runge-Kutta Methods with Limited Storage Requirements," Report TW 124/71, Mathematisch Centrum, Amsterdam, 33 (1971).

239. Houwen, P., "One-Step Methods for Linear Initial Value Problems, Numerical Results," Report TW 130/71, Mathematisch Centrum, Amsterdam, 83 (1971).

240. Houwen, P., "A Survey of Stabilized Runge-Kutta Formulas," MC Tract 37, Chapter 5, Mathematisch Centru, Amsterdam (1971).

241. Huddleston, R., "Selection of Stepsize in the Vairable-Step Predictor-Corrector Method of Van Wyke." J. Comp. Phys. 9, 528 (1972).

242. Hull, T. & Newbery, A., "Corrector Formulas for Multistep Integration Methods," J. SIAM 10 (1962).

243. Hull, T. & Newbery, A., "Corrector Formulas for Multistep Integration Methods," J. SIAM 10, 351 (1962).

244. Hull, T. & Creemer, A., "Efficiency of Predictor-Corrector Procedures," J. Assoc. Comp. Mach. 10, 291 (1963).

245. Hull, T., "The Effectiveness of Numerical Methods for Ordinary Differential Equations," Studies in Numerical Analysis 2, 114 (1968).

246. Hull, T. & Newbery, A., "Integration Procedures Which Minimize Propagated Error," J. SIAM 9, 31 (1961).

247. Hull, T. & Luxemburg, W., "Numerical Methods and Existence Theorms for Ordinary Differential Equations," Num. Math. 2, 30 (1960).

248. Hull, T., "The Numerical Integration of Ordinary Differential Equations," Information Processing #68, ed. A. J. H. Morrell. North-Holland Amsterdam AD-702-133 (1969).

249. Hull, T., "The Numerical Integration of Ordinary Differential Equations," Comp. Sci. Report #1, University of Toronto (1968).

250. Hull, T. & Johnston, R., "Optimum Runge-Kutta Methods," Math. Comp. 18, 306 (1964).

251. Hull, T., "A Search for Optimum Methods for the Numerical Integration of Ordinary Differential Equations," SIAM Review 9, 647 (1967).

252. Hull, T., "Test of Probabilistic Models for Propagation of Roundoff Errors," Comm. ACM 9, 108 (1966).

253. Hulme, B., "One-Step Piecewise Polynominal Galerkin Methods for Initial Value Problems," Math. Comp. 26, 415 (1972).

254. Hulme, B., "Piecewise Polynomial Taylor Methods for Initial Value Problems," Num. Math. 17, 367 (1971).

255. Imhof, J., "On the Method for Numerical Integration of Clenshaw and Curtis," Num. Math. 5, 138 (1963).

256. Ira, M., "A Stabilizing Device for Unstable Solutions of Ordinary Differential Equations, Design and Application of a Filter," Information Processing in Japan 4, 65 (1964).

257. Jain, M. & Srivastava, V., "High Order Stiffly Stable Methods for Ordinary Differential Equations," Dept. of Comp. Sci., Report 394, University of Illinois (1970).

258. Jain, M. & Srivastava, V., "Optimal Stiffly Stable Methods for Ordinary Differential Equations," Dept. of Comp Sci., Report 402, University of Illinois (1970).

259. Jain, R., "Some A-Stable Methods for Stiff Ordinary Differential Equations," Math. Comp. 26, 71 (1972).

260. Johnston, R., "On Optimum Runge-Kutta Methods for the Numerical Solution of Ordinary Differential Equations," M. A. Thesis, University of British Columbia (1961).

261. Jones, R., "Announcement of the Avaliability of RUNKUT: A Computer Routine for Interpreting Systems of First Order Ordinary Differential Equations," Sandia Labs., Albuquerque, New Mexico (1970).

262. Joyce, D., "Survey of Extrapolation Processes in Numerical Analysis," SIAM Review 13, 435 (1971).

263. Kahan, W., "A Computable Error Bound for Systems of Ordinary Differential Equations," SIAM Review 8, 568 (1966).

264. Kahan, D., "Minimum Norm Differentiation Formulas with Improved Roundoff Error Bounds," Math. Comp. 26, 477 (1972).

265. Karim. A., "Stability of the Fourth Order Runge-Kutta Method for the Solution of Systems of Differential Equations," Comp. J. 9, 308 (1966).

266. Karim, A., "A Theorm for the Stability of General Predictor-Corrector Methods for the Solution of Systems of Differential Equations," J. Assoc. Comp. Mach. 15, 706 (1968).

267. Karim, A., "Criterion for the Stability of Numerical Methods for the Solution of Systems of Differential Equations," J. Res. NBS, 71B, (Math. & Math. Phys.) Nos. 2 & 3, 91 (1967).

268. Karim, A., "The Stability of the Fourth Order Runge-Kutta Method for the Solution of Systems of Differential Equations," Comm. ACM 9, 113 (1966).

269. Karim, A., "Sufficient Conditions for the Instability of Numerical Integration Methods," J. of Res. NBS, 73B, 109 (1969).

270. Keller, H., "Accurate Difference Methods for Linear Ordinary Differential Systems Subject to Linear Constraints," SIAM J. Num. Anal. 6, 8 (1969).

271. Kendall, R. & Timlake, W., "A Stable k-Step Method of Order Greater Than k + 2 for the Solution of Ordinary Differential Equations," IBM Publication 37.002, Houston Scientific Cntr. (1967).

272. King, R., "Runge-Kutta Methods With Constrained Minumum Error Bounds," Math. Comp. 20, 386 (1966).

273. Klement've, A., "Stability of the Numerical Integration of Systems of Ordinary Differential Equations Arising in the Application of the Straight Line Method to Some Equations of Oscillation Theory," Zh. vychise. Mat. mat. Fiz. 9, 218 (1969).

274. Klement'ev, A., "Comments on the Stability of Two-Point Pseudo-Runge-Kutta Methods," Differentsial'nye Uravneniia, 5, 751 (1969).

275. Klopfenstein, R. & Davis, C., "PECE Algorithms for the Solution of Stiff Systems of Ordinary Differential Equations," Math. Comp. 25, 457 (1971).

276. Klopfenstein, R., "Numerical Differentiation Formulas for Stiff Systems of Ordinary Differential Equations," RCA Review, 32, 447 (1971).

277. Klopfenstein, R. & Millman, R., "Numerical Stability of a One-Evaluation Predictor-Corrector Algorithm for Numerical Solution of Ordinary Differential Equations," Math. Comp. 22, 557 (1968).

278. Knapp, H., "Ergebrusse einer Untersuchung uber den Wert der Lie-Rheihen-Theorie fur numerische Rechnungen in der Himmelsmechanik," ZAMM 42, T25 (1962).

279. Knapp, H. & Wanner, G., "LIESE, A Program for Ordinary Differential Equations Using Lie Series," Math. Res. Centr. Report #881, Univ. of Wisconsin, Madison (1968).

280. Knapp, H. & Wanner, G., "Numerical Solution of Ordinary Differential Equations by Groebner's Lie Series Method," Math. Res. Cntr. Report #880, Univ. of Wisconsin, Madison (1968).

281. Knapp, H., "Uber eine Verallgemeinerung des Verfahrens der sukzessiven Approximation zur Losung Differentialgleichungsystemen," Monatshefte fur Math. 68, 33 (1964).

282. Kohfeld, J. & Thompson, G., "Multistep Methods with Modified Predictors & Correctors," J. Assoc. for Comp. Math. 14, 155 (1967).

283. Kohfeld, J. & Thompson, G., "A Modification of Nordsieck's Method Using an 'Off-Step' Point," J. Assoc. for Comp. Math. 15, 390 (1968).

284. Kohfeld, J. & Thompson, G., "Multistep Methods with Modified Predictors & Correctors," J. Assoc. for Comp. Math. 14, 155 (1967).

285. Kohfeld, J., "Stability of Numerical Solutions of Ordinary Differential Equations," Doctorial Thesis, Oregon State Univ., Corvallis (1963).

286. Konen, H. & Luther, H., "Some Singular Explicit Fifth Order Runge-Kutta Solutions," SIAM J. Num. Anal. 4, 607 (1967).

287. Kozlov, L. & Krupnik, V., "Computer Integration of the Falkner-Scan Equation in the Case of a Retarded Flow ona Porous Surface with Suction," Vychislitel'naia i Prikladnaia Matematika 9, 88 (1969).

288. Krough, F., "A Note on the Effect of Conditionally Stable Corrector," Math. Comp. 21, 717 (1967).

289. Krough, F., "On Methods of Adam's Type for the Numerical Solution of Ordinary Differential Equations," TRW Report No. 67.3122.2.317 (1967C).

290. Krough, F., "On Testing a Subroutine for the Numerical Integration of Ordinary Differential Equations," Jet Propulsion Lab. Tech. Report #217 (1969B).

291. Krough, F., "Predictor-Corrector Methods of High Order with Improved Stability Characteristics," J. Assoc. Comp. Mach. 13, 374 (1966).

292. Krough, F., "A Test for Instability in the Numerical Solution of Ordinary Differential Equations," J. Assoc. Comp. Mach. 14, 351 (1967).

293. Krough, F., "A Variable Step, Variable Order Multistep Method for the Numerical Solution of Ordinary Differential Equations," Information Processing 68, Vol. I, ed. A. J. H. Morrell, North Holland Publishing Co., Amsterdam, 194 (1969).

294. Krough, F., "VODQ/SVDO/DVDQ-Variable Order Integrators for the Numerical Solution of Ordinary Differential Equations," TU Doc. No. CP-2308, NPO-11643, May 1969.

295. Kruckberger, F. & Unger, H., "On the Numerical Integration of Ordinary Differential Equations and the Determination of Error Bounds," PICC Symposium, Rome, On Differential Integral Equations, Birkhauser, Basel, 369 (1960).

296. Kuleshow, A., "Automation of an Analytic Method for Solving Ordinary Differential Equations," U.S.S.R. Comp. Math. & Math. Phys. 6, 274 (1966).

297. Kuntzmann, J., "Duex Formules Optimales du Fyce de Runge-Kutta," Chiffres, 2, 21 (1959).

298. Kutzmann, J., "Neuere Entwicklungen der Methode von Runge und Kutta," ZAMM, Band 41, Sonderheft, T28 (1961).

299. Kutzmann, J., "Nouvelle methode pour l'integration apporchee des equations differentielles," Information Processing 62, ed. C. Popplewell, Nothe Holland Pub. Co., Amsterdam (1962).

300. Kutta, N., "Beitrag zur naherungsweisen Integration totaler Differential-gleichungen," Z. Math. Phys. 46, 435 (1901).

301. Lambert, R., "An Analysis of the Numerical Stability of Predictor-Corrector Solutions of Nonlinear Ordinary Differential Equations," SIAM J. Num. Anal. 4, 597 (1967).

302. Lambert, J. & Shaw, B., "A Generalization of Multistep Methods for Ordinary Differential Equations," Num. Math. 8, 250 (1966).

303. Lambert, J. & Shaw, B., "A method for the Numerical Solution of y' = f (x,y) Based on a Self-Adjusting Non-Polynomial Interpolant," Math. Comp. 20, 11 (1966).

304. Lambert, J., "Linear Multistep Methods with Mildly Varying Coefficients," Math. Comp. 24, 81 (1970).

305. Lambert, J. & Shaw, B., "Numerical Solution of y' = f (x,y) by a Class of Formulas Based on Rational Approximation," Math. Comp. 19, 456 (1956).

306. Lambert, J. & Shaw, B., "On the Numerical Solution of y' = f (x,y) by a Class of Formulas Based on Polynomial Approximation," Math. Comp. 19, 456 (1965).

307. Lambert, J. & Mitchell, A., "On the Solution of y' = f (x,y) by a Class of High Accuracy Difference Formulas of Low Order," ZAMP 13, 223 (1962).

308. Lambert, J., "Predictor-Corrector Algorithms With Indentical Regions of Stability," SIAM J. Num. Anal. 8, 337 (1971).

309. Lambert, J. & Mitchell, A., "Repeated Quadrature Using Derivatives of the Integrand," Math. Phys. 15, 84 (1964).

310. Laurent, P., "Methodes speciales du type de Runge-Kutta," Premier Congress AFCAL, 27 (1961).

311. Lawson, J., "An Order Five Runge-Kutta Process With Extended Region of Stability," SIAM J. Num. Anal. 3, 593 (1966).

312. Lawson, J., "An Order Six Runge-Kutta Process With Extended Region of Stability," SIAM J. Num. Anal. 4, 620 (1967).

313. Lawson, J., "Generalized Runge-Kutta Processes for Stable Systems with Large Lipschitz Constants," SIAM J. Num. Anal. 4, 372 (1967).

314. Lawson, J., "An Order Six Runge-Kutta Process with Extended Region of Stability," SIAM J. Num. Anal. 4, 620 (1967).

315. Lea, R., "On the Stability of Numerical Solutions of Ordinary Differential Equations," NASA TN D-3760 (1967).

316. Lee, H., "Matrix Filtering as an Aid to Numerical Integration," Proc. IEEE, 55, 1826 (1967).

317. Lemaftre, G., "Remarques sur certaines methodes d'integration des systemes d'equations differentielles, Colloque sur l'analyse numerique," Gauthier-Villars, Paris (1961).

318. Lesh, R., "Methods of Simulating a Differential Analyzer on a Digital Computer," J. Assoc. Comp. Mach. 5, 281 (1968).

319. Lester, W., "DeVogelaere's Method for the Numerical Integration of Second Order Differential Equations without Explicit First Derivatives: Application to Coupled Equations Arising from the Schrodiner Equations," J. Comp. Phys. 3, 322 (1968).

320. Lether, F., "The Use of Richardson Extrapolation in One-Step Methods with Variable Step Size," Math. Comp. 20, 379 (1966).

321. Lewis, H. & Stovall E., "Comments on a Floating Point Version of Nordsieck's Scheme for the Numerical Integration of Differential Equations," Math. Comp. 21, 157 (1967).

322. Liniger, W., "A Criterion for A-Stability of Linear Multistep Integration Formulae," Computing Arch. Elektron. Rechnen. 3, 280 (1968).

323. Liniger, W. & Willoughby, R., "Efficient Integration Method for Stiff Systems of Ordinary Differential Equations," SIAM J. Num. Anal. 7, 47 (1970).

324. Liniger, W. & Willoughby, R., "Efficient Numerical Integration of Stiff Systems of Ordinary Differential Equations," Res. Rep. RC 1970, IBM, Yorktown Heights, N. Y. (1967).

325. Liniger, W., "Global Accuracy and A-Stability of One- and Two-Step Integration Formulae for Stiff Ordinary Differential Equations," Conference on the Numerical Solution of Differential Equations, Springer-Verlag, Berlin, 188 (1969).

326. Liniger, W., "Optimization of a Numerical Method for Stiff Systems of Ordinary Differential Equations," IBM Research Report #RC-2198 (1968).

327. Liniger, W., "Zur Stabilitat der numerischen Integrationsmethoden fur Differential Gleichungen," Doctoral Thesis, U. of Lausanne, Zurich (1957).

328. Little, W., Hansen, E., Mason, E., & Keon, B., "A Stable Numerical Solution of the Reactor Kinetics Equations," Trans. Amer. Nuclear Society, Philadelphia Meeting, 7, 3.4 (1964).

329. Lomakovich, A., "Solution of a Volterra Integrao-Differential Equation by the Runge-Kutta-Fehlberg Method," Vychislitel'naia Prikladnaia Matematika 64 (1969).

330. Lomakovich, A. & Vasilenko, A., "Solution of Volterra Integral Equations by a Runge-Kutta Type Method," Differentsial'nye Uravnenua 4, 2094 (1968).

331. Lomax, H. & Bailey, H., "A Critical Analysis of Various Numerical Integration Methods for Computing the Flow of a gas in Chemical Nonequilibrum," NASA Technical Note, NASA TN D-4109 (1967).

332. Lomax, H., "An Operational Unification of Finite Difference Methods for the Numerical Integration of Ordinary Differential Equations," NASA Technical Report, NASA TR R-262, (1967).

333. Lomax, H., "On the Construction of Highly Stable, Explicit, Numerical Methods for Integration Coupled Ordinary Differential Equations with Parasitic Eigenvalues." NASA Technical Note, NASA TN D-4547, (1968).

334. Lomax, H., "Stable Implicit and Explicit Numerical Methods for Integrating Quasi-Linear Differential Equations with Parasitic, Stiff, and Parasitic-Saddle Eigenvalues," NASA Technical Note, NASA TN D-4703 (1968).

335. Loscalzo, F., "An Introduction to the Applications of Spline Functions to Initial Value Problems," Theory & Applications of Spline Functions, T. N. E. Grenville, ed., Academic Press, New York, 37 (1969).

336. Loscalzo, F., "Numerical Solution of Ordinary Differential Equations by Spline Functions (SPLINDIF)," Tech. Summary Rep. 842, Mathematics Research Center, U. S. Army, Univ. of Wisconsin, Madison (1968).

337. Loscalzo, F. & Schoenberg, I., "On the Use of Spline Functions for the Approximation of Solutions of Ordinary Differential Equations," Math. Res. Cntr. Report #723, Madison (1967).

338. Loscalzo, F., "On the Use of Spline Functions for the Numerical Solution of Ordinary Differential Equations," Tech. Summary Rep. 869, Mathematics Research Center, U. S. Army, Univ. of Wisconsin, Madison (1968).

339. Loscalzo, F. & Talbot, T., "Spline Function Approximations for Solutions of Ordinary Differential Equations," SIAM J. Num. Anal. 4, 433 (1967).

340. Lotkin, M., "On the Accuracy of Runge-Kutta Formulas," Mathematical Table and Other Aids to Computations, 5, 128 (1966).

341. Lozinsku, S., "Error Estimate for Numerical Integration of Ordinary Differential Equations," Soviet Math. Dokl. 163, 1014 (1965).

342. Luther, H., "An Explict Sixth-Order Runge-Kutta Formula," Math. Comp. 22, 434 (1968).

343. Luther, H., "Further Explicit Fifth-Order Runge-Kutta Formulas," SIAM Review 8, 374 (1966).

344. Luther, H. & Sierra, H., "On the Optimal Choice of Fourth-Order Runge-Kutta Formulas," Num. Math. 15, 354 (1970).

345. Luther, H. & Koen, H., "Some Fifth-Order Classical Runge-Kutta Formulas," SIAM Review 7, 551 (1965).

346. Lyche, T., "Chebyshevian Multistep Methods for Ordinary Differential Equations," Num. Math. 19, 65 (1972).

347. Lyubich, Yu., U.S.S.R. Comp. Math. & Math. Phys. 6, 228 (1966).

348. Makinson, G., "High Order Implicit Methods for the Numerical Solutions of Systems of Differential Equations," Comp. J. 11, 305 (1968).

349. Malyshev, Yu., "A Numerical Method of Integrating a First-Order Differential Equation," Diff. Eq. 1, 533 (1965).

350. Martens, H., "A Comparative Study of Digital Integration Methods," Sim. 87 (1969).

351. Martin, D., "Runge-Kutta Methods for Integrating Differential Equations on High Speed Digital Computers," Comp. J. 1, 118 (1958).

352. Maslovskaya, L., "The Stability of Difference Equations," Differentsial'nye Uraveniya 608 (1965).

353. Mayers, D., "The Deferred Approach to the Limit in Ordinary Differential Equations," Comp. J. 7, 54 (1964).

354. Merson, R., "An Operational Method for Study of Integration Processes," Proceeding of Symposium on Data Processing, Weapons Research Establishment, Salisbury, Australia (1957).

355. Merson, R., "On Some Developments in the Study of Processes for Solving Ordinary Differential Equations," An internal memorandum of the Royal Aircraft Est. (1960).

356. Merson, R., "On Some Developments in the Study of Processes for Solving Ordinary Differential Equations," Royal Aircraft Est., Tech. Mem. Space 126, Farmborough, (1969).

357. Meshaka, P., "Arux Methodes D'Integration Numerique Pour Systems Differentiels," Rev. Franc. de Traitment de L'Information, 135 (1964).

358. Metzger, C., "Methodes de Runge-Kutta de Rang Superieure al'Ordre," These de l'Universite de Grenoble (1967).

359. Miller, R., "An Experimental Method for Testing Numerical Stability on Initial-Value Problems," J. Comp. Phys. 2, 1 (1967).

360. Miller, J., "The Numerical Solution of Ordinary Differential Equations," Numerical Analysis: An Introduction, ed. J. Walsh, Academic Press, London, 63 (1966).

361. Milne, W. & Reynolds, R., "Fifth-Order Methods for the Numerical Solutions of Ordinary Differential Equations," J. Assoc. Comp. Mach. 9, 64 (1962).

362. Milen, W. & Reynolds, R., "Stability of a Numerical Solution of Differential Equations," J. Assoc. Comp. Mach. 6, 196 (1959).

363. Milne, W. & Reynolds, R., "Stability of Numerical Solution of Differential Equations-Part II," J. Assoc. Comp. Mach. 7, 46 (1960).

364. Miranker, W., "Difference Schemes for the Integration of Stiff Systems of Ordinary Differential Equations," IBM Research Center, Report #Rc-1977 (1968).

365. Miranker, W. & Liniger, W., "Parallel Methods for the Numerical Integration of Ordinary Differential Equations," Math. Comp. 21, 303 (1967).

366. Miranker, W., "Matricial Difference Schemes for Integrating Stiff Systems of Ordinary Differential Equations," Math. Comp. 25, 717 (1971).

367. Milne, W. & Reynolds, R., "Fifth-Order Mrthods for the Numerical Solution of Ordinary Differential Equations," J. Assoc. Comp. Mach. 9, 64 (1962).

368. Moore, R., "The Automatic Analysis and Control of Error in Digital Computation Based on the Ude of Interval Numbers," Error in Digital Computation 1, 61, New York: John Wiley and Sons, Inc., (1965).

369. Moore, R., "Automatic Local Coordinate Transformations to Reduce the Growth of Error Bounds in the Interval Computations of Solutions of Ordinary Differential Equations," Error in Digital Computation 2, ed. L. B. Ball, Wiley, New York (1965).

370. Morel, H., "Evaluation de l'erreur sur un pas dans la methode de Runge-Kutta," Comptes rendus de l'Academie des Sciences Paris, 243, 1999 (1965).

371. Moretti, G., "The Chemical Kinetics Problem in the Numerical Analysis of Nonequilibrium Flows," Proc. IBM Sci. Comp. Symp., Large-Scale Problems in Physics, 167 (1963).

372. Moretti, G., "A New Technique for the Numerical Analysis of Non-equilibrium Flows," AIAA J. 3, 223 (1965).

373. Morrison, D. & Stoller, L., "A Method for the Numerical Integration of Ordinary Differential Equations," Math. Tables and Other aids to Computation 12, 269 (1958).

374. Morrison, D., "Optimal Mesh Size in the Numerical Integration of an Ordinary Differential Equation," J. Assoc. Comp. Mach. 9, 98 (1962).

375. Morrison, N. & Paler, G., "Solution of Aircraft Differential Equations of Motion on a Digital Computer," Report #GER-10780 Goodyear Aircraft Corp., Akron, Ohio (1962).

376. Murav'ev, C., "On Solution of Ordinary Differential Equations by Separating a Multiplicative Derivative," Diff. Eq. 1, 340 (1965).

377. Neimark, Yu., Smirnova, V., & Sterlin, A., "Investigation of the Stability of Difference Methods," U.S.S.R. Comp. Math. & Math. Phys. 8, 15 (1968).

378. Nesterchuk, A., "On the Solution of Ordinary Differential Equations by Means of Numerical Integration Operators," Applied Physics Lab., Translation KLB, 3, T-642, Johns Hopkins (1971).

379. Newberry, A., "Convergence of Successive Substitution Starting Procedures," Math. Comp. 21, 489 (1967).

380. Nickel, K. & Rieder, P., "A New Procedure Similar to the Runge-Kutta Method," Numerical Mathematics: Theory of Approximation of Differential Equations Edited by L. Collatz, et al., Internationale Schriftenreihe zur numerischen Mathematik, 9, 83 (1968).

381. Nickel, K. & Ritter, K., "Termination Criterion and Numerical Convergence," SIAM J. Num. Anal. 9, 277 (1972).

382. Nigro, B., Woodward, R., & Brucks, C., "A Digital Computer Program for Deriving Optimum Numerical Integration Technique for Real-Time Flight Simulation," AMRL-TR-68-4 Aerospace Medical Research Lab., Wright-Patterson AFB, Ohio (1968).

383. Nigro, B., "An Investigation of Optimally Stable Numerical Integration Methods with Application to Real-Time Simulation," Sim. 13, 253 (1969).

384. Nigro, B., "Study of Numerical Integration Techniques for Real-Time Digital Flight Simulation," AMRL-TR-67-4, Aerospace Medical Research Lab., Wright Patterson Air Force Base, Ohio (1967).

399. Papian, F. & Ball, W., "The Spectrum of Numerical Integration Methods with Computed Variable Stepsize," J. Math. Anal. & Appl. 31, 259 (1970).

400. Pelios, A. & Klopfenstein, R., "Minimual Error Constant Numerical Differentiation Formulas," Math. Comp. 26,467 (1972).

401. Perlin, I., "Study of Methods for the Numerical Solution of Ordinary Differential Equations," Perlin. I. E., Georgia Tech (1963-64).

402. Pickard, W., "Tables for the Step-by-Step Integration of Ordinary Differential Equations of the First Order," J. Assoc. Comp. Mach. 11, 229 (1964).

403. Pope, D., "An Exponential Method of Numerical Integration of Ordinary Differential Equations," Comm. ACM 6, 491 (1963).

404. Prothero, A., "Local Error Estimates for Varibale Step Runge-Kutta Methods," Conference on the Numerical Solution of Differential Equations, J. Tl. Morris, ed., Springer-Verlag, Berlin 228 (1969).

405. Rahme, H., "Stability Analysis of a New Algorithm Used for Integrating a System of Ordinary Differential Equations," J. Assoc. Comp. Mach. 17, 284 (1970).

406. Rahme, H., "A New Look at the Numerical Integration of Ordinary Differential Equations," J. Assoc. Comp. Mach. 16, 496 (1969).

407. Rakitskii, Yu., "New Methods for Calculating the Commencement of a Table in the Numerical Integration of Ordinary Differential Equations," U.S.S.R. Comp. Math. & Math. Phys. 8, 14 (1968).

408. Ralston, A., "Runge-Kutta Methods with Minimun Error Bounds," Math. Comp. 16, 431 (1962).

409. Ralston, A., "Relative Stability in the Numerical Solution of Ordinary Differential Equations," SIAM Review 7, 114 (1965).

410. Ralston, A., "Some Theoretical and Computational Matters Relating to Predictor-Corrector Methods of Numerical Integration," Comp. J. 4, 64 (1961).

411. Ratliff, K., "A Comparison of Techniques for the Numerical Integration of Ordinary Differential Equations," Univ. of Illinois Dept. of Computer Science Report, #274 (1968).

412. Reimer, M., "An Integration Procedure Involving Error Estimation," BIT 5, 164 (1965).

385. Nikoslaev, V., "The Solution of Systems of Ordinary Differential Equations by Expansion in Power Series on High Speed Computers," U.S.S.R. Comp. Math. & Math. Phys. 5, 608 (1965).

386. Nordseick, A., "Automatic Numerical Integration of Ordinary Differential Equations," Proc. Sympos. Appl. Math., Amer. Math. Soc., Providence, R. I., 15, 241 (1963).

387. Nordsieck, A., "Numerical Integration of Ordinary Differential Equations," Math. Comp. 16, 22 (1962).

388. Norsett, S., "An A-Stable Modification of the Adams-Bashforth Methods," Conference on the Numerical Solution of Differential Equations, Dundee, Scotland (1969).

389. Norsett, S., "A Criterion for A (α) Stability of Linear Multistep Methods," BIT 9, 259 (1969).

390. Nugeyre, J., "Un procede mixte (Runge-Kutta, pas lies) d'integration des systems differentield du type $x'' = X(x,t)$, Chiffres 4, 55 (1961).

391. Nystrom, E., "Uber die numerische Integration of Differential Equations in an Automatic Digital Computing Machine," Proc. Camb. Phil. Soc. 47, 96 (1951).

392. Nystrom, E., "Uber die numerische Integration von Differential gleichungen," Acta. Soc. Sci. Fenn. 13, 50 (1925).

393. O'Regan, C., "Step-Size Adjustment at Discontinuities for Fourth-Order Runge-Kutta Methods," Comp. J. 13, 401 (1970).

394. Osborne, M., "A New Method for the Integration of Stiff Systems of Ordinary Differential Equations," Information Processing 68, ed. A. J. H. Morrel, North Holland Pub. Co., Amsterdam, 200 (1969).

395. Osborne, M., "On Nordsieck's Method for the Numerical Solution of Ordinary Differential Equations," BIT 6, 51 (1966).

396. Osborne, M., "An Error Analysis of Finite-Difference Methods for the Numerical Solution of Ordinary Differential Equations," Comp. J. 7, 232 (1964).

397. Osborne, M., "Minimising Truncation Error in Finite Difference Approximations to Ordinary Differential Equations," Math. Comp. 21, 133 (1967).

398. Osborne, M., "A Method for Finite-Difference Approximation to Ordinary Differential Equations," Comp. J. 7, 58 (1964).

413. Reimer, M., "Eine Fehlerabschatzung fur linear Differenzenverfahren," Num. Math. 7, 277 (1965).

414. Reimer, M., "Finite Difference Forms Containing Derivatives of High Order," SIAM J. Num. Anal. 5, No. 4, 725 (1968).

415. Reimer, M., "Eine Fehlerabschatzung fur lineare Differenverfahren," Num. Math. 7, 277 (1965).

416. Reimer, M., "Zur Theorie der linearen Differenzenformeln," Math. Z. 95, 373 (1967).

417. Reutter, F. & Knapp, H., "Uber die numerische Auswertung von Grobners Methode sur Losung von Anfangsevertproblemen gewohnlichers Differential-gleichungsystem." Forschungsbericht Nr. 1367 des Landes, Nordrhein-Westfalan, Westdeutscher Verlag, Koln und Oplader (1964).

418. Rice, J., "Split Runge-Kutta Method for Differential Equations," Journal of Research of the National Bureau of Standards 64B, 151 (1960).

419. Richards, P., Lanniy, W. & Torrey, M., "Numerical Integration of Large, Highly-Damped, Nonlinear Systems," SIAM Review 7, 376 (1965).

420. Richter, W., "Estimation de l'erreur connise dans la methode de M. W. E. Milne pour l'integration d'un systeme de n equations differentielles du premier ordre," Bull. Soc. Neuchaleloise des Sciences Naturelles, 75, 6 (1952).

421. Robertson, H., "The Solution of a Set of Reaction Rate Equations," Numerical Analysis: An Introduction, ed. J. Walsh, Academic Press, London, 178 (1966).

422. Rodabaugh, D., "On Stable Correctors," Comp. J. 13, 98 (1970).

423. Rodabaugh, D. & Wesson, J., "On the Efficient Use of Predictor-Corrector Methods in the Numerical Solution of Differential Equations," NASA TN D-2946 (1965).

424. Roe, G., "Experiments with a New Integration Algorithm," General Electric Rept. 67-C-037 (1967).

425. Romanelli, M., "Runge-Kutta Methods for the Solution of Ordinary Differential Equations," Mathematical Methods for Digital Computers, A. Ralston & H. Wief, ed., John Wiley, New York, 110 (1960).

426. Romberg, W., "Vereinfachte Numerische Integration," Norske Vid. Selsk. Forh. (Trondheim) 28, 30 (1955).

427. Rose, M., "Finite Difference Schemes for Differential Equations," Math. Comp. 18, 179 (1964).

428. Rosen, J., "Multistep Runge-Kutta Methods," NASA Technical Note, NASA TN D-4400 (1968).

429. Rosen, J., "Stability and Bounds for Nonlinear Systems of Difference and Differential Equations," J. Math. Anal. & Appl. 2, 370 (1961).

430. Rosen, J., "The Runge-Kutta Equations by Quadrature Methods," NASA Technical Report, NASA TR R-275 (1967).

431. Rosenbrock, H., "General Implicit Processes for the Numerical Solution of Differential Equations," Comp. J. 5, 329 (1963).

432. Rosenbrock, H., "Some General Implicit Processes for the Numerical Solution of Differential Equations," Comp. J. 5, 329 (1963).

433. Rosser, J., "A Runge-Kutta for All Seasons," SIAM Review 9, 417 (1967).

434. Rubinoff, M., "Digital Computers for Real-Time Simulations," J. Assoc. Comp. Mach. 1, 186 (1955).

435. Rutishauser, H., "Bermerkungen zur numerischen Integration gewohnlicher Differentialgleichungen n-ter Ordnung," Num. Math. 2, 263 (1960).

436. Salzer, H., "Equally-Weighted Formulas for Numerical Differentiation," Num. Math. 4, 381 (1962).

437. Salzerm H., "Numerical Integration of y' = ∅ (x,y,y') Using Oscillatory Interpolation," J. Franklin Inst. 263, 401 (1957).

438. Salzer, H., "Optimal Points for Numerical Differentiation," Num. Math. 2, 214 (1960).

439. Salzer, H., "Trigonometric Interpolation and Predictor-Corrector Formulas for Numerical Integration," ZAMM 42, 403 (1962).

440. Samarskii, A., "Some Problems of Difference Scheme Theory," U.S.S.R. Comp. Math. & Math. Phys. 6, 74 (1966).

441. Sammet, J., "Formula Manipulation Compiler," Datamation 12, 32 (1966).

442. Sandberg, I. & Shichman, H., "Numerical Integration of Stiff Nonlinear Differential Equations," Bell System Tech. J. 47, 511 (1968).

443. Sandberg, I., "Some Properties of a Class of Numerical Integration Forumla," Bell System Tech. J. 46, 2061 (1967).

444. Sandberg, I., "Two Theorems on the Accuracy of Numerical Solutions of Systems of Ordinary Differential Equations," Bell System Tech. J. 46, 1243 (1967B).

445. Sankar-Rao, M., "A Method of Solutions for a System of Second-Order Ordinary Differential Equations Arising in the Theory of the Mean Atmospheric Waves," J. Comp. Phys. 2, 228 (1967).

446. Sarafyan, D., "Composite and Multistep Runge-Kutta Formulas," Louisianna State University in New Orleans, Technical Report #18 (1966).

447. Sarafyan, D. & Brown, R., "Computer Derivations of Algebraic Equations Associated with Runge-Kutta Formulas," BIT 7, 156 (1967).

448. Sarafyan, D., "Determination of Optimal Step-Size for Runge-Kutta Processes," Louisiana State University, Tech. Report #54, Forthcoming (May, 1971).

449. Sarafyan, D., "Error Estimation for Runge-Kutta Methods, Notices Amer. Math. Soc. 12, 572 (1965).

450. Sarafyan, D., "Error Estimation for Runge-Kutta Methods Through Pseudoiterative Formulas," Louisiana State University, Tech. Report #14, (1966).

451. Sarafyan, D., "Estimation of Errors for the Approximate Solution of Differential Equations and Their Systems," Louisiana State Univ., Tech. Report #15 (1966).

452. Sarafyan, D., "Improved Sixth-Order Runge-Kutta Formulas & Approximate Continuous Solution of Ordinary Differential Equations," Tech. Report #40, L.S.U. in New Orleans, Dept. of Math. (1970).

453. Sarafyan, D., "Multi-Order Property of Runge-Kutta Formulas and Error Estimation," Louisiana State University, Tech. Report #29 (1967).

454. Sarafyan, D., "Multistep Methods for the Numerical Solution of Ordinary Differential Equations Made Self-Starting," Math. Research Center Report #495 (1965).

455. Sarafyan, D., "Runge-Kutta Formula in Pseudo-Iterative Form," Notices Amer. Math. Soc. 13, 224 (1966).

456. Sarafyan, D., "7th Order lo-Stage Runge-Kutta Formulas," Tech. Report #38, Louisiana State University in New Orleans, Dept. of Math. (1970).

457. Sarmin, E. & Chudov, L., "On the Stability of the Numerical Integration of Systems of Ordinary Differential Equations Arising in the Use of the Straight Line Method," U.S.S.R. Comp. Math. & Math. Phys. 3, 1537 (1963).

458. Sarmin, E. & Chudov, L., "Stability of the Numerical Integration of Systems of Ordinary Differential Equations Arising in the Application of the Straight Line Method," Zh. vychise, Mat. mat. Fig. 3, 6, 1122 (1963).

459. Schermerhorn, I., "RKS3 FORTRAN Floating Point Runge-Kutta Integration," SHARE Distribution 3010, Devon, Ill. (1964).

460. Schneider, A., "Bemerkung zur numerischen Differentiation," Num. Math. 6, 332 (1964).

461. Schoen, K., "Fifth and Sixth Order PECE Algorithms with Improved Stability Properties," SIAM J. Num. Anal. 8, 244 (1971).

462. Scholz, S., "A Runge-Kutta Method with a Variable Parameter Alpha," ZAMM 49, 517 (1969).

463. Scott, M., "Numerical Solution of Unstable Initial Value Problems by Invariant Inbedding," Comp. J. 13, 397 (1970).

464. Scraton, R., "Estimation of the Truncation Error in Runge-Kutta and Allied Processes," Comp. J. 7, 246 (1964).

465. Scranton, R., "The Numerical Solution of Second-Order Differential Equations not Containing the First Derivative Explicity," Comp. J. 6, 368 (1963).

466. Sedore, S., "SCEPTRE: A Second Generation of Transient Analysis Program," Proc. Computer-aided Circuit Design Seminar (NASA), Cambridge, Mass. 55 (1967).

467. Seidman, T., "On the Stability of Certain Difference Schemes," Num. Math. 5, 201

468. Seinfeld, J., Lapidus, L., & Hwang, M., "Review of Numerical Integration Techniques for Stiff Ordinary Differential Equations," Iand Ec Fundamentals 9, 266 (1970).

469. Shampine, L. & Watts, H., "Block Implicit One-Step Methods," Math. Comp. 23, 731 (1969).

470. Shampine, L. & Watts, H., "Comparing Error Estimators for Runge-Kutta Methods," Math. Comp. 25, 445 (1971).

471. Shampine, L., "Local Extrapolation in the Solution of Ordinary Differential Equations," Math. Comp. 27, 91 (1973).

472. Shampshire, L., "Stability Regions for Extrapolated Runge-Kutta and Adams Methods," SC-RR-72-0223 (1972).

473. Shanks, E., "Higher Order Approximations of Runge-Kutta Type," NASA-TN-D 2920, 6 (1965).

474. Shaw, H., "Discrete Analogs for Continuous Filters," J. Assoc. Comp. Mach. 13, 600 (1966).

475. Shaw, B., "Modified Multistep Methods Based on a Non-Polynomial Intepolant," J. Assoc. Comp. Mach. 14, 143 (1967).

476. Shaw, B., "Some Multistep Formulae for Special High Order Ordinary Differential Equations," Num. Math. 9, 367 (1966).

477. Shell, D., "Extending Stability of AMSINT Interpretation Systems," General Electric Rept. 63GL32 (1963).

478. Shil'krut, D., "A Method for the Approximate Solution of Ordinary Differential Equations," U.S.S.R. Comp. Math. & Math. Phys. 5, 41 (1965).

479. Shintani, H., "Two-Step Processes by One-Step Methods of Order 3 and of Order 4," J. Sci. Hiroshina Univ. Ser. A-I Math. 30, 183 (1966).

480. Silverberg, M., "A New Method of Solving State Variable Equations Permitting Large Step Sizes," Proc. IEEE, 56, 1343 (1968).

481. Simon, W., "Numerical Technique for Solution and Error Estimate for the Inital Value Problem," Math. Comp. 19, 387 (1965).

482. Skatetskli, V., "Finding Integrals in Finite Form for Certain Systems of Second Order Ordinary Differential Equations," Diff. Eq. 3, 443 (1967).

483. Sloan, I., "Errors in Numerov and Runge-Kutta Methods," J. Comp. Phys. 2, 414 (1968).

484. Sloan, I., "Method for the Numerical Solution of Linear Second-Order Differential Equations," J. Comp. Phys. 3, 40 (1968).

485. Sloan, I., "Errors in the Numerov Runge-Kutta Methods," J. Comp. Phys. 2, 414 (1967).

486. Sloate, H. & Bickart, T., "A-Stable Composite Multistep Methods," J. Assoc. Comp. Mach. 20, 7 (1973).

487. Smith, A., "A Lower Estimate of the Cumulative Truncation Error in Milnis Method," Comp. J. 8, 395 (1965).

488. Sommer, D., "New Implicit Runge-Kutta Roumulas & Possible Applications of Them," Aachen, Techische Hochschule, Mathematisch-Naturvissenschaftliche Fakutat, Doktor der Naturwissenscheften Dissertation, 122 (1967).

489. Spijker, M., "On the Structure of Error Estimates for Finite Difference Methods," Num. Math.18, 73 (1971).

490. Spijker, M., "On the Consistency of Finite-Difference Methods for the Solution of Initial-Value Problems," J. Math. Anal. & Appl. 19, 125 (1967).

491. Spijker, M., "Roundoff Error in the Numerical Solution of Second Order Differential Equations," Proc. Conference on the Numerical Solution of Differential Equations, in Dundee (1969).

492. Spijker, M., "Reduction of Roundoff Error by Splitting of Difference Formulas," SIAM J. Num. Anal. 8, 345 (1971).

493. Spijker, M., "Stability and Convergence of Finite-Difference Methods," Doctoral Thesis, Leiden University, Leiden (1968).

494. Springer, T. & Farmer, O., "TAF--A Steady State, Frequency Response, and Time Response Simulation Program," AFIPS Conference Proceedings 33, FJCC (1968).

495. Squier, D., "One-Step Methods for Ordinary Differential Equations," Num. Math. 13, 176 (1969).

496. Stetter, H., "A Study of Strong and Weak Stabilitly in Discretization Algorithms," SIAM J. Num. Anal. 2, 265 (1965).

497. Stetter, H., "Asymptotic Expansions for the Error of Discretization Algorithms for Non-Linear Functional Equations," Num. Math. 7, 18-31 (1965).

498. Stetter, H., "Improved Absolute Stability of Prdeictor-Corrector Schemes," Computing (Arch. Elektron, Rechmen) 3, 286 (1968).

499. Stetter, H., "Local Estimation of the Global Discretization Error," SIAM J. Num. Anal. 8, 512 (1971).

500. Stetter, H., "Stability Properties of the Extrapolation Method," Conference on the Numerical Solution of Differential Equations, j. Tl. Morris, ed., Springer-Verlag, Berlin 225 (1969).

501. Stetter, H., "Stabilizing Prdictors for Weakly Unstable Correctors," Math. Comp. 19, 84 (1965).

502. Stetter, H., "Symmetric Two-Step Algorithms for Ordinary Differential Equations," Computing 5, 267 (1970).

503. Stewart, N., "Certain Equivalent Requirements of Approximate Solutions of x' = f (t,x)," _SIAM J. Num. Anal._ 7, 256 (1970).

504. Stewart, N., "The Comparison of Numerical Methods for Ordinary Differential Equations," Computer Science Report #3, Univ. of Toronto (1968).

505. Stimberg, C., "Simplified Derivation of Runge-Kutta Techniques," _ZAMM_ 47, 413 (1967).

506. Stimberg, C. "Vereinfachte Herleitung von Runge-Kutta Verfahren," _ZAMM_ 47, 413 (1967).

507. Stineman, R., "Digital Time-Domain Analysis of Systems with Widely Separated Poles," _J. Assoc. Comp. Mach._ 12, 286 (1965).

508. Stoer, J., "Uber zwei Algorithmen zur Interpolation mit Rationalen Funktionen," _Num. Math._ 3, 285 (1961).

509. Stoller, L. & Morrison, D., "A Method for the Numerical Integration of Ordinary Differential Equations," _MTAC_ 12, 269 (1958).

510. Stormer, C., "Methodes d'integration numerique des equations differentielles ordinires," _C. R. Congr. Intern. Math._, Strasbourg, 243 (1921).

511. Strauss, J., "Digital Simulation of Continuous Dynamic Systems: An Overview," Fall Joint Computer Conference 339 (1968).

512. Sumner, H., "A Digital Computer Code for the Solution of Sets of First Order Differential Equations," A.E.E.W.R.-453, (A.E.E. Winfrith) (1965).

513. Swartz, B. & Wendroff, B., "Generalized Finite-Difference Schemes," _Math. Comp._ 23, 37 (1969).

514. Tavernini, L., "On UNIP and the Construction of Digital Simulation Programs," _Sim._ 7, 263 (1966).

515. Teptin, A., "Some Properties of the Solutions of Linear Difference Equations in the Nonoscillation Internal," Izhevsk Mechanics Institute, 364 (1964).

516. Tikhonov, A. & Gorbunov, A., "Estimates of the Error of a Runge-Kutta Method of the Choice of Optimal Meshes," _U.S.S.R. Comp. Math. & Math. Phys._ 4, 30 (1964).

517. Tikhonov, A. & Gorbunov, A., "On the Optimality of Implicit Difference Schemes of Adams Type," U.S.S.R. Comp. Math. & Math. Phys. 2, 1089 (1962).

518. Timlake, W., "On An Algorithm of Milne and Reynolds," BIT 5, 276 (1965).

519. Treanor, C., "A Method for the Numerical Integration of Coupled First-Order Differential Equations with Greatly Different Time Constants," Math. Comp. 20, 39 (1966).

520. Tu, K., "Stability and Convergence of General Multistep and Multivalue Methods with Variable Step Size," Dept. of Comp. Science ULUCDCS-R-72-526, (1972).

521. Tyson, T., "An Implicit Integration Method for Chemical Kinetics," TRW Report #9840-6002-Ru000, Redondo Beach, Calif. (Sept. 1964).

522. Urabe, M., "An Implicit One-Step Method of High-Order Accuracy for the Numerical Integration of Ordinary Differential Equations," Num. Math. 15, 151 (1971).

523. Urabe, M., "Error Estimation in Numerical Solution of Equations by Iteration Process," J. Science Hiroshima Univ. Ser. A-I 26, 77 (1962).

524. Urabe, M., "Theory Errors in Numerical Integration of Ordinary Differential Equations," Tech. Sum. Rep. 183, U.S. Army Math. Res. Ctr., Univ. of Wisconsin, Madison, Wis., (1960).

525. Urabe, M., "Theory of Errors in Numerical Integration of Ordinary Differential Equations," J. Science Hiroshima Univ. A-1 Math., 25, p.p. 3-62.

526. Usmani, R., "A Multi-Step Method for the Numerical Integration of Ordinary Differential Equations," ZAMP 23, 465 (1972).

527. Usmani, R., "Boundary Value Techniques for the Numerical Solution of Certain Initial Value Problems in Ordinary Differential Equations," J. Assoc. Comp. Mach. 13, 287 (1966).

528. Vainikko, G., "On the Convergence of the Method of Collocation for Nonlinear Differential Equations," Zh. Vychisl. Mat. Fiz. 6, 35 (1966).

529. Van Wyk, R., "A Variable Mesh Method for the Efficient Numerical Solution of Ordinary Differential Equations," Rocketdyne Res. Rept. 65 (1965).

530. Van Wyk, R., "Variable Mesh Multistep Methods for Ordinary Differential Equations," J. Comp. Phys. 5, 244 (1970).

531. Van Wyk, R., "Variable Mesh Multistep Methods for Ordinary Differential Equations," Rocketdyne Res. Report #67-12 (1967).

532. Van Wyk, R., "Variable Mesh Methods for Differential Equations," NASA Report, CR-1247 (Nov., 1968).

533. Varga, R., "On Higher Order Stable Implicit Methods for Solving Parabolic Partial Differential Equations," J. Math. Phys. 40, 220 (1961).

534. Verner, J., "The Order of Some Implicit Runge-Kutta Methods," Num. Math. 13, 14 (1969).

535. Vitguk, A., "The Problem of Approximate Integration of a System of Ordinary Differential Equations Using S. A. Chaplygin's Method," Diff. Eq. 1, 714 (1965).

536. Warten, R., "Automatic Step-Size Control for Runge-Kutta Integration," IBM J. Res. Develop. 2, 340 (1963).

537. Waters, J., "Methods of Numerical Integration Applied to a System Having Trivial Function Evaluations," Comm. ACM 9, 293 (1966).

538. Watt, J., "Consistency, Convergence and Stability of General Discretizations of the Initial Value Problem," Num. Math. 12, 11 (1968).

539. Watt, J., "The Asympototic Discretization Error of a Class of Methods for Solving Ordinary Differential Equations," Proc. Camb. Phil. Soc. 63, 461 (1967).

540. Watts, H., "A-Stable Black Implicit One-Step Methods," Ph.D. Dissertation, Univ. of New Mexico, Albuquerque, New Mexico (1971).

541. Weiner, H., "Nonlinear Predictors Depending on the Last Observation," SIAM J. Appl. Math. 15, 1378 (1965).

542. Weissinger, J., "Uber zulassige Schrittweiten bei den Adams-Verfahren," ZAMM 53, 121 (1973).

543. Whitney, D., "More About Similarities Between Runge-Kutta and Matrix Exponential Methods for Evaluating Transient Response," Proc. IEEE 57, 2053 (1969).

544. Whitney, D., "Propagated Error Bounds for Numerical Solution of Transient Response," Proc. IEEE 54, 1084 (1966).

545. Widlund, O., "A Note on Unconditionally Stable Linear Multistep Methods," BIT 7, 65 (1967).

546. Wilf, H., "A Stability Criterion for Numerical Integration," J. Assoc. Comp. Mach. 6, 363 (1959).

547. Witty, W., "A New Method of Numerical Integration of Differential Equations," Math. Comp. 18, 497 (1964).

548. Wolfe, M., "The Numerical Solution of Implicit First Order Ordinary Differential Equations with Initial Conditions," Comp. J. 14, 173 (1971).

549. Wojtkowiak, H., "Bedingungen fur stabile Multistep-Integrationformeln mit kleinem Fehler," ZAMM 53, 62 (1973).

550. Wright, K., "Some Relationships Between Implicit Runge-Kutta, Collocation and Lanczos Methods and Their Stability Properties," Nordisk Tidskr. Informationsbehandling 10, 217 (1970).

551. Young, E., "Numerical Integration and Other Techniques for Computer Aided Network Design Programming," Semi-Annal Rep., School of Engineering, Old Dominion University, Norfolk (1970).

552. Zajac, E., "Note on Overly-Stable Difference Approximation," J. Math. Phys. 18, 51 (1964).

553. Zonneveld, J., "Automatic Numerical Integration," Mathematical Centre Tracts #8, Amsterdam (1964).

554. Zurmuhl, R., "Runge-Kutta-Verfahren zur numercshen Integration von Differentialgleichungen n-ter Orndung," ZAMM, 28, 173 (1948).

555. Zverkina, T., "A New Class of Finite Difference Operators," Dokl. Akad. Nauk SSSR, Tom 171, 1412 (1966).

556. Lapidus, L. & Seinfield, J., Numerical Solution of Ordinary Differential Equations, Mathematics in Science and Engineering, 74, Academic Press, Inc., 111 Fifth Ave., New York, New York (1971).

APPENDIX I

RUNGE-KUTTA METHODS

This appendix is intended as a reference for various Runge-Kutta methods available in current literature.

Short form notation will be used henceforth: an Explicit p order, V stage Runge-Kutta method is written as RKE(p,V).

RKSI(p,V) - semi-implicit

RKI(p,V) - implicit

RKMS(p,V) - multi-step

The Runge-Kutta coefficients will be presented in the following form:

$$\begin{array}{c|c} \underline{c} & \underline{a} \\ \hline & \underline{w}^T \end{array}$$

$$k_i = hf(T_n + c_i h,\ Y_n + \sum_{i=1}^{M_i} a_{ij} k_j)$$

$$Y_{n+1} = Y_n + \sum_{i=1}^{V} w_i k_i \qquad i = 1,2,\text{---}V \qquad M_i \le V \tag{A.1}$$

In the following sections various references, where different Runge-Kutta methods may be found, are listed.

Many authors have developed the equations leading to the derivation of the coefficients defining a particular order Runge-Kutta method. These equations will be henceforth called the deriving equations; and these authors will be listed first. If a reader desires to develop his own Runge-Kutta method, he can do so by solving the deriving equations. The field is still wide open, many Runge-Kutta methods can still be developed.

A.1 Explicit Runge-Kutta Methods

RKE(p,V):

Restating the equations for a RKE(p,V)

$$k_i = hf[T_n + c_i h,\ Y_n + \sum_{j=1}^{i-1} a_{ij}k_j]$$

$$\text{with } c_1 = 0 \qquad \text{A.2}$$

$$Y_{n+1} = Y_n + \sum_{i=1}^{V} w_i k_i \qquad i=1,2---V$$

The deriving equations for a RKE(p,V), based on a Taylor series analysis, (as developed in Ch. 2) are listed in; Butcher [3], up to a 8th order method; and Fehlberg [4, 12], up to a 8th order method.

A formulation for solving the non-linear deriving equations, for a RKE, on a computer is being developed and the principle is listed in Sarafyan and Brown [7].

The tedious Taylor series analysis for the deriving equations can be replaced by a, practically, equally tedious 'Quadrative Method.' The Quadrative Method for the deriving equations is, however, a more convenient form for use, as it is general for any older method, unlike the Taylor series form. Quadrative form deriving equations for any order RKE's are listed in Rosen [5].

An excellent reference for the general Runge-Kutta class of methods, all in one handy cover, is Lapidus and Seinfeld [6]. This book also lists commonly used Runge-Kutta methods of various orders.

References for RKE of specific orders are now listed. References for deriving equations will be given first, these will be the deriving equations of that specific order method.

A.1-1 RKE(1,1):

The famous (infamous?) Euler method may be considered to be such a method

$$\begin{array}{c|c} 0 & 0 \\ \hline & 1 \end{array}$$

Euler

A.3

A.1-2 RKE(2,2):

The deriving equations, specifically for a RKE(2,2) are listed in Ralston [8] and Fehlberg [12]. In Ralston [8] the free parameter was manipulated to give a Minimum Truncation Error Bound, henceforth referred to as TEB. It must be noted here that, the truncation error is not minimized but rather its bound is minimized. This may or may not result in a minimized truncation error. However, as the exact value of the truncation error depends on the differential equation being integrated, this course is the only one open for optimizing a method. Bounds of these kinds are quite conservative.

Lapidus and Seinfeld [6] list 3 such methods. One of them, the Heun form, is Ralston's optimum.

$$\begin{array}{c|cc} 0 & & \\ 2/3 & 2/3 & \\ \hline & 1/4 & 3/4 \end{array}$$

Heun

(optimum)

A.4

$$\begin{array}{c|cc} 0 & & \\ 1/2 & 1/2 & \\ \hline & 0 & 1 \end{array}$$

Mid-point Rule

A.5

$$\begin{array}{c|cc} 0 & & \\ 1 & 1 & \\ \hline & 1/2 & 1/2 \end{array}$$

Euler-Cauchy

A.6

When $\frac{dY}{dT}$ is independent of Y, the Heun form becomes a 3rd order method. Johnston [9], Kuntzmann[10], and King [11] also find optimum methods, these are the Heun form, eq. A.4.

Fehlberg [12] has developed methods of order p, coupled with a p+1 order method. Suitable choice of free parameters is made so as to minimize the leading term of the truncation error. This results in a larger permissable step size. Comparison of the solution of the p+1 order with the p order is used to control step size.

The p+1 order shares most of the coefficients of the p order; hence with only 1 or 2 additional stages, a p+1 order solution is available. Though the Fehlberg methods require more stages than conventionally used; this extra calculation results in stepsize control, which is well worth the cost.

The automatic step size control feature of these Fehlberg methods make them computationally more efficient (speed and overall number of function evaluations), than conventional methods. This is so; because when step size controls, like the Richardson Extrapolation procedure (see Lapidus and Seinfeld [6]), are applied to conventional methods; the computational effort doubles. In the opinion of the author; these methods when used without the coupled p+1 order method, are still useful because the TEB has been carefully minimized. The p+1 order method can also be used by itself.

Henceforth Fehlberg methods will be written as: RKE[(p,V); (p+1, V+n)] n = 1,2,3--- .

Fehlberg [12] RKE[(2,2); (3,)]:

	0				
p=2	1	1			Fehlberg
p=3	1/2	1/4	1/4		A.7
		1/2	1/2	- w's for p=2	
		1/6	1/6	2/3 - w's for p=3	

	0					
	1/4	1/4				Fehlberg
p=2	27/40	-189/800	729/800			A.8
p=3	1	214/891	1/33	650/891		
		214/891	1/33	650/891		p=2
		533/2106	0	800/1053	-1/78	p=3

<u>A.1-3</u> <u>RKE(3,3)</u>:

The deriving equations for such methods are listed in Fehlberg [12], and in Ralston [8], as a two parameter family. In Ralston [8] the TEB was minimized by suitable choices of the two free parameters, and the 'optimum' method is:

0				
1/2	1/2			Ralston Optimum
3/4	0	3/4		A.9
	2/9	1/3	4/9	

Johnston [9], Kuntzmann [10] and King [11] also list such optimum methods.

The two optimum methods of King [11] are, however, of a fourth and fifth order when $\frac{dY}{dT}$ is independent of Y. When $\frac{dY}{dT}$ is dependent on Y his methods have a slightly larger TEB than Ralston's method (the method which becomes fourth order has a slightly smaller TEB than the method which becomes fifth order.)

0			
1/3	1/3		
5/6	-5/12	5/4	
	1/10	1/2	2/5

King optimum (when $\frac{dY}{dT}$ independent of Y--fourth order)

A.10

0			
.3550510257	.3550510257		
.8449489743	-.4021612205	1.247110195	
	.1111111111	.5124858262	.3764030627

King optimum (when $\frac{dY}{dT}$ independent of Y--fifth order)

A.11

0			
$\frac{(10-2\sqrt{13})}{6}$	$(10-2\sqrt{13})$		
$\frac{(1+\sqrt{13})}{6}$	-.0581020	.8256939	
	.2071768	.3585646	.4342585

Kuntzmann optimum

A.12

Lapidus and Seinfeld [6] list three methods:

0				Classic
1/2	1/2			A.13
1	-1	2		
	1/6	2/3	1/6	

0				Heun
1/3	1/3			A.14
2/3	0	2/3		
	1/4	0	3/4	

0				
2/3	2/3			Nystrom
2/3	0	2/3		A.15
	1/4	3/8	3/8	

Fehlberg [12] gives methods which incorporate automatic step size adjustment .

Fehlberg [12] RKE[(3,4); (4,5)]:

	0							Fehlberg
	1/4	1/4						A.16
	4/9	4/81	32/81					
p=3	6/7	57/98	-432/343	1053/686				
	1	1/6	0	27/52	49/156			
p=4								
		1/6	0	27/52	49/156		p=3	
		43/288	0	243/614	343/1872	1/12	p=4	

Fehlberg A.17

	0						
	2/7	2/7					
	7/14	77/900	343/900				
p=3	35/38	805/1444	-77175/54872	97125/54872			
p=4	1	79/490	0	2175/3626	2166/9065		
		79/490	0	2175/3626	2166/9065		p=3
		229/1470	0	1125/1813	13718/81585	1/18	p=4

The Fehlberg coupled second order methods, eqa. A.7 and A.8, could be used as third order methods.

The author developed optimum methods of a special kind, these methods require only two, instead of three function evaluations per step.

I. Back Step Method:

A.18

0			
-24/100	-24/100		
76/100	19627/10200	-475/408	
	409/684	-7/36	36/57

Butcher [18] developed such fifth order processes, using negative c's. This back step method has a built-in advantage: in every step except the first; k_3 from (T_n, 1/n) may be used as the k_2 for (T_{n+1},Y_n). This results in one less function evaluation per step, hence has V = 2 rather than V = 3, a considerable advantage. A disadvantage of back step methods is that a starting method is required. For a starting method solve the deriving equations with $c_3 = \frac{76}{100}$ and any choice of c_2.

II. Front Step Methods:

$$
\begin{array}{c|ccc}
0 & & & \\
\frac{1}{10} & \frac{1}{10} & & \\
1\frac{1}{10} & -4\frac{133}{170} & 5\frac{15}{17} & \\
\hline
 & -1\frac{5}{22} & 2\frac{1}{6} & \frac{17}{66}
\end{array}
\qquad \text{A.19}
$$

$$
\begin{array}{c|ccc}
0 & & & \\
1/100 & 1/100 & & \\
101/100 & -990103/19700 & 10000/197 & \\
\hline
 & -9994/606 & 103/6 & 197/606
\end{array}
\qquad \text{A.20}
$$

These front step methods have a built in advantage: in every step except the last; k_3 from (T_n, Y_n) may be used as the k_2 for (T_{n+1}, Y_{n+1}). Like the back step method, this results in a V = 2 rather than V = 3 method. The front step method is self starting and so has an advantage over the back step method.

Each of the methods eqs. A.18, A.19 and A.20 was optimized using the procedure found in Ralston [8]. The c_2's and c_3's were interchanged: a TEB was calculated using a particular c_2 and c_3; then another TEB was calculated using the same c_2 as a c_3, and the same c_3 as a c_2. Eqs. A.18, A.19 and A.20 are the resulting optimum methods.

Method (p, V)	Error Bound (TEB)
Ralston A.9 (3,3)	$.1111\ ML^3h^4$
King A.10 (3,3)	$.1389\ ML^3h^4$
King A.11 (3,3)	$.1391\ ML^3h^4$
Classis A.13 (3,3)	$.5\ ML^3h^4$
Heun A.14 (3,3)	$.6667\ ML^3h^4$
Nystrom A.15 (3,3)	$.25\ ML^3h^4$
Back step A.18 (3,2)	$.3493\ ML^3h^4$
Front step A.19 (3,2)	$.3933\ ML^3h^4$
Front step A.19 (3,2)	$.3649\ ML^3h^4$

A.1-4 RKE(4,4):

More has been written and analyzed about the RKE(4,4) than any other Runge-Kutta method. The references listed here try to sample all the aspects of these methods.

The deriving equations are listed in Butcher [3], Rosen [5], Ralston [8], and Fehlberg [12].

Ralston [8] gives an optimum method (minimum TEB).

0				
.4	.4			Ralston Optimum
.45573725	.29697761	.15875964		A.21
1	.21810040	-3.05096516	3.83286476	
	.17476028	- .55148066	1.20553560	.17118478

In King [11], two optimum methods are derived, both whose TEB's are larger than Ralston's optimum, eq. A.21. The first of these becomes fifth order, and the second, sixth order when $\frac{dY}{sT}$ is independent of Y.

0				King
$\frac{(4-\sqrt{6})}{10}$	$\frac{(4-\sqrt{6})}{10}$			A.22
$\frac{(4+\sqrt{6})}{10}$	$-\frac{(11+4\sqrt{6})}{25}$	$\frac{(42+13\sqrt{6})}{50}$		
1	$\frac{(1+5\sqrt{6})}{4}$	$-\frac{(3+2\sqrt{6})}{2}$	$\frac{(9-\sqrt{6})}{4}$	
	0	$\frac{(16-\sqrt{6})}{36}$	$\frac{(16+\sqrt{6})}{36}$	$\frac{1}{9}$

$$
\begin{array}{c|cccc}
0 & & & & \\
\frac{(5-\sqrt{5})}{10} & \frac{(5-\sqrt{5})}{10} & & & \\
\frac{(5+\sqrt{5})}{10} & -\frac{(5+3\sqrt{5})}{20} & \frac{(3+\sqrt{5})}{4} & & \\
1 & \frac{(-1+5\sqrt{5})}{4} & -\frac{(5+3\sqrt{5})}{4} & \frac{(5-\sqrt{5})}{2} & \\
\hline
 & \frac{1}{12} & \frac{5}{12} & \frac{5}{12} & \frac{1}{12}
\end{array}
$$

King

A.23

Kuntzmann [10] developed an optimum method close to Ralson's optimum, eq. A.21.

$$
\begin{array}{c|cccc}
0 & & & & \\
2/5 & 2/5 & & & \\
3/5 & -3/20 & 3/4 & & \\
1 & 19/44 & -15/44 & 40/44 & \\
\hline
 & 55/360 & 125/360 & 125/360 & 55/360
\end{array}
$$

Kuntzmann

A.24

Hull and Johnston [13] point out that, $c_2 \approx .35$ and $c_3 \approx .45$ (solve deriving equations for remaining coefficients) lead to a minimum TEB. As previously mentioned in A.1-2, these bounds are usually larger than the actual error.

Lapidus and Seinfeld [6] list four methods; one of these, the Gill form, is optimized for round-off errors (has a larger TEB than Ralston's optimum).

Gill

A.25

$$
\begin{array}{c|cccc}
0 & & & & \\
1/2 & 1/2 & & & \\
1/2 & \frac{(\sqrt{2}-1)}{2} & \frac{(2-\sqrt{2})}{2} & & \\
1 & 0 & \frac{-(\sqrt{2})}{2} & 1+\frac{(\sqrt{2})}{2} & \\
\hline
 & 1/6 & \frac{(\sqrt{2}-2)}{6} & \frac{(\sqrt{2}+2)}{6} & \frac{1}{6}
\end{array}
$$

Classic

A.26

$$
\begin{array}{c|cccc}
0 & & & & \\
1/2 & 1/2 & & & \\
1/2 & 0 & 1/2 & & \\
1 & 0 & 0 & 1 & \\
\hline
 & 1/6 & 1/3 & 1/3 & 1/6
\end{array}
$$

Kutta

A.27

$$
\begin{array}{c|cccc}
0 & & & & \\
1/3 & 1/3 & & & \\
2/3 & -1/3 & 1 & & \\
1 & 1 & -1 & 1 & \\
\hline
 & 1/8 & 3/8 & 3/8 & 1/8
\end{array}
$$

Method (p,V)	Error Bound (TEB)
Ralston A.21 (4,4)	$.0546\ ML^4h^5$
King A.22 (4,4)	$.0944\ ML^4h^5$
King A.23 (4,4)	$.1218\ ML^4h^5$
Gill A.25 (4,4)	$.0882\ ML^4h^5$
Classic A.26 (4,4)	$.1014\ ML^4h^5$
Kutta A.27 (4,4)	$.0991\ ML^4h^5$

Tests made by the author showed that the Gill and Classic forms are equal on accuracy while the Ralston form is superior to both these.

Hull [19] gives a good discussion on the problem of optimizing Runge-Kutta and predictor-corrector type methods.

The ambivalence of the minimum TEB measure can be pointed out; according to Luther and Sierra [14], using their own minimum TEB measure, the Kutta form, eq. A.27 is optimum for truncation error.

Blum [15] claims to wipe out the optimum round-off error advantage of the Gill form, eq. A.25, by modifying the arithmetic (computer) sequence of the Kutta form, eq. A.27, making it comparable to the Gill form. Fyfe [16] extends this procedure to any RKE(4,4) (though these modifications require extra programming effort).

Lawson [17] derives a RKE(4,4) with an extended region of stability; for integrating ordinary differential equations with large Lipschitz constants. This is comparable to the other RKE(4,4)'s on accuracy but can take larger step-sizes. Gates [26] gives a RKE(4,5).

Fehlberg [12] lists two methods with his usual feature of automatic step size control (see section A.1-2). Also eqs. A.16 and A.17, Fehlberg third order methods, have fourth order methods coupled, these can be used separately as fourth orders.

	0					Fehlberg		
	2/9	1/9				A.28		
	1/3	1/12	1/4					
	3/4	69/128	-243/128	135/64				
p=4	1	-17/12	27/4	-27/5	16/15			
p=5	5/6	65/432	-5/16	13/16	4/27	5/144		
		1/9	0	9/20	16/45	1/12		p=4
		47/450	0	12/25	32/225	1/30	6/25	p=5

Fehlberg

A.29

$$
\begin{array}{ll|llllll l}
 & 0 & & & & & & & \\
 & \frac{1}{4} & \frac{1}{4} & & & & & & \\
 & \frac{3}{8} & \frac{3}{32} & \frac{9}{32} & & & & & \\
 & \frac{12}{13} & \frac{1932}{2197} & -\frac{7200}{2197} & \frac{7296}{2197} & & & & \\
p=4 & 1 & \frac{439}{216} & -8 & \frac{3680}{513} & -\frac{845}{4104} & & & \\
\hdashline
p=5 & \frac{1}{2} & -\frac{8}{27} & 2 & -\frac{3544}{2565} & \frac{1859}{4104} & -\frac{11}{40} & & \\
\hline
 & & \frac{25}{216} & 0 & \frac{1408}{2565} & \frac{2197}{4104} & -\frac{1}{5} & & p=4 \\
\hdashline
 & & \frac{16}{135} & 0 & -\frac{6656}{12825} & \frac{28561}{56430} & -\frac{9}{50} & \frac{2}{55} & p=5
\end{array}
$$

Sarafyan [28] RKE[(4,4); (5,6)]

Sarafyan

A.30

$$
\begin{array}{ll|llllll l}
 & 0 & & & & & & & \\
 & \frac{1}{2} & \frac{1}{2} & & & & & & \\
 & \frac{1}{2} & \frac{1}{4} & \frac{1}{4} & & & & & \\
p=4 & 1 & 0 & -1 & 2 & & & & \\
\hdashline
 & \frac{2}{3} & \frac{7}{27} & \frac{10}{27} & 0 & \frac{1}{27} & & & \\
p=5 & \frac{1}{5} & \frac{28}{625} & -\frac{125}{625} & \frac{546}{625} & \frac{54}{625} & -\frac{378}{625} & & \\
\hline
 & & \frac{1}{6} & 0 & \frac{2}{3} & \frac{1}{6} & & & p=4 \\
\hdashline
 & & \frac{14}{336} & 0 & 0 & \frac{35}{336} & \frac{162}{336} & \frac{125}{336} & p=5
\end{array}
$$

Fehlberg [12] tested the above three methods, eqs. A.28, A.29 and A.30. Eq. A.28 was the best on accuracy, while eq. A.29 was the worst. On time eq. A.29 was the best while eq. A.30 was the worst. All these three methods, by the same test were faster and more accurate than the Kutta form, eq. A.27. The time savings of these three methods is a result of their automatic step size control feature.

Sarafyan [37] shows that the Classic form, eq. A.26, has embedded within it first, second and third order methods. This is naturally expected of a fourth order method. This embedding principle of Sarafyan can be used to monitor step size. Using eq. A.26: $Y_{n+1}^{[1]} = Y_n + k_1$, $Y_{n+1}^{[2]} = Y_n + k_2$ with $Y_{n+1}^{[4]}$ available by the complete calculation of the method. So here $Y_{n+1}^{[2]}$ (second order accuracy 1/n) is available at no extra cost. Comparing $Y_{n+1}^{[2]}$ and $Y_{n+1}^{[4]}$ gives step size control. $Y_{n+1}^{[3]}$ is not directly available, it is available at $\frac{3}{4}h$ and not at h; so to use it for step size control requires extra effort.

This embedding principle can be extended to any order method, and is a built in advantage of the Runge-Kutta methods.

Various authors have studied the errors involved in fourth order Runge-Kutta methods and developed embedding methods to check these errors and hence institute step size control; regions of stability have also been studied. These authors are: Merson [29], Scraton [20], England [21], Ceschino and Kuntzmann [32], Collatz [33], Lotkin [34], Chai [35], Sarafyan [36,37], Karim [38], Warten [29], Shampaine and Watts [40], Christiansen [41], and O'Regan [42].

Henrici [43] shows how to control round-off error.

Computer programs using fourth order processes with automatic step size control are listed in Basnett [69] (uses the England [31] method), and in Jones [70] (uses the Classic, eq. A.26, method).

The author, using the same principle outlined in section A.1-3, has developed two new classes of RKE(4,4)'s:

I. Back-step RKE(4,4):

0					Optimum
-.1	- .1				A.31
.9	4.65	-3.75			
1	7.44	-6.24	- .2		
	1.06	- .61	1.11	-.56	

Error Bound (TEB) < .22254 ML^4. If for this method $c_3 = .1$ and $c_2 = .9$ was used, the bound would be < .50641 ML^4, more than twice that of eq. A.31. This happens in every case, for the coefficients of a back step method; so it is better to have c_2 as the negative back step coefficient rather than c_3. One function evaluation per step is saved here as the k_3 of (T_n, Y_n) can be used as the k_2 of (T_{n+1}, Y_{n+1}). This method is not self-starting, and needs a regular fourth order method with $c_3 = .9$ to start it.

0					A.32
-1/2	-1/2				
1/2	3/4	-1/4			
1	- 2	1	2		
	1/6	0	2/3	1/6	

Error Bound (TEB) < .26181 ML^4. This method is close to the optimum, has simple coefficients, and can use either the Classic or the Gill form as a starting method.

II. Front Step Method (4,3):

$$
\begin{array}{c|cccc}
0 & & & & \\
.01 & .01 & & & \\
1.01 & -50.52 & 51.53 & & \\
1 & -48.557981 & 49.548558 & .009423 & \\
\hline
 & -8.080858 & 8.585858 & -8.085804 & 8.580804
\end{array}
$$

Error Bound (TEB) < .23178 ML^4. Here again, as in the case of Back Step methods interchanging values of c_2 and c_3 results in a bound nearly twice as large, i.e., < .43719 ML^4 for $c_2 = 1.01$ and $c_3 = .01$. For this method the k_3 of (T_n, Y_n) can be used as the k_2 of (T_{n+1}, Y_{n+1}). One major advantage over the Back Step methods is that the Front Step methods are self starting. The optimum Front Step eq. A.33 has a bound not much larger than the optimum Back Step eq. A.31; so the Front Step methods should have great potential.

As there are only two free parameters in the fourth order case, a back and front step (V = 2) fourth order method is not possible; but as the fifth order case has five free parameters a back and front step fifth order is possible. The author is working on such fifth and higher order methods.

On comparing error bounds with the other fourth order methods it is seen that the bounds of the Front and Back step methods are much larger; this is to be expected, and is the price paid for achieving V = 3 in a fourth order method.

A.1-5 RKE(5,6):

The RKE(5,6)'s are claimed by many exponents, in the Runge-Kutta field, to be the best compromise between accuracy and computational efficiency (computer time).

The deriving equations are presented in Butcher [3], Fehlberg [4], Luther and Konen [20], Luther [21], Konen and Luther [22]. These also give general solutions to the deriving equations. The last three references explore the complete range of solutions possible, in terms of two and one parameter families. These lead to various Newton-Cotes, Gauss, Radan and Lobatto family of formulae. Cassity [23], Cassity [24] and Lawson [25] also state and solve the deriving equations generally in terms of various free parameters.

Rosen [4] gives deriving equations using the Quadrative approach.

Lapidus and Seinfeld [6] also list a number of RKE (5,6) including the Kutta form corrected by Nystom.

0							
1/3	1/3						Nystrom
2/5	4/25	6/25					A.34
1	1/4	-12/4	15/4				
2/3	6/81	90/81	-50/81	8/81			
4/5	6/75	36/75	10/75	8/75	0		
	23/192	0	125/192	0	-81/192	125/192	

I. RKE(5,6) of Newton-Cotes Quadrative Family:

0							
1	1						Luther [21]
1	1/2	1/2					A.35
1/4	14/64	5/64	-3/64				
1/2	-12/96	-12/96	8/96	64/96			
3/4	0	-9/64	5/64	16/64	36/64		
	7/90	0	7/90	32/90	12/90	32/90	

Luther [21]

A.36

0						
1	1					
1/2	3/8	1/8				
1	-1/2	-1/2	4/2			
1/2	4/64	-5/64	20/64	-3/64		
3/4	12/64	9/64	-12/64	7/64	32/64	
	7/90	0	12/90	7/90	32/90	32/90

Butcher [18]

A.37

0						
1/8	1/8					
1/4	0	1/4				
1/2	1/2	-1	1			
3/4	3/16	0	0	9/16		
1	-5/7	4/7	12/7	-12/7	8/7	
	7/90	0	32/90	12/90	32/90	7/90

Butcher [18]

A.38

0						
1/4	1/4					
1/4	1/8	1/8				
1/2	0	-1/2	1			
3/4	3/16	0	0	9/16		
1	-3/7	2/7	12/7	-12/7	8/7	
	7/90	0	32/90	12/90	32/90	7/90

This Butcher form, eq. A.38, is recommended by Lapidus and Seinfeld [6] (after extensive tests) as the best RKE (p,v) to use. Also Sarafyan [44] uses this form for an ingenious error analysis, hence this form is highly recommended. The author's test show this method to be the best RKE(5,6) on accuracy.

0						
-1/2	-1/2					
1/4	5/16	-1/16				
1/2	-3/4	1/4	1			
3/4	3/16	0	0			
1	0	-1/7	12/7	-12/7	8/7	
	7/90	0	32/90	12/90	32/90	7/90

Butcher [18]

A.39

This Butcher form is of the back step type, discussed in section A.1-3. This allows in all steps after the first, to use k_4 from (T_n, Y_n) for the k_2 of (T_{n+1}, Y_{n+1}). Hence, overall, this method requires V = 5 instead of V = 6. Another such method is (not of the Newton-Cotes family):

0						
-1/5	-1/5					
2/5	4/5	-2/5				
1/3	7/36	0	5/36			
4/5	0	0	4/5	0		
1	1/4	0	-35/4	54/7	25/14	
	5/48	0	0	27/56	125/336	1/24

Butcher [18]

A.40

Here $c_2 = -1/5$ $c_5 = 4/5$ so allows the use of k_5 from (T_n, Y_n) for the k_2 of (T_{n+1}, Y_{n+1}). The larger number of zeros present in eq. A.40 as compared to eq. A.39 should make eq. A.40 more computationally efficient than eq. A.39.

The extra programming effort required to save the k's from a previous step, for use at a present step, is trivial.

Another Butcher [18] method, which can be used to start eq. A.40, is:

0						
1/5	1/5					
2/5	0	2/5				
1/3	7/36	0	5/36			
4/5	0	0	4/5	0		
1	1/4	0	-35/4	54/7	25/14	
	5/48	0	0	27/56	125/336	1/24

Butcher [18]

A.41

Lawson [25] claims a form, similar to his RKE(4,4). This method has an extended region of stability:

0						
1/2	1/2					
1/4	3/16	1/16				
1/2	0	0	1/2			
3/4	0	-3/16	6/16	9/16		
1	1/7	4/7	6/7	-12/7	8/7	
	7/90	0	32/90	12/90	32/90	7/90

Lawson-form

A.42

The author's tests confirmed that this Lawson form, eq. A.42, does have a larger region of stability than other RKE(5,6)'s, and so can take larger step sizes. On accuracy this form is as good as the Bulcher form, eq. A.38. These two are the best RKE(5,6)'s on accuracy.

II. RKE(5,6) of the Gauss Quadrative family:

Luther and Konen [20]

A.43

$$
\begin{array}{c|cccccc}
0 & & & & & & \\
1 & 1 & & & & & \\
\frac{1}{2} & \frac{3}{8} & \frac{1}{8} & & & & \\
1 & -\frac{1}{2} & -\frac{1}{2} & \frac{4}{2} & & & \\
\frac{(5-\sqrt{15})}{10} & -\frac{\sqrt{15}}{100} & -\frac{10}{100} & \frac{(60-8\sqrt{15})}{100} & -\frac{\sqrt{15}}{100} & & \\
\frac{(5+\sqrt{15})}{10} & \frac{(-6-\sqrt{15})}{20} & -\frac{2}{20} & \frac{12}{20} & \frac{(6-\sqrt{15})}{20} & \frac{4\sqrt{15}}{20} & \\
\hline
 & 0 & 0 & \frac{8}{18} & 0 & \frac{5}{18} & \frac{5}{18}
\end{array}
$$

III. RKE(5,6) of the Radau Quadrative family:

Luther and Konen [20]

A.44

$$
\begin{array}{c|cccccc}
0 & & & & & & \\
\frac{4}{11} & \frac{4}{11} & & & & & \\
\frac{2}{5} & \frac{9}{50} & \frac{11}{50} & & & & \\
1 & 0 & -\frac{11}{4} & \frac{15}{4} & & & \\
\frac{(6-\sqrt{6})}{10} & \frac{(81+9\sqrt{6})}{600} & 0 & \frac{(255-55\sqrt{6})}{600} & \frac{(24-14\sqrt{6})}{600} & & \\
\frac{(6+\sqrt{6})}{10} & \frac{(81-9\sqrt{6})}{600} & 0 & \frac{(255+55\sqrt{6})}{600} & \frac{(24+14\sqrt{6})}{600} & 0 & \\
\hline
 & \frac{4}{36} & 0 & 0 & 0 & \frac{(16+\sqrt{6})}{36} & \frac{(16-\sqrt{6})}{36}
\end{array}
$$

The deriving equations for the Radau family of RKE(5,6)'s are given and solved in terms of free parameters in Luther [21] and Konen and Luther [22]. The above equation is one such solution.

IV. RKE(5,6) of the Lobatto Quadrative family:

Luther and Konen [20]

A.45

$$
\begin{array}{c|cccccc}
0 & & & & & & \\
\frac{1}{2} & \frac{1}{2} & & & & & \\
\frac{(5-\sqrt{5})}{10} & \frac{2}{10} & \frac{(3-\sqrt{5})}{10} & & & & \\
\frac{1}{2} & \frac{1}{4} & \frac{1}{4} & 0 & & & \\
\frac{(5+\sqrt{5})}{10} & \frac{(1-\sqrt{5})}{20} & -\frac{4}{20} & \frac{(5+3\sqrt{5})}{20} & \frac{8}{20} & & \\
1 & \frac{(\sqrt{5}-1)}{4} & \frac{(2\sqrt{5}-2)}{4} & \frac{(5-\sqrt{5})}{20} & -\frac{8}{4} & \frac{(10-2\sqrt{5})}{4} & \\
\hline
 & \frac{1}{12} & 0 & \frac{5}{12} & 0 & \frac{5}{12} & \frac{5}{12}
\end{array}
$$

Luther [21]

A.46

$$
\begin{array}{c|cccccc}
0 & & & & & & \\
1 & 1 & & & & & \\
\frac{1}{2} & \frac{3}{8} & \frac{1}{8} & & & & \\
1 & -\frac{1}{2} & -\frac{1}{2} & \frac{4}{2} & & & \\
\frac{(5-\sqrt{5})}{10} & \frac{(25-7\sqrt{5})}{100} & \frac{(5-5\sqrt{5})}{100} & \frac{(20+4\sqrt{5})}{100} & -\frac{2\sqrt{5}}{100} & & \\
\frac{(5+\sqrt{5})}{10} & \frac{(3+\sqrt{5})}{20} & \frac{(1+\sqrt{5})}{20} & \frac{(4-4\sqrt{5})}{20} & \frac{2}{20} & \frac{4\sqrt{5}}{20} & \\
\hline
 & \frac{1}{12} & 0 & 0 & \frac{1}{12} & \frac{5}{12} & \frac{5}{12}
\end{array}
$$

Konen and Luther [22] solve the deriving equations of the Lobatto family in terms of free parameters. The above two equations are two such solutions.

The advantage of the Newton-Cotes, Gauss, Radau and Lobatto families is that when $\frac{dY}{dT}$ is independent of Y, they become sixth order quadrative formulae of these families.

Luther and Konen [20] point out that, on basis of computational efficiency, due to the number of zeros present, the Gauss and Radau forms are better than the Lobatto forms, which in turn is better than the Newton Cotes form. While a round-off error minimization; following the principle outlined in Gates [26], shows that on accuracy (based on round-off error), the Lobatto form is better than the other three.

Shanks [27] developed a class of RKE(5,5)'s rather than RKE(5,6)'s. This was done by solving the non-linear deriving equation approximately, not exactly, resulting in one less function evaluation. The theory for reducing V for RKE's is also given here for $p \leq 7$, making it possible to derive more efficient formulae as compared to the conventional approach. However, it must be pointed out that (in the opinion of the author) methods derived this way are not exactly p order, though they are greater than p-1 order. This is because the truncation error, in such methods has a larger bound. The author's tests confirmed that Shank's RKE(5,5)'s are faster, though less accurate than, RKE(5,6)'s.

$$
\begin{array}{c|ccccc}
0 & & & & & \\
\frac{1}{1000N} & \frac{1}{1000N} & & & & \\
\frac{3}{10} & \frac{(-450N + 3)}{10} & \frac{450N}{10} & & & \\
\frac{3}{4} & \frac{(2250N - 9)}{8} & -\frac{2250N}{8} & \frac{15}{8} & & \\
1 & \frac{(-103500N + 459)}{81} & \frac{103500N}{81} & -\frac{490}{81} & \frac{112}{81} & \\
\hline
 & \frac{105}{1134} & 0 & \frac{500}{1134} & \frac{448}{1134} & \frac{81}{1134}
\end{array}
$$

Shanks [27]

A.47

Shanks suggests that $1200\ Nh^2 > 1$ be chosen to give a valid fifth order method. The author tested values of N from 100,000 to .0001 and found that $N \geq 10$ gave discrepencies in the coefficients, while $N < 10$ does give a valid method. Shanks suggests $N = 9$. The author's test with various values of N showed that $N = 5$ gave the most accurate method.

The author modified the Shanks form to give a method of four instead of five stages. This was done by using $N = 100$, resulting in $c_2 = \frac{1}{100{,}000}$. As c_2 is much smaller than c_3, c_4 or c_5; $c_2 = 0$ was used. This makes $k_2 = k_1$ and hence one function evaluation is saved.

$$
\begin{array}{c|ccccc}
0 & & & & & \\
0 & 0 & & & & \\
\frac{3}{10} & \frac{(-450N+3)}{10} & \frac{450N}{10} & & & \\
\frac{3}{4} & \frac{(2250N-9)}{8} & -\frac{2250N}{8} & \frac{15}{9} & & \\
1 & \frac{(-103500N+459)}{81} & \frac{103500N}{81} & -\frac{490}{81} & \frac{112}{81} & \\
\hline
 & \frac{105}{1134} & 0 & \frac{500}{1134} & \frac{448}{1134} & \frac{81}{1134}
\end{array}
$$

Modified Shanks

A.48

The author's tests showed that this method is faster though less accurate than the other Shanks forms.

Fehlberg [4] develops methods of order p, coupled with a p+1 order method for automatic step-size control, based on minimizing the TEB for the p order method as discussed in section A.1-2.

Fehlberg

A.49

$$
\begin{array}{c c|c c c c c c c c}
 & 0 & & & & & & & & \\
 & \frac{1}{6} & \frac{1}{6} & & & & & & & \\
 & \frac{4}{15} & \frac{4}{75} & \frac{16}{75} & & & & & & \\
 & \frac{2}{3} & \frac{5}{6} & -\frac{8}{3} & \frac{5}{2} & & & & & \\
 & \frac{4}{5} & -\frac{8}{5} & \frac{144}{25} & -4 & \frac{16}{25} & & & & \\
p=5 & 1 & \frac{361}{320} & -\frac{18}{5} & \frac{407}{128} & -\frac{11}{80} & \frac{55}{128} & & & \\
\hdashline
 & 0 & -\frac{11}{640} & 0 & \frac{11}{256} & -\frac{11}{160} & \frac{11}{256} & 0 & & \\
p=6 & 1 & \frac{93}{640} & -\frac{18}{5} & \frac{803}{256} & -\frac{11}{160} & \frac{96}{256} & 0 & 1 & \\
\hline
p=5 & & \frac{31}{384} & & & & & \frac{5}{66} & & \\
 & & & 0 & \frac{1125}{2816} & \frac{9}{32} & \frac{125}{768} & & & \\
\hdashline
p=6 & & \frac{7}{1480} & & & & & 0 & \frac{5}{66} & \frac{5}{66}
\end{array}
$$

Sarafyan [28] develops similar type methods with step size control, based on the embedding principle outlined in section A.1-4. But Sarafyan prefers to control his p order methods by embedded p-i, i = 1,2,3---, order methods. This requires no extra function evaluations, so is faster than Fehlberg's methods. But as Fehlberg uses a p order method controlled by a p+1 order method, his error estimate will be better than Sarafyan; and hence a larger step size is possible with Fehlberg's methods, at the cost of extra function evaluations. However the Sarafyan methods can obviously be used in the same way as the Fehlberg methods, by using the p order method to control a p-1 order solution.

The Sarafyan and Fehlberg step size control principles are convenient for use and are extendible to any order method.

A Sarafyan fifth order with an embedded fourth order, is eq. A.30 listed in sections A.1-4. Sarafyan [28] gives the deriving equations of such methods and solves them generally, giving six such embedded methods.

Sarafyan

A.50

$$
\begin{array}{ll|llllll l}
 & 0 & & & & & & & \\
p=2 & \frac{1}{2} & \frac{1}{2} & & & & & & \\
\hdashline
 & \frac{1}{2} & \frac{1}{4} & \frac{1}{4} & & & & & \\
p=4 & 1 & 0 & -1 & 2 & & & & \\
\hdashline
 & \frac{6}{10} & \frac{24}{1000}(11 & 15 & -2 & 1) & & & \\
p=5 & \frac{3}{4} & \frac{3}{256}(18 & 24 & 40 & 7 & -25) & & \\
\hline
 & & 0 & 1 & & & & & p=2 \\
\hdashline
 & & \frac{1}{6}(1 & 0 & 2 & 1) & & & p=4 \\
\hdashline
 & & \frac{1}{54}(7 & 0 & 108 & 0 & -125 & 64) & p=5
\end{array}
$$

Sarafyan

A.51

$$
\begin{array}{ll|llllll l}
 & 0 & & & & & & & \\
p=2 & \frac{1}{2} & & & & & & & \\
\hdashline
 & \frac{1}{2} & \text{Same as eq. A.50} & & & & & & \\
p=4 & 1 & & & & & & & \\
\hdashline
 & \frac{8}{10} & \frac{16}{1000}(13 & 10 & 24 & 3) & & & \\
p=5 & \frac{7}{10} & \frac{7}{80}(3 & 4 & 0 & 0 & 1) & & \\
\hline
 & & \text{Same as} & & & & & & p=2 \\
\hdashline
 & & \text{eq. A.50} & & & & & & p=4 \\
\hdashline
 & & \frac{1}{504}(69 & 0 & 616 & -56 & 875 & -1000) & p=5
\end{array}
$$

Sarafyan

A.52

$$
\begin{array}{ll|l}
 & 0 & \\
p=2 & \frac{1}{2} & \text{Same as} \\
\hline
 & \frac{1}{2} & \text{eq. A.50} \\
p=4 & 1 & \\
\hline
 & \frac{2}{3} & \frac{1}{27}\,(7 \quad 10 \quad 0 \quad 1) \\
p=5 & \frac{3}{2} & \frac{3}{8}\,(0 \quad -30 \quad 0 \quad -12 \quad 45) \\
\hline
 & & \text{Same as} \qquad\qquad\qquad\qquad p=2 \\
\hline
 & & \text{eq. A.50} \qquad\qquad\qquad\qquad p=4 \\
\hline
 & & .15 \quad 0 \quad \frac{65}{75} \quad \frac{20}{75} \quad -.27 \quad -\frac{1}{75} \qquad p=5
\end{array}
$$

Sarafyan

A.53

$$
\begin{array}{ll|l}
 & 0 & \\
p=2 & \frac{1}{2} & \text{Same as} \\
\hline
 & \frac{1}{2} & \text{eq. A.50} \\
p=4 & 1 & \\
\hline
 & \frac{2}{3} & \frac{1}{27}\,(7 \quad 10 \quad 0 \quad 1) \\
p=5 & \frac{2}{10} & \frac{16}{1000}\,(28 \quad -125 \quad 546 \quad 54 \quad -378) \\
\hline
 & & \text{Same as} \qquad\qquad\qquad\qquad p=2 \\
\hline
 & & \text{eq. A.50} \qquad\qquad\qquad\qquad p=4 \\
\hline
 & & \frac{1}{336}\,(14 \quad 0 \quad 0 \quad 35 \quad 162 \quad 125) \qquad p=5
\end{array}
$$

Sarafyan

A.54

	0	Same as eq. A.50						
p=2	$\frac{1}{2}$							
	$\frac{1}{2}$							
p=4	1							
	$\frac{2}{3}$	$\frac{1}{27}$ (7	10	0	1)			
p=5	$\frac{7}{10}$	.2478	.35	.0896	.0504	−.0378		
			or					
		.014(177	250	64	36	−27)		
		Same as						p=2
		eq. A.50						p=4
		$\frac{1}{84}$ (11	0	140	−567	500	0)	p=5

Sarafyan

A.55

	0	Same as eq. A.50						
p=2	$\frac{1}{2}$							
	$\frac{1}{2}$							
p=4	1							
	$.1(6 - \sqrt{6})$	$.002[(93 + 2\sqrt{6})$	0	$4(56 - 11\sqrt{6})$	$(3 - 8\sqrt{6})]$			
p=5	$.1(6 + \sqrt{6})$	$.0004[9(29 - 6\sqrt{6})$	0	$4(123 - 47\sqrt{6})$	$(363 - 32\sqrt{6})$	$4(96 + 131\sqrt{6})]$		
		Same as						p=2
		eq. A.50						p=4
		$\frac{1}{36}$ [4	0	0	0	$(16 + \sqrt{6})$	$(16 - \sqrt{6})$	p=5

All these embedding forms eq. A.50 to eq. A.55 are comparable to other conventional fifth order methods on accuracy but their step size control makes them faster.

Sarafyan [44] employs an ingenious step size control with:

$$
\begin{array}{c|cccccc}
0 & & & & & & \\
1/2 & 1/2 & & & & & \\
1/2 & 1/4 & 1/4 & & & & \\
1 & 0 & -1 & 2 & & & \\
3/2 & 3/8 & 0 & 0 & 9/8 & & \\
2 & -6/7 & 4/7 & 24/7 & -24/7 & 16/7 & \\
\hline
 & 7/45 & 0 & 32/45 & 12/45 & 32/45 & 7/45
\end{array}
$$

Sarafyan

A.56

This form is gotten from the Butcher form, eq. A.38, by multiplying all coefficients by a factor of 2. Sarafyan shows that in going from (T_n, Y_n) to (T_{n+1}, Y_{n+1}); using eq. A.38 with h, Y_{n+1} is available to a fourth order accuracy; at the same time, using eq. A.50 with 2h, Y_{n+2} is available to a fifth order accuracy. Net result - overall V=3 is achieved although the RKE(5,6) of Butcher is used; and by comparing fourth and fifth order Y_{n+1}'s step size can be controlled.

In the opinion of the author, this Sarafyan method should be more accurate than eq. A.50 to eq. A.55, because the most accurate fifth order, eq. A.38, is used here.

A.1-6 RKE(6,7):

The deriving equations are presented in Butcher [3], Fehlberg [4], and Rosen [5]. The first two also give the general solution of these equations, in terms of various free parameters.

In Butcher [3] are listed four RKE(6,7)'s

0							
1/3	1/3						
2/3	0	2/3					
1/3	1/12	1/3	-1/12				
1/2	-1/16	9/8	-3/16	-3/8			
1/2	0	9/8	-3/8	-3/4	1/2		
1	9/44	-9/11	63/64	18/11	0	-16/11	
	11/120	0	27/40	27/40	-4/15	-4/15	11/120

Butcher Newton-Cotes

A.57

A back step and front step type method (see sections A.1-3 and A.1-4) is also listed.

0							
1	1						
2/3	4/9	2/9					
1/3	11/36	1/9	-1/12				
-1/3	151/36	29/9	-7/4	-6			
4/3	-112/9	-116/9	32/3	18	-2		
1	-5/4	-29/9	397/276	152/69	-10/69	1/69	
	23/160	0	29/80	29/80	-1/160	-1/160	23/160

Butcher Newton-Cotes

A.58

In eq. A. 58 k_3 from (T_n, Y_n) can be used for the k_5 of (T_{n+1}, Y_{n+1}); and the k_6 of (T_n, Y_n) can be used as the k_4 of (T_{n+1}, Y_{n+1}). So overall V=4 is achieved, hence this method is highly recommended.

Butcher Newton-Cotes

A.59

$$
\begin{array}{c|ccccccc}
0 & & & & & & & \\
\frac{1}{2} & \frac{1}{2} & & & & & & \\
\frac{2}{3} & \frac{2}{9} & \frac{4}{9} & & & & & \\
\frac{1}{3} & \frac{7}{36} & \frac{2}{9} & -\frac{1}{12} & & & & \\
\frac{5}{6} & -\frac{35}{144} & -\frac{55}{36} & \frac{35}{48} & \frac{15}{8} & & & \\
\frac{1}{6} & -\frac{1}{360} & -\frac{11}{36} & -\frac{1}{8} & \frac{1}{2} & \frac{1}{10} & & \\
1 & -\frac{41}{260} & \frac{22}{13} & \frac{43}{156} & -\frac{118}{39} & \frac{32}{195} & \frac{80}{39} & \\
\hline
 & \frac{13}{200} & 0 & \frac{11}{40} & \frac{11}{40} & \frac{4}{25} & \frac{4}{25} & \frac{13}{200}
\end{array}
$$

Butcher Lobatto

A.60

$$
\begin{array}{c|ccccccc}
0 & & & & & & & \\
\frac{5 \mp \sqrt{5}}{10} & \frac{5 \mp \sqrt{5}}{10} & & & & & & \\
\frac{5 \pm \sqrt{5}}{10} & \mp\frac{\sqrt{5}}{10} & \frac{5 \pm 2\sqrt{5}}{10} & & & & & \\
\frac{5 \mp \sqrt{5}}{10} & \frac{-15 \pm 7\sqrt{5}}{10} & \frac{-1 \pm \sqrt{5}}{4} & \frac{15 \mp 7\sqrt{5}}{10} & & & & \\
\frac{5 \pm \sqrt{5}}{10} & \frac{5 \mp \sqrt{5}}{60} & 0 & \frac{1}{6} & \frac{15 \pm 7\sqrt{5}}{60} & & & \\
\frac{5 \mp \sqrt{5}}{10} & \frac{5 \pm \sqrt{5}}{60} & 0 & \frac{9 \mp 5\sqrt{5}}{12} & \frac{1}{6} & \frac{-5 \pm 3\sqrt{5}}{10} & & \\
1 & \frac{1}{6} & 0 & \frac{-55 \pm 25\sqrt{5}}{12} & \frac{-25 \mp 7\sqrt{5}}{12} & 5 \mp 2\sqrt{5} & \frac{5 \pm \sqrt{5}}{2} & \\
\hline
 & \frac{1}{12} & 0 & 0 & 0 & \frac{5}{12} & \frac{5}{12} & \frac{1}{12}
\end{array}
$$

Luther [45] derives an 'optimum' Lobatto form:

$$
\begin{array}{c|llllllll}
0 & & & & & & & & \\
1 & 1 & & & & & & & \\
\frac{1}{2} & \frac{1}{8}[& 3 & 1\ ] & & & & & \\
\frac{2}{3} & \frac{1}{27}[& 8 & 2 & 8\ ] & & & & \\
\frac{(7-\sqrt{21})}{14} & \frac{1}{392}[& 3(3\sqrt{21}-7) & -8(7-\sqrt{21}) & 48(7-\sqrt{21}) & -3(21-\sqrt{21})] & & & \\
\frac{(7+\sqrt{21})}{14} & \frac{1}{1960}[& -5(231+51\sqrt{21}) & -40(7+\sqrt{21}) & -320\sqrt{21} & 3(21+121\sqrt{21}) & 392(6+\sqrt{21})\ ] & & \\
1 & \frac{1}{180}[& 15(22+7\sqrt{21}) & 120 & 40(7\sqrt{21}-5) & -63(3\sqrt{21}-2) & -14(49+9\sqrt{21}) & 70(7-\sqrt{21})] & \\
\hline
 & \frac{1}{180}[& 9 & 0 & 64 & 0 & 49 & 49 & 9]
\end{array}
$$

Luther Lobatto

A.61

All these quadrative forms eqs. A.57 to eq. A.61 become seventh order when $\frac{dY}{dT}$ is independent of Y.

Lawson [46] lists a form with an extended order of stability (similar analysis to his RKE(4,4) and RKE(5,6), but admits that its accuracy improvement is marginal compared to other sixth order formulae, in fact worse for some non-linear problems.

Fehlberg [4] lists a RKE[(6,8);(7,10)], with the usual advantage of Fehlberg methods (see section A.1-2), namely automatic step size control and minimum TEB. Sarafyan [47] lists four RK(6,8) with their embedding advantages (similar to his fifth order methods, see section A.1-5). Huta's RKE(6,8) is included in this reference. Some, rather obscure, advantages are claimed for these so called improved sixth order methods. One definite advantage is that all four are of the Newton-Cotes family and when $\frac{dY}{dT}$ is independent of Y they become eighth order in accuracy.

Fehlberg

A.62

	c										
	0										
	$\frac{2}{33}$	$\frac{2}{33}$									
	$\frac{4}{33}$	0	$\frac{4}{33}$								
	$\frac{2}{11}$	$\frac{1}{22}$	0	$\frac{3}{22}$							
	$\frac{1}{2}$	$\frac{43}{64}$	0	$-\frac{165}{64}$	$\frac{77}{32}$						
	$\frac{2}{3}$	$-\frac{2383}{486}$	0	$\frac{1067}{54}$	$-\frac{26412}{1701}$	$\frac{2176}{1701}$					
	$\frac{6}{7}$	$\frac{10077}{4802}$	0	$-\frac{5643}{686}$	$-\frac{116259}{16807}$	$-\frac{6240}{16807}$	$\frac{1053}{2401}$				
p=6	1	$-\frac{733}{176}$	0	$\frac{141}{8}$	$-\frac{335763}{23296}$	$\frac{216}{77}$	$-\frac{4617}{2816}$	$\frac{7203}{9152}$			
	0	$\frac{15}{352}$	0	0	$-\frac{5445}{46592}$	$\frac{18}{77}$	$-\frac{1215}{5632}$	$\frac{1029}{18304}$	0		
p=7	1	$-\frac{1833}{352}$	0	$\frac{141}{8}$	$-\frac{51237}{3584}$	$\frac{18}{7}$	$-\frac{729}{512}$	$\frac{1029}{1408}$	0	1	
p=6		$\frac{77}{1440}$	0	0	$\frac{1771561}{6289920}$	$\frac{32}{105}$	$\frac{243}{2560}$	$\frac{16807}{74880}$	$\frac{11}{270}$		
p=7		$\frac{11}{864}$	0	0	$\frac{1771561}{6289920}$	$\frac{32}{105}$	$\frac{243}{2560}$	$\frac{16807}{74880}$	0	$\frac{11}{270}$	$\frac{11}{270}$

$$Y_6(T_n + h) = Y_n + \frac{1}{840}\left[41(k_1+k_8) + 216(k_3+k_7) + 27(k_4+k_6) + 272k_5\right]$$

where

Huta

A.63

$$k_1 = hf(T_n, Y_n)$$

$$k_2 = hf(T_n + \tfrac{1}{9}h,\ Y_n + \tfrac{1}{9}k_1)$$

$$k_3 = hf(T_n + \tfrac{1}{6}h,\ Y_n + \tfrac{1}{24}(k_2 + 3k_3))$$

$$k_4 = hf(T_n + \tfrac{1}{3}h,\ Y_n + \tfrac{1}{6}(k_1 - 3k_2 + 4k_3))$$

$$k_5 = hf(T_n + \tfrac{1}{2}h,\ Y_n + \tfrac{1}{8}(-5k_1 + 27k_2 - 24k_3 + 6k_4))$$

$$k_6 = hf(T_n + \tfrac{2}{3}h,\ Y_n + \tfrac{1}{9}(221k_1 - 981k_2 + 867k_3 - 102k_4 + k_5))$$

$$k_7 = hf(T_n + \tfrac{5}{6}h,\ Y_n + \tfrac{1}{48}(-183k_1 + 678k_2 - 472k_3 - 66k_4 + 80k_5 + 3k_6))$$

$$k_8 = hf(T_n + h,\ Y_n + \tfrac{1}{82}(716k_1 - 2079k_2 + 1002k_3 + 834k_4 - 454k_5 - 9k_6 + 72k_7)).$$

$$Y_6(T_n + h) = Y_n + \frac{1}{840}\,[41(k_1+k_8) + 216(k_3+k_7) + 27(k_4+k_6) + 272k_5]$$

where

$$k_1 = hf(T_n, Y_n)$$

$$k_2 = hf(T_n + \tfrac{1}{9}h,\ Y_n + \tfrac{1}{9}\,k_1)$$

Sarafyan

$$k_3 = hf(T_n + \tfrac{1}{6}h,\ Y_n + \tfrac{1}{24}\,(k_1 + 3k_2))$$

A.64

$$k_4 = hf(T_n + \tfrac{1}{3}h,\ Y_n + \tfrac{1}{6}\,(k_1 - 3k_2 + 4k_3))$$

$$k_5 = hf(T_n + \tfrac{1}{2}h,\ Y_n + \tfrac{1}{8}\,(k_1 + 3k_4))$$

$$k_6 = hf(T_n + \tfrac{2}{3}h,\ Y_n + \tfrac{1}{3}\,(-4k_1 - 21k_2 + 46k_3 - 29k_4 + 10k_5))$$

$$k_7 = hf(T_n + \tfrac{5}{6}h,\ Y_n + \tfrac{1}{72}\,(-8k_1 + 99k_2 - 84k_3 + 44k_4 + 9k_6))$$

$$k_8 = hf(T_n + h,\ Y_n + \tfrac{1}{32}\,(107k_1 - 243k_2 + 354k_4 - 172k_5 - 36k_6 + 72k_7))$$

$$k_1 = hf(T_n, Y_n)$$

$$k_2 = hf(T_n + \tfrac{1}{9}h,\ Y_n + \tfrac{1}{9}k_1)$$

Sarafyan

$$k_3 = hf(T_n + \tfrac{1}{6}h,\ Y_n + \tfrac{1}{24}\,(k_1 + 3k_2))$$

A.65

$$k_4 = hf(T_n + \tfrac{1}{3}h,\ Y_n + \tfrac{1}{6}\,(k_1 - 3k_2 + 4k_3))$$

$$k_5 = hf(T_n + \tfrac{1}{2}h,\ Y_n + \tfrac{1}{8}\,(k_1 + 3k_4))$$

$$k_6 = hf(T_n + \tfrac{2}{3}h,\ Y_n + 2k_1 - 7k_2 + \tfrac{1}{3}\,(16k_3 + k_4))$$

$$k_7 = hf(T_n + \tfrac{5}{6}h,\ Y_n + \tfrac{1}{72}\,(-68k_1 + 99k_2 + 96k_3 - 180k_4 + 104k_5 + 9k_6))$$

$$k_8 = hf(T_n + h,\ Y_n + \tfrac{1}{82}\,(287k_1 - 243k_2 - 540k_3 + 894k_4 - 352k_5 - 36k_6 + 72k_7))$$

$$k_1 = hf(T_n, Y_n)$$

$$k_2 = hf(T_n + \tfrac{1}{9}h,\ Y_n + \tfrac{1}{9} k_1)$$

Sarafyan

$$k_3 = hf(T_n + \tfrac{1}{6}h,\ Y_n + \tfrac{1}{24}(k_1 + 3k_2))$$

A.66

$$k_4 = hf(T_n + \tfrac{1}{3}h,\ Y_n + \tfrac{1}{6}(k_1 - 3k_2 + 4k_3))$$

$$k_5 = hf(T_n + \tfrac{1}{2}h,\ Y_n + \tfrac{1}{8}(k_1 + 3k_4))$$

$$k_6 = hf(T_n + \tfrac{2}{3}h,\ Y_n + \tfrac{1}{9}(17k_1 - 63k_2 + 51k_3 + k_5))$$

$$k_7 = hf(T_n + \tfrac{5}{6}h,\ Y_n + \tfrac{1}{24}(-22k_1 + 33k_2 + 30k_3 - 58k_4 + 34k_5 + 3k_6))$$

$$k_8 = hf(T_n + h,\ Y_n + \tfrac{1}{82}(281k_1 - 243k_2 - 522k_3 + 876k_4 - 346k_5 - 36k_6 + 72k_7))$$

Shanks [27] developed an almost (6,6) method on the same lines as his RKE(5,5), eq. A.47.

$$\begin{array}{c|cccccc}
0 & & & & & & \\
\frac{1}{300} & \frac{1}{300} & & & & & \\
\frac{1}{5} & -\frac{29}{5} & \frac{30}{5} & & & & \\
\frac{3}{5} & \frac{323}{5} & -\frac{330}{5} & \frac{10}{5} & & & \\
\frac{14}{5} & -\frac{510104}{810} & \frac{521640}{810} & -\frac{12705}{810} & \frac{1925}{810} & & \\
1 & -\frac{417923}{77} & \frac{427350}{77} & -\frac{10605}{77} & \frac{1309}{77} & -\frac{54}{77} & \\
\hline
 & \frac{198}{3696} & 0 & \frac{1225}{3696} & \frac{1540}{3696} & \frac{810}{3696} & \frac{77}{3696}
\end{array}$$

Shanks

A.67

A.1-7 RKE(7,10):

Butcher [3], Fehlberg [4], and Rosen [5] list the deriving equations.

Sarafyan [48] gives four such formulae, of his usual embedded types (see section A.1-4), with built in step size control. His best form on accuracy is:

$$Y_1(T_n+h) = Y_n+k_1$$

$$Y_2(T_n+h) = Y_n + \frac{1}{8}(-55k_1+63k_2)$$

$$Y_7(T_n+h) = Y_n+\frac{1}{17280}[751(k_1+k_{10})+3577(k_4+k_9)+1323(k_5+k_8)+2989(k_6+k_7)]$$

$$k_1 = hf(T_n,Y_n)$$

$$k_2 = hf(T_n + \frac{4}{63}h,\ Y_n + \frac{4}{63}k_1)$$ Sarafyan

$$k_3 = hf(T_n + \frac{2}{21}h,\ Y_n + \frac{1}{42}(k_1+3k_2))$$ A.68

$$k_4 = hf(T_n + \frac{1}{7}h,\ Y_n + \frac{1}{28}(k_1+3k_3))$$

$$k_5 = hf(T_n + \frac{2}{7}h,\ Y_n + \frac{1}{7}(k_1-3k_3+4k_4))$$

$$k_6 = hf(T_n + \frac{3}{7}h,\ Y_n + \frac{3}{28}(k_1+3k_5))$$

$$k_7 = hf(T_n + \frac{4}{7}h,\ Y_n + \frac{2}{1827}(-101k_1+651k_4-477k_5+449k_6))$$

$$k_8 = hf(T_n + \frac{5}{7}h,\ Y_n + \frac{5}{65772}(-1881k_1-783k_3+10352k_4-3414k_5+5122k_7))$$

$$k_9 = hf(T_n + \frac{6}{7}h,\ Y_n + \frac{1}{2222850}(683663k_1+430650k_3-2032615k_4+2208930k_5 +385270k_6-740735k_7+970137k_8))$$

$$k_{10} = hf(T_n + h, Y_n + \frac{1}{19601100}(-12175421k_1-11236050k_3+62891430k_4 -43488585k_5-9947140k_6+51099720k_7-30879954k_8+13337100k_9)).$$

$$Y_1(T_n+h) = Y_n+k_1$$

$$Y_2(T_n+h) = Y_n + \frac{1}{2}(-k_1+3k_2)$$

$$Y_4(T_n+h) = Y_n + \frac{1}{6}(k_1+4k_3+k_4)$$

$$Y_7(T_n+h) = Y_n + \frac{1}{840}[41(k_1+k_{10})+216(k_5+k_9)+27(k_6+k_8)+272k_7]$$

$$k_1 = hf(T_n,Y_n)$$

$$k_2 = hf(T_n + \tfrac{1}{3}h,\ Y_n + \tfrac{1}{3}k_1)$$ Sarafyan

$$k_3 = hf(T_n + \tfrac{1}{2}h,\ Y_n + \tfrac{1}{8}[k_1+3k_2])$$ A.69

$$k_4 = hf(T_n+h,\ Y_n + \tfrac{1}{2}[k_1-3k_2+4k_3])$$

$$k_5 = hf(T_n + \tfrac{1}{6}h,\ Y_n + \tfrac{1}{648}[83k_1+32k_3-7k_4])$$

$$k_6 = hf(T_n + \tfrac{1}{3}h,\ Y_n + \tfrac{1}{54}[-3k_1-4k_3+k_4+24k_5])$$

$$k_7 = hf(T_n + \tfrac{1}{2}h,\ Y_n + \tfrac{1}{5088}[-290k_1-524k_3+145k_4+1908k_5+1305k_6])$$

$$k_8 = hf(T_n + \tfrac{2}{3}h,\ Y_n + \tfrac{1}{1431}[292k_1+108k_3+13k_4-318k_5+753k_6+106k_7])$$

$$k_9 = hf(T_n + \tfrac{5}{6}h,\ Y_n + \tfrac{1}{68688}[14042k_1+11012k_3-4477k_4+5724k_5-6903k_6 +6360k_7+31482k_8])$$

$$k_{10} = hf(T_n + h,\ Y_n + \tfrac{1}{4346}[-2049k_1-1836k_3+839k_4+5724k_5-4692k_6 +12084k_7-9540k_8+3816k_9]).$$

$$\begin{cases} Y_1(T_n+h) = Y_n+k_1 \\ Y_2(T_n+h) = Y_n + \frac{1}{2}(-25k_1+27k_2) \\ Y_7(T_n+h) = Y_n + \frac{1}{840}[41(k_1+k_{10})+216(k_5+k_9)+27(k_6+k_8)+272k_7] \end{cases}$$

$$\begin{cases} Y_1(T_n + \frac{1}{9}h) = Y_n + \frac{1}{9}k_1 \\ Y_2(T_n + \frac{1}{9}h) = Y_n + \frac{1}{18}(-k_1+3k_2) \\ Y_4(T_n + \frac{1}{9}h) = Y_n + \frac{1}{54}(k_1+4k_3+k_4) \end{cases}$$

where

$$\begin{cases} k_1 = hf(T_n,Y_n) \\ k_2 = hf(T_n + \frac{1}{27}h,\ Y_n + \frac{1}{27}k_1) \\ k_3 = hf(T_n + \frac{1}{18}h,\ Y_n + \frac{1}{72}(k_1+3k_2)) \\ k_4 = hf(T_n + \frac{1}{9}h,\ Y_n + \frac{1}{18}(k_1-3k_2+4k_3)) \\ k_5 = hf(T_n + \frac{1}{6}h,\ Y_n + \frac{1}{24}(k_1+3k_4)) \end{cases}$$

Sarafyan

A.70

and

$$\begin{cases} k_6 = hf(T_n + \frac{1}{3}h,\ Y_n + \frac{1}{18}(143k_1-420k_3+411k_4-128k_5)) \\ k_7 = hf(T_n + \frac{1}{2}h,\ Y_n + \frac{1}{1696}(-2494k_1+6876k_3-5013k_4+636k_5+843k_6)) \\ k_8 = hf(T_n + \frac{2}{3}h,\ Y_n + \frac{1}{159}(-2456k_1+7548k_3-7977k_4+3074k_5-189k_6+106k_7)) \\ k_9 = hf(T_n + \frac{5}{6}h,\ Y_n + \frac{1}{7632}(44946k_1-118428k_3+89043k_4-7844k_5-4431k_6 \\ \qquad -424k_7+3498k_8)) \\ k_{10} = hf(T_n + h,\ Y_n + \frac{1}{4346}(-67577k_1+169308k_3-106533k_4-3180k_5+4272k_6 \\ \qquad +13780k_7-9540k_8+3816k_9)) \end{cases}$$

or

$$\begin{cases} k_6 = hf(T_n + \frac{1}{3}h,\ Y_n + \frac{1}{54}(323k_1-942k_3+915k_4-278k_5)) \\ k_7 = hf(T_n + \frac{1}{2}h,\ Y_n + \frac{1}{1696}(-2070k_1+5604k_3-3741k_4+212k_5+843k_6)) \\ k_8 = hf(T_n + \frac{2}{3}h,\ Y_n + \frac{1}{1908}(-24490k_1+72132k_3-70125k_4+23108k_5 \\ \qquad -201k_6+848k_7)) \\ k_9 = hf(T_n + \frac{5}{6}h,\ Y_n + \frac{1}{11448}(60370k_1-151248k_3+96438k_4+8480k_5-9747k_6 \\ \qquad +5247k_8)) \\ k_{10} = hf(T_n + h,\ Y_n + \frac{1}{8692}(-125084k_1+297912k_3-150897k_4-42824k_5 \\ \qquad +14745k_6+26288k_7-19080k_8+7632k_9)). \end{cases}$$

$$\left[\begin{array}{l} Y_1(T_n+h) = Y_n+k_1 \\ Y_2(T_n+h) = Y_n-8k_1+9k_2 \\ Y_7(T_n+h) = Y_n+ [41(k_1+k_{10})+216(k_5+k_9)+27(k_6+k_8)+272k_7] \end{array}\right.$$

$$\left[\begin{array}{l} Y_1(T_n+ \frac{1}{6}h) = Y_n+ \frac{1}{6}k_1 \\ Y_2(T_n+ \frac{1}{6}h) = Y_n + \frac{1}{12}(-k_1+3k_2) \\ Y_4(T_n+ \frac{1}{6}h) = Y_n + \frac{1}{36}(k_1+4k_3+k_4) \end{array}\right.$$

where

$$\left[\begin{array}{l} k_1 = hf(T_n,Y_n) \\ k_2 = hf(T_n+ \frac{1}{18}h,\ Y_n + \frac{1}{18}k_1) \\ k_3 = hf(T_n + \frac{1}{12}h,\ Y_n + \frac{1}{48}(k_1+3k_2)) \\ k_4 = hf(T_n + \frac{1}{6}h,\ Y_n + \frac{1}{12}(k_1-3k_2+4k_3)) \\ k_5 = hf(T_n + \frac{1}{6}h,\ Y_n + \frac{1}{36}(k_1+4k_2+k_4)) \end{array}\right.$$

Sarafyan

A.71

and

$$\left[\begin{array}{l} k_6 = hf(T_n + \frac{1}{3}h,\ Y_n + \frac{1}{27}(6k_1-12k_3+221k_4-206k_5)) \\ k_7 = hf(T_n + \frac{1}{2}h,\ Y_n + \frac{1}{424}(-115k_1+448k_3-336k_4+215k_6)) \\ k_8 = hf(T_n + \frac{2}{3}h,\ Y_n + \frac{1}{477}(-171k_1+456k_3-8398k_4+8268k_5-49k_6+212k_7)) \\ k_9 = hf(T_n + \frac{5}{6}h,\ Y_n + \frac{1}{11448}(17367k_1-52248k_3+36218k_4+13144k_5-10188k_6+5247k_8)) \\ k_{10} = hf(T_n + h,\ Y_n + \frac{1}{2173}(-11407k_1+35012k_3-11631k_4-17490k_5+3979k_6+6572k_7-4770k_8+1908k_9)) \end{array}\right.$$

or

$$\left[\begin{array}{l} k_6 = hf(T_n + \frac{1}{3}h,\ Y_n + \frac{1}{9}(2k_1-4k_3+109k_4-104k_5)) \\ k_7 = hf(T_n + \frac{1}{2}h,\ Y_n + \frac{1}{424}(-115k_1+448k_3-548k_4+212k_5+215k_6)) \\ k_8 = hf(T_n + \frac{2}{3}h,\ Y_n + \frac{1}{477}(-171k_1+456k_3-12426k_4+12296k_5-49k_6+212k_7)) \\ k_9 = hf(T_n + \frac{5}{6}h,\ Y_n + \frac{1}{3816}(5789k_1-17416k_3+19846k_4-3392k_5-3396k_6+1749k_8)) \\ k_{10} = hf(T_n + h,\ Y_n + \frac{1}{2173}(-11407k_1+35012k_3-21277k_4-7844k_5+3979k_6+6572k_7-4770k_8+1908k_9)). \end{array}\right.$$

Fehlberg [4] lists a RKE[(7,11);(8,13)]; with the usual advantages of this form (discussed in section A.1-2): automatic step size control and a minimized TEB.

Fehlberg

A.72

	0														
	$\frac{2}{27}$	$\frac{2}{27}$													
	$\frac{1}{9}$	$\frac{1}{36}$	$\frac{1}{12}$												
	$\frac{1}{6}$	$\frac{1}{24}$	0	$\frac{1}{8}$											
	$\frac{5}{12}$	$\frac{5}{12}$	0	$-\frac{25}{16}$	$\frac{25}{16}$										
	$\frac{1}{2}$	$\frac{1}{20}$	0	0	$\frac{1}{4}$	$\frac{1}{5}$									
	$\frac{5}{6}$	$-\frac{25}{108}$	0	0	$\frac{125}{108}$	$-\frac{65}{27}$	$\frac{125}{54}$								
	$\frac{1}{6}$	$\frac{31}{300}$	0	0	0	$\frac{61}{225}$	$-\frac{2}{9}$	$\frac{13}{900}$							
	$\frac{2}{3}$	2	0	0	$-\frac{53}{6}$	$\frac{704}{45}$	$-\frac{107}{9}$	$\frac{67}{90}$	3						
	$\frac{1}{3}$	$-\frac{91}{108}$	0	0	$\frac{23}{108}$	$-\frac{976}{135}$	$\frac{311}{54}$	$-\frac{19}{60}$	$\frac{17}{6}$	$-\frac{1}{12}$					
p=7	1	$\frac{2383}{4100}$	0	0	$-\frac{341}{164}$	$\frac{4496}{1025}$	$-\frac{301}{82}$	$\frac{2133}{4100}$	$\frac{45}{82}$	$\frac{45}{164}$	$\frac{18}{41}$				
	0	$\frac{3}{205}$	0	0	0	0	$-\frac{6}{41}$	$-\frac{3}{205}$	$-\frac{3}{41}$	$\frac{3}{41}$	$\frac{6}{41}$	0			
p=8	1	$-\frac{1777}{4100}$	0	0	$-\frac{341}{164}$	$\frac{4496}{1025}$	$-\frac{289}{82}$	$\frac{2193}{4100}$	$\frac{51}{82}$	$\frac{33}{164}$	$\frac{12}{41}$	0	1		
		$\frac{41}{840}$										$\frac{41}{840}$			p=7
			0	0	0	0	$\frac{34}{105}$	$\frac{9}{35}$	$\frac{9}{35}$	$\frac{9}{280}$	$\frac{9}{280}$				
		0										0	$\frac{41}{840}$	$\frac{41}{840}$	p=8

Shanks [27] following the approach discussed in section A.1-5, lists a RKE(7,7) and a RKE(7,9). On a test system, as expected, the (7,9) was more accurate, while the (7,7) was faster.

0									
$\frac{1}{192}$	$\frac{1}{192}$								
$\frac{1}{6}$	$\frac{1}{6}$	(-15	16)						Shanks
$\frac{1}{2}$	$\frac{1}{186}$	(4867	-5072	298)					A.73
1	$\frac{1}{31}$	(-19995	20896	-1025	155)				
$\frac{5}{6}$	$\frac{1}{5022}$	(-469805	490960	-22736	5580	186)			
1	$\frac{1}{2604}$	(914314	-955136	47983	-6510	-558	2511)		
	$\frac{1}{300}$	(14	0	81	110	0	81	14)	

0											
$\frac{2}{9}$	$\frac{2}{9}$										
$\frac{1}{3}$	$\frac{1}{12}$	(1	3)						Shanks		
$\frac{1}{2}$	$\frac{1}{8}$	(1	0	3)					A.74		
$\frac{1}{6}$	$\frac{1}{216}$	(23	0	21	-8)						
$\frac{8}{9}$	$\frac{1}{729}$	(-4136	0	-13584	5264	13104)					
$\frac{1}{9}$	$\frac{1}{151632}$	(105131	0	302016	-107744	-284256	1701)				
$\frac{5}{6}$	$\frac{1}{1375920}$	(-775229	0	-2770950	1735136	2547216	81891	328536)			
1	$\frac{1}{251888}$	(23569	0	-122304	-20384	695520	-99873	-466560	241920)		
	$\frac{1}{2140320}$	(110201	0	0	767936	635040	-59049	-59049	635040	110201)	

A.1-8 RKE(8,11):

Butcher [3], Fehlberg [4] and Rosen [5] list the deriving equations.

Curtis [49] lists a RKE(8,11), listed here as eq. A.75. In eq. A.75 the 'b_i's' are the coefficients w_i's used in this study. Curtis leaves c_2, $w_9(b_9)$, and $w_{10}(b_{10})$ free; and recommends $c_2 = c_3$, $w_9(b_9) = 1/5$, and $w_{10}(b_{10}) = 13/80$.

$a_{2,1} = c_2$
c_2
$a_{3,1} = c_3 - a_{3,2}$
$a_{3,2} = +0.01690\,04862\,34/c_2$
$c_3 = +0.18385\,04078\,56$
$a_{4,1} = +0.06894\,39029\,46$
$a_{4,2} = 0$
$a_{4,3} = +0.20683\,17088\,39$
$c_4 = +0.27577\,56117\,85$
$a_{5,1} = +0.68943\,90294\,62$
$a_{5,2} = 0$
$a_{5,3} = -2.58539\,63604\,81$
$a_{5,4} = +2.58539\,63604\,81$
$c_5 = +0.68943\,90294\,62$
$a_{6,1} = +0.08273\,26835\,35$
$a_{6,2} = 0$
$a_{6,3} = 0$
$a_{6,4} = +0.41366\,34176\,77$
$a_{6,5} = +0.33093\,07341\,42$
$c_6 = +0.82732\,68353\,54$
$a_{7,1} = +0.09711\,55132\,62$
$a_{7,2} = 0$
$a_{7,3} = 0$
$a_{7,4} = +0.09730\,86865\,37$
$a_{7,5} = -0.04400\,58014\,67$
$a_{7,6} = +0.02225\,47766\,15$
$c_7 = +0.17267\,31646\,46$
$a_{8,1} = -0.06232\,01469\,88$
$a_{8,2} = 0$
$a_{8,3} = 0$
$a_{8,4} = -0.25994\,89002\,89$
$a_{8,5} = +0.24958\,29544\,83$
$a_{8,6} = -0.11345\,26333\,75$
$a_{8,7} = +0.68613\,87261\,68$
$c_8 = +0.5$
$a_{9,1} = a_{8,1} + 0.04509\,09166\,05/b_9$
$a_{9,2} = 0$
$a_{9,3} = 0$
$a_{9,4} = a_{8,4} + 0.02120\,84169\,40/b_9$
$a_{9,5} = a_{8,5} - 0.21856\,72981\,82/b_9$
$a_{9,6} = + 0.08397\,23935\,39/b_9$
$a_{9,7} = - 0.12592\,81463\,33/b_9$
$a_{9,8} = + 0.19422\,37174\,31/b_9$
$c_9 = +0.82732\,68353\,54$
$a_{10,1} = a_{7,1} + 0.00306\,07212\,81/b_{10}$
$a_{10,2} = 0$
$a_{10,3} = 0$
$a_{10,4} = a_{7,4} + 0.00143\,96037\,60/b_{10}$
$a_{10,5} = a_{7,5} - 0.01483\,61051\,69/b_{10}$
$a_{10,6} = a_{7,6} + 0.00569\,99527\,02/b_{10} - 0.04609\,85067\,65\,b_9/b_{10}$
$a_{10,7} = - 0.00854\,78625\,50/b_{10}$
$a_{10,8} = + 0.01318\,36899\,76/b_{10}$
$a_{10,9} = + 0.04609\,85067\,65\,b_9/b_{10}$
$c_{10} = +0.17267\,31646\,46$
$a_{11,1} = -0.49904\,07839\,34$
$a_{11,2} = 0$
$a_{11,3} = 0$
$a_{11,4} = -1.38639\,41348\,72$
$a_{11,5} = +1.33110\,90905\,75$
$a_{11,6} = -0.16772\,80299\,07 - 2.53149\,31576\,11\,b_9$
$a_{11,7} = +2.31464\,64507\,31 - 16.54653\,67070\,80\,b_{10}$
$a_{11,8} = - 0.59259\,25925\,93$
$a_{11,9} = + 2.53149\,31576\,11\,b_9$
$a_{11,10} = +16.54653\,67070\,80\,b_{10}$
$c_{11} = + 1.0$
$b_1 = + 0.05$
$b_2 = 0$
$b_3 = 0$
$b_4 = 0$
$b_5 = 0$
$b_6 = + 0.27222\,22222\,22 - b_9$
$b_7 = + 0.27222\,22222\,22 - b_{10}$
$b_8 = + 0.35555\,55555\,56$
b_9
b_{10}
$b_{11} = + 0.05$

Curtis

A.75

Fehlberg [4] lists a RKE[(8,15);(9,17)] with his usual advantage of automatic step-size control and minimum TEB (see section A.1-2). Fourty digit arithmetic is used for the coefficients of this method. This is listed here as eq. A.76, where the 's_i's', B_{ij}'s', and 'c_i's' are the same as the coefficients, c_i, a_{ij} and w_i respectively, used in this study. Also in eq. A.76 i and j take values from 0 to 16 while the formulation used in this study avoids i,j=0. Hence β_{10} of eq. A.76 is the usual a_{21}, c_0 is w_1 and so on. Naturally $\alpha_0(c_1) = 0$ in eq. A.76. A formula for calculating the truncation error is also listed.

$\alpha_1 = 0.4436\ 8940\ 3764\ 9818\ 3109\ 5994\ 0428\ 1370$
$\alpha_2 = 0.6655\ 3410\ 5647\ 4727\ 4664\ 3991\ 0642\ 2055$
$\alpha_3 = 0.9983\ 0115\ 8471\ 2091\ 1996\ 5986\ 5963\ 3083$
$\alpha_4 = 0.3155\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000$
$\alpha_5 = 0.5054\ 4100\ 9481\ 6906\ 8626\ 5161\ 2673\ 7384$
$\alpha_6 = 0.1714\ 2857\ 1428\ 5714\ 2857\ 1428\ 5714\ 2857$
$\alpha_7 = 0.8285\ 7142\ 8571\ 4285\ 7142\ 8571\ 4285\ 7143$
$\alpha_8 = 0.6654\ 3966\ 1210\ 1156\ 2534\ 9537\ 6925\ 5586$
$\alpha_9 = 0.2487\ 8317\ 9680\ 6265\ 2069\ 7222\ 7456\ 0771$
$\alpha_{10} = 0.1090\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000$
$\alpha_{11} = 0.8910\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000$
$\alpha_{12} = 0.3995\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000$
$\alpha_{13} = 0.6005\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000$
$\alpha_{14} = 1$
$\alpha_{15} = 0$
$\alpha_{16} = 1$

Fehlberg

A.76

$\beta_{10} = 0.4436\ 8940\ 3764\ 9818\ 3109\ 5994\ 0428\ 1370$
$\beta_{20} = 0.1663\ 8352\ 6411\ 8681\ 8666\ 0997\ 7660\ 5514$
$\beta_{21} = 0.4991\ 5057\ 9235\ 6045\ 5998\ 2993\ 2981\ 6541$
$\beta_{30} = 0.2495\ 7528\ 9617\ 8022\ 7999\ 1496\ 6490\ 8271$
$\beta_{32} = 0.7487\ 2586\ 8853\ 4068\ 3997\ 4489\ 9472\ 4812$
$\beta_{40} = 0.2066\ 1891\ 1634\ 0060\ 2426\ 5567\ 1039\ 3185$
$\beta_{42} = 0.1770\ 7880\ 3779\ 8634\ 7040\ 3809\ 9728\ 8319$
$\beta_{43} = -0.6819\ 7715\ 4138\ 6949\ 4669\ 3770\ 7681\ 5048 \cdot 10^{-1}$
$\beta_{50} = 0.1092\ 7823\ 1526\ 6640\ 8227\ 9038\ 9092\ 6157$
$\beta_{53} = 0.4021\ 5962\ 6423\ 6799\ 5421\ 9905\ 6369\ 0087 \cdot 10^{-2}$
$\beta_{54} = 0.3921\ 4118\ 1690\ 7898\ 0444\ 3923\ 3017\ 4325$
$\beta_{60} = 0.9889\ 9281\ 4091\ 6466\ 5304\ 8447\ 6543\ 4355 \cdot 10^{-1}$
$\beta_{63} = 0.3513\ 8370\ 2279\ 6396\ 6951\ 2044\ 8735\ 6703 \cdot 10^{-2}$
$\beta_{64} = 0.1247\ 6099\ 9831\ 6001\ 6621\ 5206\ 2587\ 2489$
$\beta_{65} = -0.5574\ 5546\ 8349\ 8979\ 9643\ 7429\ 0146\ 6348 \cdot 10^{-1}$
$\beta_{70} = -0.3680\ 6865\ 2862\ 4220\ 3724\ 1531\ 0108\ 0691$
$\beta_{74} = -0.2227\ 3897\ 4694\ 7600\ 7645\ 0240\ 2094\ 4166 \cdot 10^{+1}$
$\beta_{75} = 0.1374\ 2908\ 2567\ 0291\ 0729\ 5656\ 9124\ 5744 \cdot 10^{+1}$
$\beta_{76} = 0.2049\ 7390\ 0271\ 1160\ 3002\ 1593\ 5409\ 2206 \cdot 10^{+1}$
$\beta_{80} = 0.4546\ 7962\ 6413\ 4715\ 0077\ 3519\ 5060\ 3349 \cdot 10^{-1}$

$\beta_{146} = -0.6557\ 0189\ 4497\ 4164\ 5138\ 0068\ 7998\ 5251$
$\beta_{147} = -0.3908\ 6144\ 8804\ 3986\ 3435\ 0255\ 2024\ 1310$
$\beta_{148} = 0.2679\ 4646\ 7128\ 5002\ 2936\ 5844\ 2327\ 1209$
$\beta_{149} = -0.1038\ 3022\ 9913\ 8249\ 0865\ 7698\ 5850\ 7427 \cdot 10^{+1}$
$\beta_{1410} = 0.1667\ 2327\ 3242\ 5867\ 1664\ 7273\ 4616\ 8501 \cdot 10^{+1}$
$\beta_{1411} = 0.4955\ 1925\ 8553\ 1597\ 7067\ 7329\ 6707\ 1441$
$\beta_{1412} = 0.1139\ 4001\ 1323\ 9706\ 3228\ 5867\ 3814\ 1784 \cdot 10^{+1}$
$\beta_{1413} = 0.5133\ 6696\ 4246\ 5861\ 3688\ 1990\ 9719\ 1534 \cdot 10^{-1}$
$\beta_{150} = 0.1046\ 4847\ 3406\ 1481\ 0391\ 8730\ 0240\ 6755 \cdot 10^{-2}$
$\beta_{158} = -0.6716\ 3886\ 8449\ 9028\ 2237\ 7784\ 4617\ 8020 \cdot 10^{-2}$
$\beta_{159} = 0.8182\ 8762\ 1894\ 2502\ 1265\ 3300\ 6524\ 8999 \cdot 10^{-2}$
$\beta_{1510} = -0.4264\ 0342\ 8644\ 8334\ 7277\ 1421\ 3808\ 7561 \cdot 10^{-2}$
$\beta_{1511} = 0.2800\ 9029\ 4741\ 6893\ 6545\ 9763\ 3115\ 3703 \cdot 10^{-3}$
$\beta_{1512} = -0.8783\ 5333\ 8762\ 3867\ 6639\ 0578\ 1314\ 5633 \cdot 10^{-3}$
$\beta_{1513} = 0.1025\ 4505\ 1108\ 2555\ 8084\ 2177\ 6966\ 4009 \cdot 10^{-1}$
$\beta_{160} = -0.1353\ 6550\ 7861\ 7406\ 7080\ 4421\ 6888\ 9966 \cdot 10^{+1}$
$\beta_{165} = -0.1839\ 6103\ 1448\ 4827\ 0375\ 0441\ 9898\ 8231$
$\beta_{166} = -0.6557\ 0189\ 4497\ 4164\ 5138\ 0068\ 7998\ 5251$
$\beta_{167} = -0.3908\ 6144\ 8804\ 3986\ 3435\ 0255\ 2024\ 1310$
$\beta_{168} = 0.2746\ 6285\ 5812\ 9992\ 5758\ 9622\ 0773\ 2989$
$\beta_{169} = -0.1046\ 4851\ 7535\ 7191\ 5887\ 0351\ 8857\ 2676 \cdot 10^{+1}$
$\beta_{1610} = 0.1671\ 4967\ 6671\ 2315\ 5012\ 0044\ 8830\ 6588 \cdot 10^{+1}$
$\beta_{1611} = 0.4952\ 3916\ 8258\ 4180\ 8131\ 1869\ 9074\ 0287$
$\beta_{1612} = 0.1148\ 1836\ 4662\ 7330\ 1905\ 2257\ 9595\ 4930 \cdot 10^{+1}$
$\beta_{1613} = 0.4108\ 2191\ 3138\ 3305\ 5603\ 9813\ 2752\ 7525 \cdot 10^{-1}$
$\beta_{1615} = 1$

Fehlberg

A.76 contd.

$c_0 = 0.3225\ 6083\ 5002\ 1624\ 9913\ 6129\ 0096\ 0247 \cdot 10^{-1}$
$c_8 = 0.2598\ 3725\ 2837\ 1540\ 3018\ 8870\ 2317\ 1963$
$c_9 = 0.9284\ 7805\ 9965\ 7702\ 7788\ 0637\ 1430\ 2190 \cdot 10^{-1}$
$c_{10} = 0.1645\ 2339\ 5147\ 6434\ 2891\ 6477\ 3184\ 2800$
$c_{11} = 0.1766\ 5951\ 6378\ 6007\ 4367\ 0842\ 9839\ 7547$
$c_{12} = 0.2392\ 0102\ 3203\ 5275\ 9374\ 1089\ 3332\ 0941$
$c_{13} = 0.3948\ 4274\ 6042\ 0285\ 3746\ 7521\ 1882\ 9325 \cdot 10^{-2}$
$c_{14} = 0.3072\ 6495\ 4758\ 6064\ 0406\ 3683\ 0552\ 2124 \cdot 10^{-1}$

$TE = c_{14}(f_0 + f_{14} - f_{15} - f_{16})h$

$\beta_{85} = 0.3254\ 2131\ 7015\ 8914\ 7114\ 6774\ 6964\ 8853$

$\beta_{86} = 0.2847\ 6660\ 1385\ 2790\ 8888\ 1824\ 2057\ 3687$

$\beta_{87} = 0.9783\ 7801\ 6759\ 7915\ 2435\ 8683\ 9727\ 1099 \cdot 10^{-2}$

$\beta_{90} = 0.6084\ 2071\ 0626\ 2205\ 7051\ 0941\ 4520\ 5182 \cdot 10^{-1}$

$\beta_{95} = -0.2118\ 4565\ 7440\ 3700\ 7526\ 3252\ 7525\ 1206 \cdot 10^{-1}$

$\beta_{96} = 0.1959\ 6557\ 2661\ 7083\ 1957\ 4644\ 9066\ 2983$

$\beta_{97} = -0.4274\ 2640\ 3648\ 1760\ 3675\ 1448\ 3534\ 2899 \cdot 10^{-2}$

$\beta_{98} = 0.1743\ 4365\ 7368\ 1491\ 1955\ 3234\ 5255\ 8189 \cdot 10^{-1}$

$\beta_{100} = 0.5405\ 9783\ 2959\ 3191\ 7365\ 7857\ 2411\ 1182 \cdot 10^{-1}$

$\beta_{105} = 0.1102\ 9325\ 5978\ 2892\ 6530\ 2831\ 2764\ 8228$

$\beta_{107} = -0.1256\ 5008\ 5200\ 7255\ 6414\ 1477\ 6378\ 2250 \cdot 10^{-2}$

$\beta_{108} = 0.3679\ 0043\ 4775\ 8146\ 0136\ 3840\ 4356\ 6339 \cdot 10^{-2}$

$\beta_{109} = -0.5778\ 0542\ 7703\ 7207\ 3940\ 8406\ 2857\ 1866 \cdot 10^{-1}$

$\beta_{110} = 0.1273\ 2477\ 0686\ 6711\ 4646\ 6451\ 8179\ 9160$

$\beta_{117} = 0.1144\ 8305\ 0053\ 9310\ 5323\ 6588\ 7572\ 1817$

$\beta_{118} = 0.2877\ 3020\ 7036\ 9799\ 2776\ 2022\ 0184\ 9198$

$\beta_{119} = 0.5094\ 5379\ 4596\ 1136\ 3153\ 7358\ 8507\ 9465$

$\beta_{1110} = -0.1479\ 9682\ 2443\ 7257\ 5900\ 2421\ 4444\ 9640$

$\beta_{120} = -0.3652\ 6793\ 8766\ 1674\ 0535\ 8485\ 4439\ 4333 \cdot 10^{-2}$

$\beta_{125} = 0.8162\ 9896\ 0123\ 1891\ 9777\ 8194\ 2124\ 7030 \cdot 10^{-1}$

$\beta_{126} = -0.3860\ 7735\ 6356\ 9350\ 6490\ 5176\ 9434\ 3215$

$\beta_{127} = 0.3086\ 2242\ 9246\ 0510\ 6450\ 4741\ 6602\ 5206 \cdot 10^{-1}$

$\beta_{128} = -0.5807\ 7254\ 5283\ 2060\ 2815\ 8293\ 7473\ 3518 \cdot 10^{-1}$

$\beta_{129} = 0.3359\ 8659\ 3288\ 8497\ 1493\ 1434\ 5136\ 2322$

$\beta_{1210} = 0.4106\ 6880\ 4019\ 4995\ 8613\ 5496\ 2278\ 6417$

$\beta_{1211} = -0.1184\ 0245\ 9723\ 5598\ 5520\ 6331\ 5615\ 4536 \cdot 10^{-1}$

$\beta_{130} = -0.1237\ 5357\ 9212\ 4514\ 3254\ 9790\ 9613\ 5669 \cdot 10^{+1}$

$\beta_{135} = -0.2443\ 0768\ 5513\ 5478\ 5358\ 7348\ 6136\ 6763 \cdot 10^{+2}$

$\beta_{136} = 0.5477\ 9568\ 9327\ 7865\ 6050\ 4365\ 2899\ 1173$

$\beta_{137} = -0.4441\ 3863\ 5334\ 1324\ 6374\ 9598\ 9656\ 9346 \cdot 10^{+1}$

$\beta_{138} = 0.1001\ 3104\ 8137\ 1326\ 6094\ 7926\ 1785\ 1022 \cdot 10^{+2}$

$\beta_{139} = -0.1499\ 5773\ 1020\ 5175\ 8447\ 1709\ 8507\ 3142 \cdot 10^{+2}$

$\beta_{1310} = 0.5894\ 6948\ 5232\ 1701\ 3620\ 8245\ 3965\ 1427 \cdot 10^{+1}$

$\beta_{1311} = 0.1738\ 0377\ 5034\ 2898\ 4877\ 6168\ 5744\ 0542 \cdot 10^{+1}$

$\beta_{1312} = 0.2751\ 2330\ 6931\ 6673\ 0263\ 7586\ 2286\ 0276 \cdot 10^{+2}$

$\beta_{140} = -0.3526\ 0859\ 3883\ 3452\ 2700\ 5029\ 5887\ 5588$

$\beta_{146} = -0.1839\ 6103\ 1448\ 4827\ 0375\ 0441\ 9898\ 8231$

Fehlberg

A.76 contd.

Shanks [27] has listed two 'almost' eighth order formulae (see section A.1-5), one a RKE(8,10), and the other a RKE(8,12). As expected the (8,10) is faster, while the (8,12) is more accurate (not much more accurate, as his own tests prove).

Shanks

A.77

0											
$\frac{4}{27}$		$\frac{4}{27}$									
$\frac{2}{9}$	$\frac{1}{18}$	(1	3)								
$\frac{1}{3}$	$\frac{1}{12}$	(1	0	3)							
$\frac{1}{2}$	$\frac{1}{8}$	(1	0	0	3)						
$\frac{2}{3}$	$\frac{1}{54}$	(13	0	-27	42	8)					
$\frac{1}{6}$	$\frac{1}{4320}$	(389	0	-54	966	-824	243)				
1	$\frac{1}{20}$	(-231	0	81	-1164	656	-122	800)			
$\frac{5}{6}$	$\frac{1}{288}$	(-127	0	18	-678	456	-9	576	4)		
1	$\frac{1}{820}$	(1481	0	-81	7104	-3376	72	-5040	-60	720)	
	$\frac{1}{840}$	(41	0	0	27	272	27	216	0	216	41)

Shanks

A.78

0													
$\frac{1}{9}$		$\frac{1}{9}$											
$\frac{1}{6}$	$\frac{1}{24}$	(1	3)										
$\frac{1}{4}$	$\frac{1}{16}$	(1	0	3)									
$\frac{1}{10}$	$\frac{1}{500}$	(29	0	33	-12)								
$\frac{1}{6}$	$\frac{1}{972}$	(33	0	0	4	125)							
$\frac{1}{2}$	$\frac{1}{36}$	(-21	0	0	76	125	-162)						
$\frac{2}{3}$	$\frac{1}{243}$	(-30	0	0	-32	125	0	99)					
$\frac{1}{3}$	$\frac{1}{324}$	(1175	0	0	-3456	-6250	8424	242	-27)				
$\frac{5}{6}$	$\frac{1}{324}$	(293	0	0	-852	-1375	1836	-118	162	324)			
$\frac{5}{6}$	$\frac{1}{1620}$	(1303	0	0	-4260	-6875	9990	1030	0	0	162)		
1	$\frac{1}{4428}$	(-8595	0	0	30720	48750	-66096	378	-729	-1944	-1296	3240)	
	$\frac{1}{840}$	(41	0	0	0	0	216	272	27	27	36	180	41)

Increasing use of higher precision arithmetic is required, to realize the advantages of this order over the seventh order (slim advantage in accuracy)--paying the price of increased computational time.

A.1-9 RKE's of Ninth and Higher Order

The deriving equations for ninth and higher orders are listed, but not solved in Rosen [5].

Shanks is supposed to have developed (not yet published) ninth and tenth order RKE's.

There is no real reason for pushing on for orders beyond the eighth (except for special problems). According to Fehlberg [4] accuracy hardly improves after the seventh order methods.

Sarafyan is working on a computer formulation and solution, of the deriving equations leading to high order methods (see Sarafyan and Brown [7]); however the results are yet to be published.

A.2 Semi-Implicit Runge-Kutta Methods - RKSI(p,V):

It is easier to use the autonomous form of an ordinary differential equation system when dealing with these types of methods.

$$\frac{d\underline{Y}}{dT} = F(T,\underline{Y}) \qquad \text{A.79}$$

The explicit dependence of T in F(T,Y) can be removed by making
n+1
T the (n+1)th component of the n component vector $\underline{Y}$. This makes the system n+1 dimensional rather than n dimensional.

let $\quad Y_{n+1} = T$

$$\therefore \frac{dY_{n+1}}{dT} = 1$$

$$\therefore \frac{d\underline{Y}}{dT} = \underline{F}(\underline{Y}) \qquad \text{A.80}$$

where:

$$\frac{d\underline{Y}}{dT} = \begin{pmatrix} Y_1 \\ \cdot \\ \cdot \\ \cdot \\ Y_i \\ \cdot \\ \cdot \\ \cdot \\ Y_n \\ Y_{n+1} \end{pmatrix} \text{ and } F(\underline{Y}) = \begin{pmatrix} F_1(Y_1 \cdots Y_i \cdots Y_n Y_{n+1}) \\ \cdot \\ \cdot \\ \cdot \\ F_i(Y_1 \cdots Y_i \cdots Y_n Y_{n+1}) \\ \cdot \\ \cdot \\ \cdot \\ F_n(Y_1 \cdots Y_i \cdots Y_n Y_{n+1}) \\ 1 \end{pmatrix}$$

The initial conditions would also be accordingly altered, i.e., $Y_{n+1}(T_0) = T_0$. There is no need however to solve the n+1 component equation of eq. A.80 as its solution is always $Y_{n+1} = T$.

When a Runge-Kutta method is used on this autonomous form; the increments coefficients of the abcissa, i.e., the c's, become a's, i.e., the increment coefficients of the vector component Y_{n+1}.

In Chapter 2, section 2.2 the method of solving for each k_i of the semi-implicit form was discussed; this involved a Jacobian inversion. Restating eq. 2.33 in matrix form to apply to a system of equations, with the a_{ij}'s involved in the Jacobian written as a'_{ij}'a.

$$\underline{k}_i = [\underline{I} - h\, a_{ii}\, \underline{F}_y(\underline{Y}_n + \sum_{j=1}^{i-1} a'_{ij}\, \underline{k}_j]^{-1} \cdot h\underline{F}(\underline{Y}_n + \sum_{j=1}^{i-1} a_{ij}\, \underline{k}_j) \qquad \text{A.81}$$

$$i = 1,2,\text{---}V$$

The RKSI type method used to solve eq. A.80 is:

$$\underline{Y}_{n+1} = \underline{Y}_n + \sum_{i=1}^{V} w_i \underline{k}_i$$
$$\underline{k}_i = h\underline{F}(\underline{Y}_n + \sum_{j=1}^{i} a_{ij}\, \underline{k}_j) \qquad \text{A.82}$$

To solve for each $\underline{k}_i$, the formulation of eq. A.81 is used. The coefficient matrix is of the form:

$$\begin{array}{|llll}
a_{11} & & & \\
a_{21} & a_{22} & & \\
\vdots & & & \\
a_{i1} & a_{i2} & \cdots & a_{ii} \\
\vdots & & & \\
a_{v1} & a_{v2} & \cdots\cdots & a_{vv} \\
\hline
w_1 & w_2 \cdots w_i & \cdots & w_v
\end{array}
\qquad
\begin{array}{|lll}
a'_{21} & & \\
\vdots & & \\
a'_{i1} & a'_{i2} \cdots & a'_{ii-1} \\
\vdots & & \\
a'_{v1} & a'_{v2} \cdots & a'_{vv-1} \\
\hline
\end{array}$$

Allen [50], Rosenbrock [51] and Haines [52] list the deriving equations. Allen [50] gives a thorough discussion of the deriving equations and the stability advantages of RKSI's over RKE's; and also gives coefficients of the truncation errors. In general RKSI's require less stages, V per step as compared to RKE's, to achieve a p order method. The price paid by RKSI's, for achieving extended stability as compared to RKE;s, is the additional computational effort required to compute Jacobians and their inverses; but this price is not as steep as the one paid by RKI's, where iteration for k_i's is required.

Lapadis and Seinfeld [6] recommend the use of RKSI only for those cases where stability is critical, for example, with still systems; because in general RKSI's are not significantly more accurate than comparable order RKE's; and RKSI's usually require more computational time as compared to RKE's.

The following sections are titled RKSI(p) instead of RKSI(p,V).

A.2-1 RKSI(1):

Allen [50] states that no such case of interest exists.

A.2-2 RKSI(2):

Allen [50] and Rosenbrock [51] list the deriving equations and solve them. Allen [50] gives a V=2 method while Rosenbrock gives a V=2 process.

$$\begin{array}{c|c} & 1/2 \\ \hline & 1 \end{array} \qquad\qquad \begin{array}{|c} 0 \\ \hline \end{array} \qquad\qquad \begin{array}{l} \text{Allen} \\ \text{A.83} \end{array}$$

$$\begin{array}{c|cc} & 1 - \frac{\sqrt{2}}{2} & \\ & \frac{(\sqrt{2} - 1)}{2} & 1 - \frac{\sqrt{2}}{2} \\ \hline & 0 & 1 \end{array} \qquad\qquad \begin{array}{|c} 0 \\ \hline \end{array} \qquad\qquad \begin{array}{l} \text{Rosenbrock} \\ \text{A.84} \end{array}$$

Allen [50] however admits that his method, eq. A.83 does not possess significant stability advantages over a second order RKE.

A.2-3 RKSI(3):

The deriving equations are listed and solved in Allen [50], Rosenbrock [51] and Haines [52].

Allen [50] lists three methods with V=2:

$$\begin{array}{c|cc} & 0 & \\ & 0 & 1/3 \\ \hline & -1/2 & 3/2 \end{array} \qquad\qquad \begin{array}{|c} 1/3 \\ \hline \end{array} \qquad\qquad \begin{array}{l} \text{Allen} \\ \text{A.85} \end{array}$$

$$
\begin{array}{c|cc}
 & .211325 & \\
 & 1.154701 & .211325 \\
\hline
 & .75 & .25
\end{array}
\qquad
\begin{array}{|c}
0 \\
\hline
\end{array}
\qquad
\begin{array}{l}
\text{Allen} \\
\text{A.86}
\end{array}
$$

$$
\begin{array}{c|cc}
 & .788675 & \\
 & -1.154701 & .788675 \\
\hline
 & .75 & .25
\end{array}
\qquad
\begin{array}{|c}
0 \\
\hline
\end{array}
\qquad
\begin{array}{l}
\text{Allen} \\
\text{A.87}
\end{array}
$$

Eq. A.87 is the best such method as judged by various tests, conducted in Allen [50].

Rosenbrock [51] lists a method with V=2:

$$
\begin{array}{c|cc}
 & 1.40824829 & \\
 & .17378667 & .59175171 \\
\hline
 & -.41315432 & 1.41315432
\end{array}
\qquad
\begin{array}{|c}
.17378667 \\
\hline
\end{array}
\qquad
\begin{array}{l}
\text{Rosenbrock} \\
\text{A.88}
\end{array}
$$

Calahan [53] also lists a V=2 method, which is the same as Allen's eq. A.87.

Haines [52] derives a V=4 method, the extra stages per step being used to give a truncation error estimate and hence allow step size control. This method consists of two equations, each with V=4 stages. These two equations are compared to give a truncation error estimate. Both these equations have mostly common coefficients. This method is listed in two parts because each part can be used independently to give a third order method, or in combination to give an error estimate.

1								
1	1				1			Haines
1/2	1/2	1			1/2	1/2		A.89
2/99	95/99	2/99	2/3		0	0	0	
19/9	-43/18	28/9	-11/6					

1								
1	1				1			Haines
1/2	1/2	1			1/2	1/2		A.90
21/44	19/44	1/11	1/2		0	0	0	
10/3	-1	7/3	-11/3					

$\text{Error}_n = (29/9)[(Y_n \text{ with eq. A.89}) - (Y_n \text{ with eq. A.90})]$

Haines [52] also compares his method, which combines eqs. A.89 and A.90 to give step size control, with a RKE(4,4) which uses a similar step size control. His results point out that his method is superior on basis of computing time and stability.

<u>A.2-4</u> <u>RKSI(4)</u>:

Allen [50] lists thirth fourth order methods with V=3. Three such methods are presented here, these three were judged in Allen [50], and found to be superior to the rest of his methods on some points.

0						
-.31447015	.16666667			.4444779		Allen (minimum computational effort)
11.8533114	-9.9088335	.16666667		.4444779	0	
3.51324302	-2.57014293	.05689991				A.91

$$\begin{array}{r|lll|ll}
 & .28915127 & & & & \\
 & .57118895 & .28915127 & & 0 & \\
 & .53873135 & -.024693209 & .33215386 & 0 & 0 \\
\hline
 & .72862466 & 4.21831023 & -3.94693489 & &
\end{array}$$

Allen (largest region of stability)

A.92

$$\begin{array}{r|lll|ll}
 & .86234511 & & & & \\
 & -1.28325229 & 1.23119918 & & 0 & \\
 & -.80332019 & -.091154713 & 1.23119818 & 0 & 0 \\
\hline
 & -.32266573 & -.85621119 & 2.17887692 & &
\end{array}$$

Allen (best for numerical stability)

A.93

Butcher [2] has listed a RKSI(4,3); his method does not apply to the autonomous form of the differential equations, but to the usual form used throughout this study. (The method could easily be converted to apply to an autonomous form by simply removing the c_i's.)

$$\begin{array}{c|ccc}
0 & 0 & & \\
1/2 & 1/4 & 1/4 & \\
1 & 0 & 1 & 0 \\
\hline
 & 1/6 & 2/3 & 1/6
\end{array}$$

Butcher

A.94

A.3 Implicit Runge-Kutta Methods - RKI(p,V):

Restating the equations for a RKI(p,V):

$$Y_{n+1} = Y_n + \sum_{i=1}^{V} w_i k_i$$

$$k_i = hf(T_n + c_i h,\ Y_n + \sum_{j=1}^{V} a_{ij} k_j) \qquad \text{A.95}$$

The coefficient matrix for these types of methods is full; and hence each k_i has to be solved for iteratively at each integration step.

The use of these RKI's is recommended for two situations: one where stability is critical and a self-starting process is desired; two when the differential equation system is such that the cost of computing f(T,Y) for a given value of T is high compared to the cost of repeating this computation with Y changed but T unchanged. In the second situation the iterative nature of an RKI would not be an objection;especially if the differential equation system is linear, then the iterations can be replaced by standard linear algebra techniques.

Advantages: Stability characteristics better than comparable order RKSI's or RKE's. Always stable for any h. Higher p for a particular V as compared to RKSI's and RKE's. The deriving equations are easier to formulate than those of RKSI's and RKE's.

Disadvantages: For even moderately complex differential equation systems the iterative solution of k_i's requires more computational effort than comparable order semi-implicit or explicit methods.

The coefficient matrix will be presented in the following form:

$$
\begin{array}{c|ccccc}
c_1 & a_{11} & \cdots & a_{1j} & \cdots & a_{1v} \\
\vdots & \vdots & & \vdots & & \vdots \\
c_i & a_{i1} & \cdots & a_{ij} & \cdots & a_{iv} \\
\vdots & \vdots & & \vdots & & \vdots \\
c_v & a_{v1} & \cdots & a_{vj} & \cdots & a_{vv} \\
\hline
 & w_1 & \cdots & w_j & \cdots & w_v
\end{array}
$$

Most currently available implicit processes were derived by Butcher. The theory behind implicit methods, attainable order, etc., is dealt in Butcher [1] and Verner [54]. Deriving equations and implicit methods are listed in Butcher [2], Butcher [3] and Butcher [55]. Lapidus and Seinfeld [6] also list some of these Butcher methods.

Implicit Runge-Kutta methods are based on quadrative formulae (The Taylor Series analysis can also be used to derive implicit methods but the quadrative approach is simpler). Basically Implicit Runge-Kutta's can be divided into three classes of methods: Gauss-Legendre quadrative forms, Radau quadrative forms, and Lobatto quadrative forms.

Quadrative Form	Abcissa points (c_i's) specified	V	p
Gauss-Legendre	All c_i's found as roots of a Legendre polynomial, no c_i specified	V	2V
Radau_I	$c_1 = 0$ specified	V	(2V-1)
Radau_{II}	$c_V = 1$ specified	V	(2V-1)
Lobatto	$c_1 = 0$ and $c_V = 1$ specified	V	(2V-2)

$$\text{when } c_1 = 0 \quad a_{11} = a_{12} = a_{13} = \cdots\cdots = a_{1V} = 0$$

$$\text{when } c_V = 1 \quad a_{1V} = a_{2V} = a_{3V} = \cdots\cdots = a_{VV} = 0$$

Hence the Lobatto forms have the first row and last column of the coefficient matrix as zeros. Thus as $c_1 = 0$ and $c_V = 1$ were specified two k_i's, k_1 and k_V are available explicitly and so can be solved for by iteration. The Radau_I and Radau_{II} forms have only one k_i available explicitly as only one c_i is specified. While the Gauss-Legendre forms have no k_i available explicitly as no c_i is specified. Because of the number of abcissa points specified, the Gauss-Legendre forms have the highest p for a given V, while Radau forms are next and Lobatto forms last.

Butcher [2] also suggests a convergent iterative procedure applicable for calculating the implicit k_i's of any RKI.

If n is the iteration count for a particular k_i the convergent iterative procedure is:

$$k_i^{(n)} = hj(T_n + c_i h, \; Y_n + \sum_{j=1}^{i-1} a_{ij} k_j^{(n)} + \sum_{j=i}^{V} a_{ij} k_j^{(n-1)})$$

Butcher proves that this procedure is convergent.

A.3-1 RKI(p,V)'s of the Gauss-Legendre Quadrative forms:

These kinds of RKI's are listed in Butcher [2] from p = 2 to p = 10 (p = 2V), an error analysis is also included.

$$\begin{array}{c|c} \frac{1}{2} & \frac{1}{2} \\ \hline & 1 \end{array}$$

RKI(2,1)

A.96

$$\begin{array}{cc|c} \frac{1}{2} - \frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4} - \frac{\sqrt{3}}{6} \\ \frac{1}{2} + \frac{\sqrt{3}}{6} & \frac{1}{4} + \frac{\sqrt{3}}{6} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

RKI(4,2)

A.97

RKI(6,3)

A.98

$$
\begin{array}{c|ccc}
\frac{1}{2} - \frac{\sqrt{15}}{10} & \frac{5}{36} & \frac{2}{9} - \frac{\sqrt{15}}{15} & \frac{5}{36} - \frac{\sqrt{15}}{30} \\
\frac{1}{2} & \frac{5}{36} + \frac{\sqrt{15}}{24} & \frac{2}{9} & \frac{5}{36} - \frac{\sqrt{15}}{24} \\
\frac{1}{2} + \frac{\sqrt{15}}{10} & \frac{5}{36} + \frac{\sqrt{15}}{30} & \frac{2}{9} + \frac{\sqrt{15}}{15} & \frac{5}{36} \\
\hline
 & \frac{5}{18} & \frac{4}{9} & \frac{5}{18}
\end{array}
$$

RKI(8,4)

A.99

$$
\begin{array}{c|cccc}
\frac{1}{2} - \omega_2 & \omega_1 & \omega_1' - \omega_3 + \omega_4' & \omega_1' - \omega_3 - \omega_4' & \omega_1 - \omega_5 \\
\frac{1}{2} - \omega_2' & \omega_1 - \omega_3' + \omega_4 & \omega_1' & \omega_1' - \omega_5' & \omega_1 - \omega_3' - \omega_4 \\
\frac{1}{2} + \omega_2' & \omega_1 + \omega_3' + \omega_4 & \omega_1' + \omega_5' & \omega_1' & \omega_1 + \omega_3' - \omega_4 \\
\frac{1}{2} + \omega_2 & \omega_1 + \omega_5 & \omega_1' + \omega_3 + \omega_4' & \omega_1' + \omega_3 - \omega_4' & \omega_1 \\
\hline
 & 2\omega_1 & 2\omega_1' & 2\omega_1' & 2\omega_1
\end{array}
$$

$$
\Bigg[\omega_1 = \frac{1}{8} - \frac{\sqrt{30}}{144}, \quad \omega_1' = \frac{1}{8} + \frac{\sqrt{30}}{144}, \quad \omega_2 = \frac{1}{2}\sqrt{\frac{15 + 2\sqrt{30}}{35}},
$$

$$
\omega_2' = \frac{1}{2}\sqrt{\frac{15 - 2\sqrt{30}}{35}}, \quad \omega_3 = \omega_2\left(\frac{1}{6} + \frac{\sqrt{30}}{24}\right), \quad \omega_3' = \omega_2'\left(\frac{1}{6} - \frac{\sqrt{30}}{24}\right),
$$

$$
\omega_4 = \omega_2\left(\frac{1}{21} + \frac{5\sqrt{30}}{168}\right), \quad \omega_4' = \omega_2'\left(\frac{1}{21} - \frac{5\sqrt{30}}{168}\right), \quad \omega_5 = \omega_2 - 2\omega_3,
$$

$$
\omega_5' = \omega_2' - 2\omega_3'\Bigg]
$$

RKI(10,5)

A.100

$$
\begin{array}{c|ccccc}
\frac{1}{2} - \omega_2 & \omega_1 & \omega_1' - \omega_3 + \omega_4' & \frac{32}{225} - \omega_5 & \omega_1' - \omega_3 - \omega_4' & \omega_1 - \omega_6 \\
\frac{1}{2} - \omega_2' & \omega_1 - \omega_3' + \omega_4 & \omega_1' & \frac{32}{225} - \omega_5' & \omega_1' - \omega_6' & \omega_1 - \omega_3' - \omega_4 \\
\frac{1}{2} & \omega_1 + \omega_7 & \omega_1' + \omega_7' & \frac{32}{225} & \omega_1' - \omega_7' & \omega_1 - \omega_7 \\
\frac{1}{2} + \omega_2' & \omega_1 + \omega_3' + \omega_4 & \omega_1' + \omega_6' & \frac{32}{225} + \omega_5' & \omega_1' & \omega_1 + \omega_3' - \omega_4 \\
\frac{1}{2} + \omega_2 & \omega_1 + \omega_6 & \omega_1' + \omega_3 + \omega_4' & \frac{32}{225} + \omega_5 & \omega_1' + \omega_3 - \omega_4' & \omega_1 \\
\hline
 & 2\omega_1 & 2\omega_1' & \frac{64}{225} & 2\omega_1' & 2\omega_1
\end{array}
$$

$$
\Bigg[\omega_1 = \frac{322 - 13\sqrt{70}}{3600}, \quad \omega_1' = \frac{322 + 13\sqrt{70}}{3600}, \quad \omega_2 = \frac{1}{2}\sqrt{\frac{35 + 2\sqrt{70}}{63}},
$$

$$
\omega_2' = \frac{1}{2}\sqrt{\frac{35 - 2\sqrt{70}}{63}}, \quad \omega_3 = \omega_2\left(\frac{452 + 59\sqrt{70}}{3240}\right), \quad \omega_3' = \omega_2'\left(\frac{452 - 59\sqrt{70}}{3240}\right),
$$

$$
\omega_4 = \omega_2\left(\frac{64 + 11\sqrt{70}}{1080}\right), \quad \omega_4' = \omega_2'\left(\frac{64 - 11\sqrt{70}}{1080}\right), \quad \omega_5 = 8\omega_2\left(\frac{23 - \sqrt{70}}{405}\right),
$$

$$
\omega_5' = 8\omega_2'\left(\frac{23 + \sqrt{70}}{405}\right), \quad \omega_6 = \omega_2 - 2\omega_3 - \omega_5, \quad \omega_6' = \omega_2' - 2\omega_3' - \omega_5',
$$

$$
\omega_7 = \omega_2\left(\frac{308 - 23\sqrt{70}}{960}\right), \quad \omega_7' = \omega_2'\left(\frac{308 + 23\sqrt{70}}{960}\right)\Bigg].
$$

A.3-2 RKI(p,V)'s of the Radau Quadrative forms:

These types of RKI's are listed in Butcher [55], from p=1 to p=5 (p = 2V - 1). Butcher [55] also describes an ingenious error analysis applicable to the Radau and Lobatto (see section A.3-3) forms of RKI's.

$$\begin{array}{c|c} 0 & 0 \\ \hline & 1 \end{array}$$

RKI(1,1)

A.101

$$\begin{array}{c|cc} 0 & 0 & 0 \\ \frac{2}{3} & \frac{1}{3} & \frac{1}{3} \\ \hline & \frac{1}{4} & \frac{3}{4} \end{array}$$

RKI(3,2)

A.102

$$\begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ \frac{6-\sqrt{6}}{10} & \frac{9+\sqrt{6}}{75} & \frac{24+\sqrt{6}}{120} & \frac{168-73\sqrt{6}}{600} \\ \frac{6+\sqrt{6}}{10} & \frac{9-\sqrt{6}}{75} & \frac{168+73\sqrt{6}}{600} & \frac{24-\sqrt{6}}{120} \\ \hline & \frac{1}{9} & \frac{16+\sqrt{6}}{36} & \frac{16-\sqrt{6}}{36} \end{array}$$

RKI(5,3)

A.103

$$\begin{array}{c|cc} \frac{1}{3} & \frac{1}{3} & 0 \\ 1 & 1 & 0 \\ \hline & \frac{3}{4} & \frac{1}{4} \end{array}$$

RKI(3,2)

A.104

$$\begin{array}{c|ccc} \frac{4-\sqrt{6}}{10} & \frac{24-\sqrt{6}}{120} & \frac{24-11\sqrt{6}}{120} & 0 \\ \frac{4+\sqrt{6}}{10} & \frac{24+11\sqrt{6}}{120} & \frac{24+\sqrt{6}}{120} & 0 \\ 1 & \frac{6-\sqrt{6}}{12} & \frac{6+\sqrt{6}}{12} & 0 \\ \hline & \frac{16-\sqrt{6}}{36} & \frac{16+\sqrt{6}}{36} & \frac{1}{9} \end{array}$$

RKI(5,3)

A.105

A.3-3 RKI(p,V)'s of the Lobatto Quadrative forms:

Butcher [55] lists these types of RKI's from p=2 to p=12 (p = 2V - 2). An ingenious error analysis applicable to these forms is also described.

$$\begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 1 & 0 \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

RKI(2,2)

A.106

$$\begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{4} & 0 \\ 1 & 0 & 1 & 0 \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}$$

RKI(4,3)

A.107

$$\begin{array}{c|cccc} 0 & 0 & 0 & 0 & 0 \\ \frac{5-\sqrt{5}}{10} & \frac{5+\sqrt{5}}{60} & \frac{1}{6} & \frac{15-7\sqrt{5}}{60} & 0 \\ \frac{5+\sqrt{5}}{10} & \frac{5-\sqrt{5}}{60} & \frac{15+7\sqrt{5}}{60} & \frac{1}{6} & 0 \\ 1 & \frac{1}{6} & \frac{5-\sqrt{5}}{12} & \frac{5+\sqrt{5}}{12} & 0 \\ \hline & \frac{1}{12} & \frac{5}{12} & \frac{5}{12} & \frac{1}{12} \end{array}$$

RKI(6,4)

A.108

$$\begin{array}{c|ccccc} 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{7-\sqrt{21}}{14} & \frac{1}{14} & \frac{1}{9} & \frac{13-3\sqrt{21}}{63} & \frac{14-3\sqrt{21}}{126} & 0 \\ \frac{1}{2} & \frac{1}{32} & \frac{91+21\sqrt{21}}{576} & \frac{11}{72} & \frac{91-21\sqrt{21}}{576} & 0 \\ \frac{7+\sqrt{21}}{14} & \frac{1}{14} & \frac{14+3\sqrt{21}}{126} & \frac{13+3\sqrt{21}}{63} & \frac{1}{9} & 0 \\ 1 & 0 & \frac{7}{18} & \frac{2}{9} & \frac{7}{18} & 0 \\ \hline & \frac{1}{20} & \frac{49}{180} & \frac{16}{45} & \frac{49}{180} & \frac{1}{20} \end{array}$$

RKI(8,5)

A.109

RKI(10,6)

A.110

$$
\begin{array}{c|cccccc}
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{1-\omega_2}{2} & \frac{1}{30}-\omega_8 & \omega_1-\frac{1}{60} & \omega_1'+\omega_3'+\omega_4-\omega_6 & \omega_1'-\omega_3'-\omega_4-\omega_6 & \omega_1+\frac{1}{60}-\omega_7 & 0 \\
\frac{1-\omega_2'}{2} & \frac{1}{30}-\omega_8' & \omega_1+\omega_3+\omega_4'-\omega_6' & \omega_1'-\frac{1}{60} & \omega_1'+\frac{1}{60}-\omega_7' & \omega_1-\omega_3-\omega_4'-\omega_6' & 0 \\
\frac{1+\omega_2'}{2} & \frac{1}{30}+\omega_8' & \omega_1+\omega_3-\omega_4'+\omega_6' & \omega_1'+\frac{1}{60}+\omega_7' & \omega_1'-\frac{1}{60} & \omega_1-\omega_3+\omega_4'+\omega_6' & 0 \\
\frac{1+\omega_2}{2} & \frac{1}{30}+\omega_8 & \omega_1+\frac{1}{60}+\omega_7 & \omega_1'+\omega_3'-\omega_4+\omega_6 & \omega_1'-\omega_3'+\omega_4+\omega_6 & \omega_1-\frac{1}{60} & 0 \\
1 & \frac{1}{15} & 2\omega_1+\omega_5 & 2\omega_1'+\omega_5' & 2\omega_1'-\omega_5' & 2\omega_1-\omega_5 & 0 \\
\hline
 & \frac{1}{30} & 2\omega_1 & 2\omega_1' & 2\omega_1' & 2\omega_1 & \frac{1}{30}
\end{array}
$$

$$
\Big[\omega_1=\frac{14-\sqrt{7}}{120},\ \omega_1'=\frac{14+\sqrt{7}}{120},\ \omega_2=\sqrt{\frac{7+2\sqrt{7}}{21}},\ \omega_2'=\sqrt{\frac{7-2\sqrt{7}}{21}},
$$
$$
\omega_3=\frac{7-5\sqrt{7}}{60}\omega_2,\ \omega_3'=\frac{7+5\sqrt{7}}{60}\omega_2',\ \omega_4=\frac{2+5\sqrt{7}}{120}\omega_2,\ \omega_4'=\frac{2-5\sqrt{7}}{120}\omega_2',
$$
$$
\omega_5=\frac{14+\sqrt{7}}{3780}\sqrt{21},\ \omega_5'=\frac{14-\sqrt{7}}{3780}\sqrt{21},\ \omega_6=\frac{22+17\sqrt{7}}{360}\omega_2,\ \omega_6'=\frac{22-17\sqrt{7}}{360}\omega_2',
$$
$$
\omega_7=\frac{22-5\sqrt{7}}{60}\omega_5,\ \omega_7'=\frac{22+5\sqrt{7}}{60}\omega_5',\ \omega_8=\frac{1-\sqrt{7}}{90}\omega_2,\ \omega_8'=\frac{1+\sqrt{7}}{90}\omega_2'\Big].
$$

RKI(12,7)

A.111

$$
\begin{array}{c|ccccccc}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{1-\omega_2}{2} & \omega_4 & \omega_1-\frac{1}{84} & \omega_3-\omega_5+\omega_6' & \omega_7-\omega_8 & \omega_3-\omega_5-\omega_6' & \omega_1-\frac{1}{84}-\omega_9 & 0 \\
\frac{1-\omega_2'}{2} & \omega_4' & \omega_3'-\omega_5'+\omega_6 & \omega_1'-\frac{1}{84} & \omega_7'-\omega_8' & \omega_1'-\frac{1}{84}-\omega_9' & \omega_3'-\omega_5'-\omega_6 & 0 \\
\frac{1}{2} & \frac{1}{32} & \omega_{10}+\omega_{11} & \omega_{10}'+\omega_{11}' & \frac{11}{100} & \omega_{10}'-\omega_{11}' & \omega_{10}-\omega_{11} & 0 \\
\frac{1+\omega_2'}{2} & \omega_4' & \omega_3'+\omega_5'+\omega_6 & \omega_1'-\frac{1}{84}+\omega_9' & \omega_7'+\omega_8' & \omega_1'-\frac{1}{84} & \omega_3'+\omega_5'-\omega_6 & 0 \\
\frac{1+\omega_2}{2} & \omega_4 & \omega_1-\frac{1}{84}+\omega_9 & \omega_3+\omega_5+\omega_6' & \omega_7+\omega_8 & \omega_3+\omega_5-\omega_6' & \omega_1-\frac{1}{84} & 0 \\
1 & 0 & \omega_{12} & \omega_{12}' & \frac{8}{25} & \omega_{12}' & \omega_{12} & 0 \\
\hline
 & \frac{1}{42} & 2\omega_1 & 2\omega_1' & \frac{128}{525} & 2\omega_1' & 2\omega_1 & \frac{1}{42}
\end{array}
$$

$$
\Big[\omega_1=\frac{124-7\sqrt{15}}{1400},\ \omega_1'=\frac{124+7\sqrt{15}}{1400},\ \omega_2=\sqrt{\frac{15+2\sqrt{15}}{33}},\ \omega_2'=\sqrt{\frac{15-2\sqrt{15}}{33}},
$$
$$
\omega_3=\frac{413}{363}\omega_1',\ \omega_3'=\frac{413}{363}\omega_1,\ \omega_4=\frac{54+5\sqrt{15}}{2178},\ \omega_4'=\frac{54-5\sqrt{15}}{2178},\ \omega_5=\frac{98+27\sqrt{15}}{1320}\omega_2,
$$
$$
\omega_5'=\frac{98-27\sqrt{15}}{1320}\omega_2',\ \omega_6=\frac{658+199\sqrt{15}}{14520}\omega_2,\ \omega_6'=\frac{658-199\sqrt{15}}{14520}\omega_2',\ \omega_7=\frac{3276-100\sqrt{15}}{27225},
$$
$$
\omega_7'=\frac{3276+100\sqrt{15}}{27225},\ \omega_8=\frac{36-4\sqrt{15}}{165}\omega_2,\ \omega_8'=\frac{36+4\sqrt{15}}{165}\omega_2',\ \omega_9=\frac{8-\sqrt{15}}{60}\omega_2,
$$
$$
\omega_9'=\frac{8+\sqrt{15}}{60}\omega_2',\ \omega_{10}=\frac{801-73\sqrt{15}}{9600},\ \omega_{10}'=\frac{801+73\sqrt{15}}{9600},\ \omega_{11}=\frac{111-17\sqrt{15}}{640}\omega_2,
$$
$$
\omega_{11}'=\frac{111+17\sqrt{15}}{640}\omega_2',\ \omega_{12}=\frac{51+2\sqrt{15}}{300},\ \omega_{12}'=\frac{51-2\sqrt{15}}{300}\Big].
$$

A.4 Multistep Runge-Kutta Methods - RKMS(p,V):

One way of reducing the stages V of a p order Runge-Kutta is to use an implicit Runge-Kutta--the price paid would be the iterations required. Another way of reducing V for a given p order Runge-Kutta is to utilize the solutions of one or more previously evaluated steps. This result is a Multistep Runge-Kutta. The price paid here is that the inherent self-starting, easily institutable step size change, accuracy, and stability advantages of the single step Runge-Kutta's are lost. Another disadvantage (trivial) of a multistep method, compared to a single step method, is that extra storage is required to save the solutions of previously evaluated steps. The typical complex error analysis of single step Runge-Kutta's occurs in RKMS's too.

On the whole standard multistep predictor-corrector methods perform better than RKMS's; so there does not seem much point in using a RKMS instead of a RKE, if a multistep method is desired, a standard predictor-corrector is recommended.

Based on the usual formulation of a Runge-Kutta method, applicable to a first order ODE, a RKMS(p,V) is defined by:

$$Y_{n+1} = Y_n + \sum_{s=1}^{S_n} \sum_{i=1}^{V} w_i^{n-s+1} \, k_i^{n-s+1}$$

$$k_i^{n-s+1} = hj(T_{n-s+1} + c_i^{n-s+1} h, \; Y_{n-s+1} + \sum_{j=1}^{M_i} a_{ij}^{n-s+1} \, k_{ij}^{n-s+1}) \qquad \text{A.112}$$

$$S_n \leq n+1$$

$$M_i \leq V$$

$$n = 1,2,3,\cdots$$

The superscripts in eq. A.112 are used to identify w_i, k_i, c_i and a_{ij} at a particular time step. For example, S_n=1 makes eq. A.112 a single step Runge-Kutta; S_n=2 makes eq. A.112 a two step Runge-Kutta, and the k_i's of the (n-1)th time step are used together with the k_i's of the nth time step, to arrive at the solution Y_{n+1}. If S_n=n+1; then the wolutions of all previously evaluated steps, including the initial conditions; are used to calculate Y_{n+1}. Henceforth RKMS(p,V,S_n) will be used to designate these methods. The value of S_n identifies the number of steps used in the RKMS. Bryne and Lambert 59 show that there is not much point in going beyond a two step (S_n=2) Runge-Kutta, because both accuracy and stability of these RKMS's deteriorate as compared to single step Runge-Kutta's.

The rest of this section will discuss S_n=2, or two step RKMS's, eq. A.112 is reformulated in a more convenient form for S_n=2.

$$Y_{n+1} = Y_n + \sum_{i=1}^{V} u_i g_i + \sum_{i=1}^{V} w_i k_i$$

$$g_i = hf(T_{n-1} + b_i h,\ Y_{n-1} + \sum_{j=1}^{M_i} d_{ij} g_i) \qquad \text{A.113}$$

$$k_i = hf(T_n + c_i j,\ Y_n + \sum_{j=1}^{M_i} a_{ij} k_j)$$

$$M_i \leq V$$

Usually M_i = i-1 is used to give an explicit formulation.

Rosen [62] gives the deriving equations for RKMS(4,3,2) and RKMS (5,5,2) using the convenient quadrative approach (convenient as compared to the Taylor Series approach). Butcher [56,57] also gives the deriving equation, but the methods of these papers are really hybrid methods, not RKMS's, and so come under section A.5.

Bryne and Lambert [59] present the deriving equations for two RKM's: a RKMS(3,2,2) and a RKMS(4,3,2). The general solution, error analysis and proof of convergence is also presented. Bryne [58] gives the parameters which make the above two methods optimum (minimum TEB based on the Ralston [8] criteria, see sections A.1-3 and A.1-4). Lapidus and Seinfeld [6] also list these two methods.

RKMS(3,2,2):

<u>n-1</u>	<u>n</u>	
$b_1=0$	$c_1=0$	
$b_2=c_2$	$c_2=c_2$	
$d_{21}=c_2$	$a_1=c_2$	A.114
$u_1=1-w_1$	$w_1=\frac{(18c_2-5)}{12c_2}$	
$u_2=-w_2$	$w_2=\frac{5}{12c_2}$	

c_2 free, $c_2=\frac{4}{5}$ gives an optimum method.

RKMS(4,3,2):

<u>n-1</u>	<u>n</u>	
$b_1=0$	$c_1=0$	
$b_2=c_2$	$c_2=c_2$	
$b_3=c_3$	$c_3=c_3$	
$d_{21}=c_2$	$a_{21}=c_2$	
$d_{31}=a_{31}$	$a_{31}=c_3-a_{32}$	
$d_{32}=a_{32}$	$a_{32}=\frac{[2c_3(c_3-c_2)]}{[c_2(4-5c_2)]}$	A.115
$u_1=1-w_1$	$w_1=\frac{[4+18c_2c_3-5(c_2+c_3)]}{12c_2c_3}$	
$u_2=-w_2$	$w_2=\frac{[4-5c_3]}{[12c_2(c_2-c_3)]}$	
$u_3=-w_1$	$w_3=\frac{[5c_2-4]}{[12c_3(c_2-c_3)]}$	

c_2 and c_3 are free, $c_2=.541$ and $c_3=.722779927$ give an optimum method.

Lapidus and Seinfeld 6 also list the above two methods.

These two methods, eq. A.114 and A.115, are not as accurate as comparable order RKE's (though they are faster).

Chai 60 develops an optimum RKMS(4,3,2) using an ingenious derivation. Chai notes that if an RKE(4,4) has $w_2=0$ then the k_2 can be reformulated in a multistep form without changing the c_i's, a_{ij}'s, and w_i's of the RKE(4,4). The method is hence defined by the usual RKE(4,4) formulation but with k_2 given differently.

$$k_2 = hf(T_n,Y_n) + h^2c_2f'(T_n,Y_n) + \frac{h^3}{2!}\, e_1\, j'''(T_n,Y_n) + \frac{h^4}{3!}\, e_2f''''(T_n,Y_n) + \cdots \qquad \text{A.116}$$

where

$c_2 \neq 0$

e_1 and e_2 are constants.

By using a finite difference approximation for $f'(T_n, Y_n)$ in eq. A.116 Chai arrives at:

$$k_2 = q_1k_1 + q_2\nabla Y_{n-1} + q_3g_1 \qquad \text{A.117}$$

where:

$$\nabla Y_{n-1} = Y_n - Y_{n-1}$$

$$\therefore\ \nabla Y_{n-1} = \sum_{i=1}^{V} w_ig_i \qquad \text{A.118}$$

the g_i's being the k_i's of the previous step.

Thus V=3 is achieved because no function evaluation is necessary for k_2.

By comparing a Taylor series expansion of eq. A.117, up to second order terms, with the first two terms of eq. A.116, gives a set of conditions for the q's.

$$q_1 + q_2 + q_3 = 1$$
$$\frac{q_2}{2} + q_3 = -c_2 \qquad \text{A.119}$$

Thus one q is arbitrary.

An optimum RKMS(4,3,2) is hence formulated by Chai (minimum TEB) this uses the same c_i's, a_{ij}'s and w_i's as the RKE(4,4) from which it is derived, hence no separate starting method is necessary.

RKMS(4,3,2):

$c_1 = 0$

$c_2 = c_2 \quad q_1 = 1 + \frac{26}{5} c_2 \quad q_2 = -\frac{42}{5} c_2 \quad q_3 = \frac{16}{5} c_2$

$c_3 = 1/2$

$c_4 = 1$

$a_{21} = c_2$ A.120

$a_{31} = \frac{1}{2} - \frac{1}{8c_2} \qquad a_{32} = \frac{1}{8c_2}$

$a_{41} = \frac{1}{2c_2} - 1 \qquad a_{42} = -\frac{1}{2c_2} \qquad a_{43} = 2$

$w_1 = 1/6 \qquad w_2 = 0 \qquad w_3 = \frac{2}{3} \qquad w_4 = 1/6$

$c_2 = 1/2$ or $1/4$ is recommended.

Chai's tests show that this RKMS is superior on accuracy and time to the RKE with which it shares its coefficients. An error expression for corre-ting the solution, to yield a fifth order accuracy is also given. Due to its simplicity of application and other advantages, this method is highly recommended.

Gruttke [61] gives the deriving equations for a RKMS(5,4,2). These equations are solved to yield various methods which are extensively tested. An optimum based on the Ralston [8] criteria was determined.

RKMS(5,4,2):

n-1	n
$b_1 = 0$	$c_1 = 0$
$b_2 = c_2$	$c_2 = c_2$
$b_3 = c_3$	$c_3 = c_3$
$b_4 = c_4$	$c_4 = \frac{62}{85}$
$d_{21} = c_2$	$a_{21} = c_2$
$d_{31} = a_{31}$	$a_{31} = c_3 - a_{32}$
$d_{32} = a_{32}$	$a_{32} = \frac{31 - 40c_4}{240\ w_3c_2(c_3 - c_4)}$ A.121
$d_{41} = a_{41}$	$a_{41} = c_4 - a_{42} - a_{43}$
$d_{42} = a_{42}$	$a_{42} = \frac{1}{[120\ w_4c_2(c_4 - c_3)]} \cdot [\frac{(c_3 - c_4)(31 - 60c_3)}{3(c_3 - c_2)} + \frac{(31 - 40c_4)}{2}]$
$d_{43} = a_{43}$	$a_{43} = \frac{31 - 60c_2}{360\ w_4c_3(c_3 - c_2)}$
$u_1 = 1 - w_1$	$w_1 = \frac{3}{2} - (w_2 + w_3 + w_4)$
$u_2 = -w_2$	$w_2 = \frac{31 + 50c_3c_4 - 40(c_3 + c_4)}{120c_2(c_2 - c_3)(c_2 - c_4)}$
$u_3 = -w_3$	$w_3 + \frac{31 + 50c_2c_4 - 40(c_2 + c_4)}{120c_3(c_3 - c_2)(c_3 - c_4)}$
$u_4 = w_4$	$w_4 = \frac{31 + 50c_2c_3 - 40(c_2 + c_3)}{120c_4(c_4 - c_2)(c_4 - c_3)}$

c_2 and c_3 free, $c_2 = .25$ and $c_3 = .869$ give an optimum method.

Again, eq. A.117 would be faster than comparable fifth order RKE's but not quite as accurate.

There would be very little advantage in deriving an implicit or semi-implicit RKMS form ($M_i = V$ or $M_i = i$ in eqs. A.112 or A.113), because of the additional complexities involved.

A.5 Special Types or Hybrid Runge-Kutta Methods

Two most widely used methods of numerically solving ODE's are the Runge-Kutta methods and the predictor-corrector methods.

For complex systems of ODE's numerical solutions, using the usual Runge-Kutta's discussed so far, have one big disadvantage--function evaluations take up the major portion of the computational time involved. Multi-step predictor-corrector methods require considerably less function evaluations per step. So obviously reducing the number of function evaluations (V) of a Runge-Kutta, while retaining some or all the good points of a Runge-Kutta, would yield good methods. Some compromise between Runge-Kutta's and predictor-correctors is indicated.

Single step Runge-Kutta's have certain definite advantages and disadvantages, and so do the predictor-corrector multi-step method.

Characteristics of Comparable Order Methods

Single Step Runge-Kutta	Multi-step predictor-correctors
Advantages:	Disadvantages:
1) Self-starting	1) Not self-starting
2) Comparatively good stability characteristics because single step (so large h possible)	2) Comparatively bad stability characteristics because multi-step (h comparatively small)
3) Step size and order easily changeable at each step.	3) Step size and order change not easily instituted at each step.

Disadvantages:

1) May require greater computational time as comparatively large number of function evaluations required

2) No easily computable and accurate error estimate available

3) High order methods involve extremely complicated deriving equations.

Advantages:

1) Computationally fast as comparatively small number of function evaluations required (provided not too many iterations done).

2) Fairly easily computable and accurate error estimate usually available.

3) High order methods easily deroved/

Hybrid Runge-Kutta's have been developed to combine the advantages of the Runge-Kutta and predictor-corrector methods, while trying to avoid their disadvantages.

Rosser 63 developed a class of ingenious hybrid Runge-Kutta's with great potential. These methods favorably combine the advantageous features of Runge-Kutta's and predictor-correctors. They may be described as implicit, multi-step, predictor-corrector Runge-Kutta's.

These methods are "Block" methods and proceed by blocks of N steps. Each block is completely independent of other blocks. The solutions from previously evaluated steps are only required from, and used within, a particular block. A large degree of flexibility is built into these block methods giving them three concrete advantages:

1) Step size is constant within a block but can easily be changed from block to block

2) Order of accuracy can be varied from block to block by changing

N, the number of steps constituting a particular block; and by changing the number of iterations used in each block.

3) Accurate estimates of errors incurred within a block are available and so give good pointers for the best values of h and N for the next block.

These "Block" methods result in a considerable savings of function evaluations and are hence highly recommended. The Rosser "Block" Runge-Kutta's can be defined using the following formulation:

For a block of N steps:

$x_1 - x_0 = x_2 - x_1 = \cdot\cdot\cdot\cdot\cdot\cdot = x_n - x_{n-1} = h > 0$

The subscript 0 denotes the starting or initial step, hence x_0, y_0 are known (from initial values or the results of the previous block).

$$\left.\begin{array}{l} y_0 = y_0 \\ y_0{}' = j(x_0,y_0) \end{array}\right] \text{ Starting values for a block} \qquad \text{A.122}$$

$$y_m^r = y_0 + mhf(x_0,y_0) \qquad \text{A.123}$$

$$y_m'^{r} = f(x_m, y_m^r) \qquad \text{A.124}$$

$$r \geq 0 \quad m = 1,2,3,\cdots N$$

The superscript r is the iterate count.

$$y_1^{r+1} = y_0 + h[w_1y_0' + a_{11}y_1'^r + \cdots + a_{1m}y_m'^r + \cdots + a_{1n}y_n'^r]$$

$$\vdots$$

$$y_m^{r+1} = y_0 + h[w_my_0' + a_{m1}y_1'^r + \cdots + a_{mm}y_m'^r + \cdots + a_{mn}y_n'^r] \qquad \text{A.125}$$

$$\vdots$$

$$y_n^{r+1} = y_0 + h[w_ny_0' + a_{n1}y_1'^r + \cdots + a_{nm}y_m'^r + \cdots + a_{nn}y_n'^r]$$

$$r \geq 0 \quad m = 1,2,3,\cdots N$$

$w_1, w_2 \cdots w_n$ and $a_{11}, a_{1m} \cdots a_{nn}$ are constants defining a particular block method.

The working of the block method is easily understood. Consider the r=0 case, i.e., the first iteration: y_1^1 is calculated from eq. A.125; this requires N function evaluations--$y_1^{'o}$, $y_2^{'o}$, $\cdots y_n^{'o}$ which are calculated from eqs. A.122 and A.124 (y_0' is assumed known from a previous block and remains the same for any iteration; though y_0' must be calculated for the first block from eq. A.122 and hence the first block needs N+1 function evaluations). Similarly y_2^1, $y_3^1 \cdots y_n^1$ are calculated from eq. A.125, these use the same $y_1^{'o}$, $y_2^{'o}, \cdots y_n^{'o}$ used by y_1^1. Thus a complete sweep for a block (one complete iteration of eq. A.125) requires N function evaluations. For r=1, i.e., the second iteration: y_1^2 is calculated from eq. A.125 in the same way as for the r=0 case: the values y_1^1, y_2^1, $\cdots y_n^1$ are substituted in eq. A.124 to yield $y_1^{'1}$, $y_2^{'1}$, $\cdots y_n^{'1}$, which are then used in eq. A.125. y_2^2, y_3^2, $\cdots y_n^2$ are similarly calculated.

Up to a point increasing r, the number of iterations, increases the order of accuracy of the solutions $y_1, y_2 \cdots y_n$. For example, for a N=4 method (Milne method) with $0 \leq r \leq 4$ each iteration gives y_m^r to an order of accuracy of 4+2. For r > 4 no improvement in order of accuracy is gotten. If r=3 is used a fifth order method results and uses a total of 16 (4N) function evaluation for a block of four steps (N=4). Thus a fifth order Runge-Kutta is gotten which uses four function evaluations per step. If r=4 is used a sixth order Runge-Kutta results which uses a total of twenty (5N) function evaluations for a block of four steps (N+4). Thus a sixth order Runge-Kutta is gotten which uses five function evaluations per step.

The above discussion illustrates the flexibility of these "Block" Runge-Kutta's--high order of accuracy for the number of function evaluations, and variable order of accuracy possible.

Rosser 63 has derived several such methods where the number of function evaluations are further reduced by curtailing some of the early steps, and accelerating the convergence of the later steps. In these methods order of accuracy is changeable, and is changed by varying N or r. An error estimate is also given, and a procedure for changing N and h from block to block outlined. In general if $N = 2r$ or $2r + 1$ a Runge-Kutta of order $2r + 1$ results.

The Block approach improves conventional Runge-Kutta's and makes them comparable to predictor-correctors on speed. But the "Block" approach is applicable to predictor-correctors also and will improve them too.

Rosser has derived methods up to $N \leq 8$, i.e., up to a tenth order Runge-Kutta. In the following formulae $y^{(s)}$ is used in the error estimates and is $\frac{d^s y}{dx^s}$ evaluated in $[x_0, x_N]$.

One-point formulas:

$$y_1 - y_0 = hy_0' + \frac{h^2 y^{(2)}}{2}. \qquad \text{A.126}$$

Two-point formulas:

$$y_1 - y_0 = \frac{h}{2}(y_0' + y_1') - \frac{h^3 y^{(3)}}{12},$$

$$y_2 - y_0 = 2hy_1' + \frac{h^3 y^{(3)}}{3}. \qquad \text{A.127}$$

Three-point formulas:

$$y_1 - y_0 = \frac{h}{12}(5y_0' + 8y_1' - y_2') + \frac{h^4 y^{(4)}}{24}, \qquad \text{A.128}$$

$$y_2 - y_0 = \frac{h}{3}(y_0' + 4y_1' + y_2') - \frac{h^5 y^{(5)}}{90},$$

$$y_3 - y_0 = \frac{3h}{4}(y_0' + 3y_2') + \frac{3h^4 y^{(4)}}{8}.$$

Four-point formulas:

$$y_1 - y_0 = \frac{h}{24}(9y_0' + 19y_1' - 5y_2' + y_3') - \frac{19h^5 y^{(5)}}{720}, \qquad \text{A.129}$$

$$y_2 - y_0 = \frac{h}{3}(y_0' + 4y_1' + y_2') - \frac{h^5 y^{(5)}}{90},$$

$$y_3 - y_0 = \frac{3h}{8}(y_0' + 3y_1' + 3y_2' + y_3') - \frac{3h^5 y^{(5)}}{80},$$

$$y_4 - y_0 = \frac{4h}{3}(2y_1' - y_2' + 2y_3') + \frac{14h^5 y^{(5)}}{45},$$

$$y_6 - y_3 = \frac{h}{32}(10y_0' - 81y_2' + 136y_3' + 31y_6') - \frac{351h^5 y^{(5)}}{160}.$$

Five-point formulas:

$$y_1 - y_0 = \frac{h}{720}(251y_0' + 646y_1' - 264y_2' + 106y_3' - 19y_4') + \frac{3h^6 y^{(6)}}{160}, \qquad \text{A.130}$$

$$y_2 - y_0 = \frac{h}{90}(29y_0' + 124y_1' + 24y_2' + 4y_3' - y_4') + \frac{h^6 y^{(6)}}{90},$$

$$y_3 - y_0 = \frac{3h}{80}(9y_0' + 34y_1' + 24y_2' + 14y_3' - y_4') + \frac{3h^6 y^{(6)}}{160},$$

$$y_4 - y_0 = \frac{2h}{45}(7y_0' + 32y_1' + 12y_2' + 32y_3' + 7y_4') - \frac{8h^7 y^{(7)}}{945},$$

$$y_5 - y_0 = \frac{5h}{144}(19y_0' - 10y_1' + 120y_2' - 70y_3' + 85y_4') + \frac{95h^6 y^{(6)}}{288},$$

$$y_2 - y_0 = \frac{h}{540}(329y_0' + 1539y_2' - 1216y_3' + 459y_4' - 31y_6') + \frac{16h^6 y^{(6)}}{45},$$

$$y_6 - y_4 = \frac{h}{540}\left(-31y_0' + 459y_2' - 1216y_3' + 1539y_4' + 329y_6'\right) - \frac{16h^6y^{(6)}}{45},$$

$$y_6 - y_0 = \frac{h}{20}\left(11y_0' + 81y_2' - 64y_3' + 81y_4' + 11y_6'\right) - \frac{9h^7y^{(7)}}{35}.$$

Six-point formulas:

$$y_1 - y_0 = \frac{h}{1440}\left(475y_0' + 1427y_1' - 798y_2' + 482y_3' - 173y_4' + 27y_5'\right) - \frac{863h^7y^{(7)}}{60480},$$

$$y_2 - y_0 = \frac{h}{90}\left(28y_0' + 129y_1' + 14y_2' + 14y_3' - 6y_4' + y_5'\right) - \frac{37h^7y^{(7)}}{3780},$$ A.131

$$y_3 - y_0 = \frac{3h}{160}\left(17y_0' + 73y_1' + 38y_2' + 38y_3' - 7y_4' + y_5'\right) - \frac{29h^7y^{(7)}}{2240},$$

$$y_4 - y_0 = \frac{2h}{45}\left(7y_0' + 32y_1' + 12y_2' + 32y_3' + 7y_4'\right) - \frac{8h^7y^{(7)}}{945},$$

$$y_5 - y_0 = \frac{5h}{288}\left(19y_0' + 75y_1' + 50y_2' + 50y_3' + 75y_4' + 19y_5'\right) - \frac{275h^7y^{(7)}}{12096},$$

$$y_6 - y_0 = \frac{3h}{10}\left(11y_1' - 14y_2' + 26y_3' - 14y_4' + 11y_5'\right) + \frac{41h^7y^{(7)}}{140}.$$

Seven-point formulas:

$$y_1 - y_0 = \frac{h}{60480}\left(19087y_0' + 65112y_1' - 46461y_2' + 37504y_3' - 20211y_4' + 6312y_5' - 863y_6'\right) + \frac{275h^8y^{(8)}}{24192},$$

$$y_2 - y_0 = \frac{h}{3780}\left(1139y_0' + 5640y_1' + 33y_2' + 1328y_3' - 807y_4' + 264y_5' - 37y_6'\right) + \frac{8h^8y^{(8)}}{945},$$ A.132

$$y_3 - y_0 = \frac{h}{2240}(685y_0' + 3240y_1' + 1161y_2' + 2176y_3'$$
$$- 729y_4' + 216y_5' - 29y_6') + \frac{9h^8y^{(8)}}{896},$$

$$y_4 - y_0 = \frac{2h}{945}(143y_0' + 696y_1' + 192y_2' + 752y_3'$$
$$+ 87y_4' + 24y_5' - 4y_6') + \frac{8h^8y^{(8)}}{945},$$

$$y_5 - y_0 = \frac{5h}{12096}(743y_0' + 3480y_1' + 1275y_2' + 3200y_3'$$
$$+ 2325y_4' + 1128y_5' - 55y_6') + \frac{275h^8y^{(8)}}{24192},$$

$$y_6 - y_0 = \frac{h}{140}(41y_0' + 216y_1' + 27y_2' + 272y_3'$$
$$+ 27y_4' + 216y_5' + 41y_6') - \frac{9h^9y^{(9)}}{1400},$$

$$y_7 - y_0 = \frac{7h}{8640}(751y_0' - 840y_1' + 8547y_2' - 11648y_3'$$
$$+ 14637y_4' - 7224y_5' + 4417y_6') + \frac{5257h^8y^{(8)}}{17280},$$

$$y_8 - y_0 = \frac{4h}{945}(115y_0' + 1312y_2' - 2048y_3' + 3132y_4'$$
$$- 2048y_5' + 1312y_6' + 115y_8') - \frac{2944h^9y^{(9)}}{14175}.$$

Eight-point formulas:

$$y_1 - y_0 = \frac{h}{120960}(36799y_0' + 139849y_1' - 121797y_2' + 123133y_3' \qquad \text{A.133}$$
$$- 88547y_4' + 41499y_5' - 11351y_6' + 1375y_7') - \frac{33953h^9y^{(9)}}{3628800},$$

$$y_2 - y_0 = \frac{h}{3780}(1107y_0' + 5864y_1' - 639y_2' + 2448y_3'$$
$$- 1927y_4' + 936y_5' - 261y_6' + 32y_7') - \frac{119h^9y^{(9)}}{16200},$$

$$y_3 - y_0 = \frac{h}{4480}(1325y_0' + 6795y_1' + 1377y_2' + 5927y_3'$$
$$- 3033y_4' + 1377y_5' - 373y_6' + 45y_7') - \frac{369h^9y^{(9)}}{44800},$$

$$y_4 - y_0 = \frac{2h}{945}(139y_0' + 724y_1' + 108y_2' + 892y_3'$$
$$- 53y_4' + 108y_5' - 32y_6' + 4y_7') - \frac{107h^9y^{(9)}}{14175},$$

$$y_5 - y_0 = \frac{5h}{145152}(8341y_0' + 46030y_1' + 1510y_2' + 63670y_3'$$

$$- 800y_4' + 34186y_5' - 9830y_6' + 2290y_7' - 245y_8')$$

$$+ \frac{25h^{10}y^{(10)}}{3584},$$

$$y_6 - y_0 = \frac{h}{1400}(401y_0' + 2232y_1' + 18y_2' + 3224y_3'$$

$$- 360y_4' + 2664y_5' + 158y_6' + 72y_7' - 9y_8')$$

$$+ \frac{9h^{10}y^{(10)}}{1400},$$

$$y_7 - y_0 = \frac{7h}{518400}(21361y_0' + 116662y_1' + 6958y_2' + 155134y_3'$$

$$+ 7840y_4' + 105154y_5' + 74578y_6' + 31882y_7' - 1169y_8')$$

$$+ \frac{8183h^{10}y^{(10)}}{1036800},$$

$$y_8 - y_0 = \frac{4h}{14175}(989y_0' + 5888y_1' - 928y_2' + 10496y_3'$$

$$- 4540y_4' + 10496y_5' - 928y_6' + 5888y_7' + 989y_8')$$

$$- \frac{2368h^{11}y^{(11)}}{467775}.$$

$$y_5 - y_0 = \frac{5h}{24192}(1431y_0' + 7345y_1' + 1395y_2' + 8325y_3' + 2725y_4' + 3411y_5' - 495y_6' + 55y_7') - \frac{175h^9y^{(9)}}{20736},$$

$$y_6 - y_0 = \frac{h}{140}(41y_0' + 216y_1' + 27y_2' + 272y_3' + 27y_4' + 216y_5' + 41y_6') - \frac{9h^9y^{(9)}}{1400},$$

$$y_7 - y_0 = \frac{7h}{17280}(751y_0' + 3577y_1' + 1323y_2' + 2989y_3' + 2989y_4' + 1323y_5' + 3577y_6' + 751y_7') - \frac{8183h^9y^{(9)}}{518400},$$

$$y_8 - y_0 = \frac{8h}{945}(460y_1' - 954y_2' + 2196y_3' - 2459y_4' + 2196y_5' - 954y_6' + 460y_7') + \frac{3956h^9y^{(9)}}{14175}.$$

Nine-point formulas:

$$y_1 - y_0 = \frac{h}{3628800}(1070017y_0' + 4467094y_1' - 4604594y_2' + 5595358y_3' - 5033120y_4' + 3146338y_5' - 1291214y_6' + 312874y_7' - 33953y_8') + \frac{8183h^{10}y^{(10)}}{1036800},$$ A.134

$$y_2 - y_0 = \frac{h}{113400}(32377y_0' + 182584y_1' - 42494y_2' + 120088y_3' - 116120y_4' + 74728y_5' - 31154y_6' + 7624y_7' - 833y_8') + \frac{9h^{10}y^{(10)}}{1400},$$

$$y_3 - y_0 = \frac{h}{44800}(12881y_0' + 70902y_1' + 3438y_2' + 79934y_3' - 56160y_4' + 34434y_5' - 14062y_6' + 3402y_7' - 369y_8') + \frac{25h^{10}y^{(10)}}{3584},$$

$$y_4 - y_0 = \frac{h}{14175}(4063y_0' + 22576y_1' + 244y_2' + 32752y_3' - 9080y_4' + 9232y_5' - 3956y_6' + 976y_7' - 107y_8') + \frac{94h^{10}y^{(10)}}{14175},$$

Butcher [56,57] also creates hybrid methods which are a combination of RKMS's and predictor-corrector methods. Butcher [57] lists RKMS's with $S_n = 2,3$, and 4 (see section A.4) and of order $p = 2S_n + 2$, with $V = 4$. Methods up to $S_n = 15$ are proved to exist. On basis of various tests these methods are highly recommended (they compare favorably with standard predictor-correctors). In the opinion of the author, the Rosser "Block" methods are, on the whole, better than these.

Gear [64], Gragg and Stetter [65], and Dahlquist [66] present methods similar to Butcher's methods.

Another special class of Runge-Kuttas is given in Gates [26]. Gates formulates explicit Runge-Kuttas in which each stage (each k_i) is independent of the other stages. In the conventional Runge-Kuttas each stage is not independent of the other stages. Each independent stage of the Gates formulae corresponds to a standard quadrative formula, e.g., Gauss, Radau, Lobatto, etc. Making each stage independent of the others has a penalty--V increases for a given order p as compared to the usual Runge-Kuttas. For example, for $p = 4,5$, and 6 Gates forms have $V = 5,7$ and 11 respectively. The advantage of Gates' formulation is the flexibility gotten by having the coefficients of one stage independent of another; another advantage is that deriving equations for these methods are much simpler than the usual RKE's and so high order methods can be derived fairly easily.

Stoller and Morrison [67] and Day [68] discuss similar quadrative Runge-Kuttas.

From the computational time point of view these methods are obviously not recommended.

A.6 General Comments for Further Development of the Runge-Kutta Class of Methods:

The previous sections have tried to sample various types of Runge-Kuttas available in current literature.

The field of Runge-Kuttas is wide open and improvements are possible. Runge-Kuttas in general arrive at an accurate solution at the end of an integration step, by combining solutions (k_i's) at intermediate steps which are not, in themselves so accurate. So one way of improving Runge-Kuttas would be to choose the coefficients of each stage in such a way that the k_i's would be accurate estimates of f(T,Y).

This should be possible without significantly increasing V, unlike the Gates [26] approach (see section A.5). Then as each k_i would be an accurate estimate of f(T,Y) a hermitian curve fit would be possible giving a higher order of accuracy.

The most obvious improvement needed by Runge-Kuttas is to reduce their stages V per step, this would make them comparable to predictor-correctors on computational time. In all other aspects (except error estimation), i.e., stability and self-starting features Runge-Kuttas are superior to predictor-correctors. So the second area of improvement lies in reducing V. This is possible to some extent by using the "Back Step" and "Front Step" methods outlined in section A.1-2, or by using the Shanks' technique outlined in section A.1-5. Conventional RKE's always have $c_1=0$ and usually have a $c_i=1$, hence the k_i corresponding to the $c_i=1$ of the (T_n,Y_n) step could be used for the k_i of the (T_{n+1}, Y_{n+1}) step reducing V by one. This would cost something--accuracy, because the solution Y_n, available at the end of

(T_n, Y_n) gives a more accurate estimate for k_i than is possible by using the k_i of the previous step. Also if two or more c_i's have the same value, then only one k_i corresponding to the first of these c_is need be evaluated. For the rest of these k_i's no more function evaluations are necessary, they can use the same value as the first of these k_i's. Again V is reduced but a loss in accuracy would occur. For example, a RKE(4,4) which has $c_1=0$, $c_2=\frac{1}{2}$, $c_3=\frac{1}{2}$, $c_4=1$ would become a RKE(4,2) by using the two techniques outlined above--k_i at (T_{n+1}, Y_{n+1}) would be the k_4 of (T_n, Y_n), k_3 at (T_{n+1}, Y_{n+1}) would be k_2 of the same step. It is not possible to say, off hand, whether the "Back Step" or "Front Step" technique would be more accurate than this one of interchanging k_i's. Obviously the interchange techniques could also be used to further reduce the V of a "Back Step" or "Front Step" method, with (probably) a further loss of accuracy, or the Rosser [63] (see section A.5) "Block" technique could be used to further speed up these methods.

A third area of development (not speed improvement) lies in the RKSI class of methods. Here a calculation procedure for $\frac{f}{y}$ or a Jacobian is required. As this is available it might as well be used to improve the order of accuracy, though at the cost of extra computational effort.

$$Y_{n+1} = Y_n + \sum_{i=1}^{V} w_i k_i + \sum_{i=1}^{V} u_i g_i$$

$$k_i = hf(T_n + c_i h,\ Y_n + \sum_{j=1}^{i} a_{ij} k_j) \qquad \text{A.135}$$

$$g_i = h^2 fy(T_n + d_i h,\ Y_n + \sum_{j=1}^{L_i} b_{ij} g_j)$$

$$L_i \le V$$

The Rosser [63] (see section A.5) should be applicable to speed up any type of Runge-Kutta.

Many authors have created hybrid methods which combine Runge-Kutta's with predictor-correctors, to try and combine the advantages of both methods (see section A.5). Similarly hybrid methods combining Runge-Kuttas with extrapolation and other methods should be possible.

The Rosser [63] "Block" technique (see section A.5) could be adapted to further speed up these hybrid methods. These hybrid methods may grow to such forms that Runge or Kutta would hardly recognize their method.

There are other areas in which Runge-Kutta's can be developed and improved, it is hoped that present and future mathematicians will do so.

Bibliography

1. Andrus, J., "Integration of Control Equations & the Problem of Small Time Constants," NASA Tech. Note D-3907 (1967).

2. Anderson, W., Ball, R., and Voss, J., "A Numerical Method for Solving Control Differential Equations on Digital Computers," J. of the Assoc. for Comp. Mach., 17, 61 (1960).

3. Babuska, I., Prager, M., and Vitosek, E., Numerical Processes in Differential Equations, Wiley (Interscience), New York (1966).

4. Ball, W., et al., "Automat," Comm. ACM, 9, 626 (1966).

5. Bard, A., Ceschino, F., Kuntzmann, J., and Laurent, P., "Formules de Base la Methode de Runge-Kutta," Chiffres, 4, 31 (1961).

6. Baxter, D., "The Digital Simulation of Transfer Functions," Nat. Res. Council of Canada Mechanical Eng. Report MK-13 (1964).

7. Baxter, D., "The Step Response of Digital Simulations," Nat. Res. Council of Canada Mechanical Eng. Report MK-14 (1965).

8. Baxter, D., "Digital Simulation Using Approximate Methods," Nat. Res. Council of Canada Mechanical Eng. Report MK-15 (1965).

9. Bennett, A., Milne, W., and Bateman, H., Numerical Integration of Differential Equations, Dover, New York (1956).

10. Benyon, P., "A Review of Numerical Methods for Digital Simulation," Simulation, 11, 219 (1968).

11. Benyon, P., "Comments on Giese's Paper, State Variable Difference Methods for Digital Simulation," Simulation, 8, 270 (1967).

12. Bieberbach, L., Theorie der Differentialgleichungen, Dover, New York (1944).

13. Blum, E., "A Formal System for Differentiation," J. of the Assoc. for Comp. Mach., 13, 495 (1966).

14. Branin, F., "Computer Aided Design: Part 4, Analyzing Circuits by the Numbers," Electronics, 40, 88 (1967).

15. Butcher, J., "On the Convergence of Numerical Solutions to Ordinary Differential Equations," Math. of Comp., 20, 1 (1966).

16. Butcher, J., "On the Integration Processes of A. Hulta," J. of Austral. Math. Soc., 3, 203 (1963).

17. Carr, J., "Error Bounds for the Runge-Kutta Singlestep Integration Process," J. of the Assoc. for Comp. Mach., 5, 1 (1958).

18. Chase, P., "Stability Properties of Predictor-Corrector Methods for Ordinary Differential Equations," J. of the Assoc. for Comp. Mach., 9, 457 (1962).

19. Christiansen, J., "Handbook Series Numerical Integration, Numerical Solution of Ordinary Simultaneous Differential Equations of the First Order Using a Method for Automatic Step Changes," Num. Math., 14, 317 (1970).

20. Cooper, G., "Interpolation & Quadrature Methods for Ordinary Differential Equations," Math. of Comp., 22, 69 (1968).

21. Cooper, G., and Gal, E., "Single Step Methods for Linear Differential Equations," Num. Math., 10, 307 (1967).

22. Davison, E., "A High Order Crank-Nicholson Technique for Solving Differential Equations," Comp. J., 10, 195 (1967).

23. Day, J., "A One-Step Method for the Numerical Integration of the Differential Equation $y'' = f(x)y + g(x)$," Comp. J., 7, 314 (1964).

24. Day, J., "A One-Step Method for the Numerical Solution of Second Order Linear Ordinary Differential Equations," Math. of Comp. 18, 664 (1964).

25. Ehle, B., "High Order A-Stable Methods for the Numerical Solution of Systems of Differential Equations," BIT, 8, 276 (1968).

26. Eisenpress, H. and Bomberault, A., "Efficient Symbolic Differentiation Using PL/1-FORMAC," IBM New York Scientific Center Report 320-29561 (1968).

27. El-Sherif, H., "Implicit Implementation of the Weighted Backward Euler Formula," IBM J. Res. Develop., 6, 336 (1968).

28. Fehlberg, E., "Eine Methode zur Fehlerverkleinerung beim Runge-Kutta Verfahren," ZAMM, 38, 421 (1958).

29. Fehlberg, E., "Neu Genauere Runge-Kutta Formeln fur Differentialgleichungen n-ter Ordung," ZAMM, 40, 449 (1960).

30. Filippi, S., "Contribution to the Implicit Runge-Kutta Process," Elektronische Datenverarbeitung, 10, 113 (1968).

31. Filippi, S. and Glasmacher, W., "New Results in Applying the Runge-Kutta Processes with the Aid of Nonnumerical Programs," Elektronische Datenverarbeitung, 10, 16 (1968).

32. Forrington, C. V. D., "The Numerical Solution of Systems of Differential Equations Using a Different Step-Length for Each Equation," R.A.E. Tech. Note M.S., 69 (1960).

33. Fowler, M., "A New Numerical Method for Simulation," Simulation, 4, No. 5, 324 (1965).

34. Fox, L., Numerical Solution of Ordinary and Partial Differential Equations, Pergamon Press, London (1962).

35. Frey, T., "Solution of Equations by Stepwise Perturbation," Studia Scientiarum Mathematicarum Nungorica, 3, Nos. 1-3, 93 (1968).

36. Froese, C., "An Evaluation of Runge-Kutta Type Methods for Higher Order Differential Equations," J. of the Assoc. of Comp. Mach., 8, 637 (1961).

37. Gear, C. W., "Rational Approximations by Implicit Runge-Kutta Schemes," BIT, 10, 20 (1970).

38. Gibbons, A., "A Program for the Automatic Integration of Differential Equations Using the Method of Taylor-Series," Comp. J., 3, 108 (1960).

39. Gilbert, E. G., "Dynamic-Error Analysis of Digital and Combined Analog-Digital Computer Systems," Simulation, 6, No. 4, 24 (1966).

40. Gorbunov, A. D. and Shakhov, Y. A., "On the Approximate Solution of Cauchy's Problem With Ordinary Differential Equations to a Given Number of Correct Figures," U.S.S.R. Comput. Math. and Math. Phys., 3, 239 (1963).

41. Gorbunov, A. D. and Tikhonov, A. H., "Estimates of Runge-Kutta Method and the Choice of Optimal Meshes," U.S.S.R. Comput. Math. and Math. Phys., 4, 427 (1964).

42. Grable, E. M., Ramos, S. and Wollridge, D. E., Handbook of Automation Computation and Control, 1, Wiley, New York (1958).

43. Gray, H. J., "Numerical Methods in Digital Real-Time Simulation," Quart. Appl. Math., XII, 133 (1954).

44. Gray, H. J., "Digital Computer Solution of Differential Equations," Proceedings of the Western Joint Computer Conference AFIPS, 13, 87 (1958).

45. Gray, H. J., "Propagation of Truncation Errors in the Numerical Solution of Ordinary Differential Equations by Repeated Closures," J. of the Assoc. of Comp. Math., 52, 1, 5 (1955).

46. Grosswald, E., "Transformation Useful in Numerical Integration Methods," J. of the Society for Industrial and Appl. Math., 7, 76 (1959).

47. Gurk, H. M., "The Use of Stability Charts in the Synthesis of Numerical Quadrature Formulae," Quart. Appl. Math. XIII, 73 (1955).

48. Gurk, H. M. and Rubinoff, M., "Numerical Solution of Differential Equations," Proceedings of the Eastern Joint Computer Conference AFIPS, 6, 58 (1954).

49. Hamming, R. W., Numerical Methods for Scientists and Engineers, McGraw-Hill, New York (1962).

50. Heinrich, K. K., "Runge-Kutta Formulas With Multiple Nodes," Universitat Philosophische Fakultat, Ph.D. Dissertation, 45 (1969).

51. Henrici, P., Discrete Variable Methods in Ordinary Differential Equations, Wiley, New York (1962).

52. Henrici, P., Error Propagation in Difference Methods, Wiley, New York (1963).

53. Hildebrand, F. B., Introduction to Numerical Analysis, McGraw-Hill, New York (1956).

54. Ho, Y. C. and Brentani, P. B., "On Computing Optimal Control With Inequality Constraints," J. SIAM Control Ser. A, 1, 319 (1963).

55. Howard, J. C. and Tashjian, H., "An Algorithm for Deriving the Equations of Mathematical Physics by Symbolic Manipulation," Comm. ACM, 11, 814 (1968).

56. Hull, T. E. and Swenson, J. R., "Test of Probabilistic Models for Propagation of Round-Off Errors," Comm. ACM, 9, 108 (1966).

57. Hurt, J. M., "New Difference Equation Technique for Solving Non-linear Differential Equations," Proceedings of the Spring Joint Computer Conference AFIPS, 25, 169 ( ).

58. Huta, A., Acta Fac. Nat. Univ. Comenian. Math., Tom. 1, Fasc. IV - VI, 201 (1965).

59. Huta, A., "Contribution à la formule de sixième ordre dans la méthode de Runge-Kutta-Nyström," Acta Fac. Nat. Univ. Comenian. Math., 2, 21 (1957).

60. Huta, A., "Une Amélioration de la méthode de Runge-Kutta-Nyström pour la résolution numérique des équations différentielles du premier ordre," Acta Fac. Nat. Univ. Comenian. Math., 1, 201 (1956).

61. Ince, E. L., Ordinary Differential Equations, Dover, New York (1956).

62. Jones, R. E., "Announcement of the Availability of RUNKUT: A Computer Routine for Integrating Systems of First Order Ordinary Differential Equations," Sandia Labs., Albuquerque, New Mexico (1970).

63. Jones, J. P. and Noak, N., "Some Problems in the Numerical Solution of Vortex Motion Equations," Aerodynamics of Rotary Wing and V/STOL Aircraft; Cornell Aeronautical Lab and U. S. Army Aviation Materiel Lab., Symposium, 3rd, Buffalo, N. Y. (1969).

64. Klement'ev, A. F., "Comments on teh Stability of Two-Point Pseudo Runge-Kutta Methods," Differentsial'nye Uravneniia, 5, 751 (1969).

65. Kolmogorov, A. N. and Fomin, S. V., Elements of the Theory of Functions and Functional Analysis, 1, Graylock Press, Rochester (1957).

66. Kopal, Z., Numerical Analysis, 2nd ed., Wiley, New York (1961).

67. Kozlov, L. F. and Krupnik, "Computer Integration of the Falkner-Scan Equation in the Case of a Retarded Flow on a Porous Surface With Suction," Vychislitel'naia i Prikladnaia Matematika, 9, 88 (1969).

68. Kunz, K. S., Numerical Analysis, McGraw-Hill, New York (1957).

69. Kuntzmann, J., "Evaluation de l'erreur Sur un Pas Dans les Methodes a Pas Separes," Chiffres, 2, 97 (1959).

70. Kutta, N., "Beitrag zur Naherungsweisen Integration Totaler Differential-Glerchungen," Z. Math. Phys., 46, 435 (1901).

71. Lance, G. N., Numerical Methods for High Speed Computers, Iliffe and Sons, London (1960).

72. Lapidus, L. and Lewis, R., Optimal Control of Engineering Processes, Random House (Blaisdell), New York (1964).

73. Laurent, P. J., "Methodes Speciales du Type de Runge-Kutta," Premier Congress AFCAL, 27 (1961).

74. Lebedinek, V. N. and Shushkova, "Evaporation and Deceleration of Small Meteoroids," Proceedings of International Astronomical Union Symposium, Tatraska Lomnica, Czechoslovakia (1967).

75. Liniger, W., "Zur Stabilitat der Numerischen Integrations-Methoden fur Differential Gleichungen," Ph.D. Dissertation, University of Lausanne, Zurich (1957).

76. Lomakovich, A. N., "Solution of a Volterra Integro-Differential Equation by the Runge-Kutta-Fehlberg Method," Vychislitel'naia i Prikladnaia Matematika, 7, 64 (1969).

77. Lomakovich, A. N., "Solution of Volterra Integral Equations by the Runge-Kutta-Fehlberg Method," Adademiia Nauk Ukrains Koi RSR, Dopovidi, Seriia A-Fiziko-Tekhnichni i Matematichni Nauki, 31, 301 (1969).

78. Lomakovich, A. N. and Vasilenko, A. P., "Solution of Volterra Integral Equations by a Runge-Kutta Type Method," Differential'nye Uravneniia, 4, 2094 (1968).

79. Lomax, H., "An Operational Unification of Finite Difference Methods for the Numerical Integration of ODE," NASA Tech. Report R-262 (1967).

80. Mah, R. S. H., Michaelson, S., and Sargent, W. H., "Dynamic Behavior of Multi-Component Multi-Step Systems-Numerical Methods for the Solution," Chem. Engr. Sci., 17, 619 (1962).

81. Makinson, G. J., "High Order Implicit Methods for the Numerical Solution of Systems of Differential Equations," Comput. J., 11, 305 (1968).

82. Martens, H. R., "A Comparative Study of Digital Integration Methods," Simulation, 87 (1969).

83. Martin, D. W., "Runge-Kutta Methods for Integrating Differential Equations on High Speed Digital Computers," Computer Journal, 1, 3, 118 (1958).

84. Mason, C., "A Numerical Solution of the Conduction Problem With Radiating Surface," AIAA Journal, 7, 2182 (1969).

85. Merson, R. N., "On Some Developments in the Study of Processes for Solving Ordinary Differential Equations," An internal memorandum of the Royal Aircraft Est. (1960).

86. Miller, J. C. P., "The Numerical Solution of Ordinary Differential Equations," Numerical Analysis (J. Walsh, ed.), Ch. 4, Thompson Book (1967).

87. Milne, W. E. and Reynolds, R. R., "Fifth-Order Methods for the Numerical Solution of Ordinary Differential Equations," J. of the Assoc. for Comp. Mach., 9, 1, 64 (1962).

88. Milne, W. E. and Reynolds, R. R., "Stability of a Numerical Solution of Differential Equations," J. of the Assoc. for Comp. Mach., 6, 2, 196 (1959).

89. Milen, W. E. and Reynolds, R. R., "Stability of a Numerical Solution of Differential Equations - Part II," J. of the Assoc. for Comp. Mach., 7, 1, 46 (1960).

90. Moore, R. E., Internal Analysis, Prentice-Hall, New Jersey (1966).

91. Morel, H., "Evaluation de L'erreur Sur un Pas Dans la Methode de Runge-Kutta," Comptes rendus de l'Academie des Sciences Paris, 243, 1999 (1965).

92. Muir, T., A Treatise on the Theory of Determinants, Dover, New York, 437 (1960).

93. Nickel, K. and Rieder, P., "A New Procedure Similar to the Runge-Kutta Method, Numerical Mathematics: Theory of Approximation of Differential Equations," Edited by L. Collatz, et al., Internationale Schriftenreihe zur Numerischen Mathematik, 9, 83 (1968).

94. Niesse, D., "Letter Dated October 15, 1969," Simulation, 9, 168 (1969)

95. Niesse, D., "Technical Comment," Simulation, 9, 168 (1969).

96. Nikolaen, V., "The Solution of Systems of Ordinary Differential Equations by Expansion in Power Series on High Speed Computers," U.S.S.R. Comp. Math. and Math. Phys., 5, 608 (1965).

97. Nordsieck, A., "On Numerical Integration of Ordinary Differential Equations," Math. of Comp., 16, 22 (1962).

98. Nugeyre, J., "Un Procede Mixte (Runge-Kutta, pas lies) d'Integration des Systems Differentield du Type x" = X(x,t)," Chiffres, 4, 55 (1961).

99. Nystrom, E., "Uber die Numerische Integration of Differential Equations in an Automatic Digital Computing Machine," Proc. Camb. Phil. Soc., 47, 96 (1951).

100. Nystrom, E., "Uber die Numerische Integration von Differentialgleichungen," Acta Soc. Sci. Fenn., 13, 50 (1925).

101. Obrechkoff, N., "Sur les Quadratures Mecaniques (Bulgarian, French Summary)," Spisanie Bulgar. Akad. Nauk., 65, 191 (1942).

102. Polya, G. and Szego, G., "Aufgaben und Lehrsatze aus der Analysis I," Grund. Math. Wiss., 19, 301 (1925).

103. Pope, D., "An Exponential Method of Numerical Integration of Ordinary Differential Equations," Communications of the Assoc. for Comp. Mach., 6, 491 (1963).

104. Radau, M., "Etude sur les Formules d'Approximation qui Servent a Calculer la Valeur Numerique d'une Integrale Definie," J. Math. Pures Appl., 6, 283 (1880).

105. Ralston, A. and Wilf, H., Mathematical Methods for Digital Computers, Wiley, New York, 128 (1960).

106. Rice, J., "Split Runge-Kutta Method for Differential Equations," J. of Res. of the Nat. Bureau of Standards, 64B, 151 (1960).

107. Richter, W., "Estimation de l'erreur Commise Dans la Methode de M. W. E. Milne pour l'Integration d'un Systeme de n Equations Differentielles du Premier Ordre," Bull. Soc. Neuchaleloise des Sci. Naturellis, 75, 6 (1952).

108. Rubinoff, M., Computer Handbook, McGraw-Hill, New York, 49 (1962).

109. Rubinoff, M., "Digital Computers for Real-Time Simulation," J. of the Assoc. for Comp. Mach., 1, 186 (1955).

110. Runge, C., Math. Ann. 46, 167 (1895).

111. Sammet, J., "Formula Manipulation Compiler," Datamation, 12, 32 (1966).

112. Sarafyan, D., "Determination of Optimal Step-Size for Runge-Kutta Processes," Louisiana State Univ., Tech. Rep. No. 54, Forthcoming.

113. Sarafyan, D., "Estimation of Errors for the Approximate Solution of Differential Equations and Their Systems," Louisiana State Univ., Tech. Rep. No. 15 (1966).

114. Sarafyan, D., "Composite and Multistep Runge-Kutta Formulas," Louisiana State Univ., Tech. Rep. No. 18 (1966).

115. Scheid, F., Theory and Problems of Numerical Analysis, McGraw-Hill, New York, 202 (1968).

116. Scholz, S., "A Runge-Kutta Method with a Variable Parameter Alpha," ZAMM, 49, 517 (1969).

117. Scraton, R., "The Numerical Solution of Second-Order Differential Equations not Containing the First Derivative Explicitly," Comp. J., 6, 368 (1963).

118. Sinha, K., "Laminar Incompressible Boundary Layer in a Divergent Channel with Homogeneous Suction at the Wall," Nat. Inst. of Sci. of India, Proceedings, Part A--Physical Sciences, Supplement 2, 35, 162 (1969).

119. Sloan, I., "Errors in Numerov and Runge-Kutta Methods," J. of Comp. Physics, 2, 414 (1968).

120. Sommer, D., "New Implicit Runge-Kutta Formulas and Possible Applicationf of Them," Aachen, Technische Hocschule Mathematisch-Naturwissenschaftliche Fakultat, Doktor der Naturwissenschaftlichen Dissertation, 122 (1967).

121. Squier, D., "One-step Methods for Ordinary Differential Equations," Num. Math., 13, 176 (1969).

122. Stewart, N., "The Comparison of Numerical Methods for Ordinary Differential Equations," Ph.D. Thesis, Dept. of Comuter Sci., Univ. of Toronto (1968).

123. Stimberg, C., "Runge-Kutta Verfahren zur Numerischen Behandlung von Anfangswertproblemen bie Partiellen Hyperbolischen Differentialgleichungen," Dissertation, Aachen (1966).

124. Stimberg, C., "Simplified Derivation of Runge-Kutta Techniques," ZAMM, 47, 413 (1967).

125. Stimberg, C., "Vereinfächte Herleitung von Runge-Kutta Verfahren," ZAMM, 47, 413 (1967).

126. Tavernini, L., "On UNIP and the Construction of Digital Simulation Programs," Simulation, 7, 263 (1966).

127. Thompson, W. E., "Solution of Linear Differential Equations," Comput. J., 10, 417 (1967).

128. Treanor, C. E., "A Method for the Numerical Integration of Coupled First-Order Differential Equations With Greatly Different Time Constants," Math. Comp., 20, 39 (1966).

129. Warten, R. M., "Automatic Step-Size Control for Runge-Kutta Integration," IBM J. Res. Develop., 2, 340 (1963).

130. Waters, J., "Methods of Numerical Integration Applied to a System Having Trivial Function Evaluations," Comm. ACM, 9, 293 (1966).

131. Whitney, D. E., "Propagated Error Bounds for Numerical Solution of Transient Response," Proc. IEEE, 54, 8, 1084 (1966).

132. Whitney, D. E., "More About Similarities Between Runge-Kutta and Matrix Exponential Methods for Evaluating Transient Response," Proc. IEEE, 57, 11, 2053 (1969).

133. Wedlund, O. B., "A Note on Unconditionally Stable Linear Multistep Methods," BIT, 7, 65 (1967).

134. Wilf, S., "A Stability Criterion for Numerical Integration," J. of the Assoc. for Comp. Mach., 6, 3, 363 (1959).

135. Zee, C., "On Solving Second-Order Nonlinear Differential Equations," Quart. Appl. Math., 22, 71 (1964).

136. Zonneveld, J. A., "Automatic Numerical Integration," Math. Centre Tracts, 8, Mathematische Centrum, Amsterdam, 1612 (1964).

137. Zurmuhl, R., "Runge-Kutta-Verfahren zur Numerishen Integration von Differential Gleichungen n-ter Ordung," ZAMM, 28, 173 (1948).

APPENDIX II

MULTI-STEP METHODS

This appendix lists many of the multi-step methods that are in the current literature.

The most common form of multi-step methods is

$$\sum_{i=0}^{k} (\alpha_i y_{n+1-i} + h\ \beta_i\ y'_{n+1-i}) = 0.$$

When $\beta_0 = 0$, then the equation becomes an explicit equation for y_{n+1}; such equations are called Predictor (P) equations. When $\beta_0 \neq 0$, the equation becomes an implicit equation for y_{n+1}, i.e., the equation must be iterated for y_{n+1}. Such equations are called Corrector (C) equations.

Although the Predictor (P) equation can be used with one derivative evaluation (E) at each step of the numerical solution, usually a combination of the P and C equations is used to solve the initial value problem. The various modes of these P-C combinations are PECE, PEC, $PE(CE)^s$ and $P(EC)^s$. The first two modes are not iterative methods. First, y_{n+1} is predicted. Then, there is a derivative evaluation of y'_{n+1}. Then y_{n+1} is corrected using y'_{n+1}. For the first mode a new y'_{n+1} is evaluated. The third and fourth modes simply iterate s times on the corrector.

The Predictor and Corrector equations presented below are arranged in related groups. The format of each equation is in the following order: 1) the equation number S, 2) the order m and number of backsteps n, written (m,n), and 3) the α and β coefficients. The general format is

S. (m,n) $\left(\frac{1}{\alpha_0}\right)$ $(\alpha_1 + \alpha_2 + \cdots + \alpha_n)$

$$+ \left(\frac{1}{\beta_0}\right) (\beta_0 + \beta_1 + \cdots + \beta_n).$$

Euler Predictor: [556]

1. (1,1) $\left(\frac{1}{1}\right)(1) + \left(\frac{1}{1}\right)(0 + 1)$

Milne Predictor: [556]

2. (4,3) $\left(\frac{1}{1}\right)(0 + 0 + 0 + 1) + \frac{1}{3}\ (0 + 8 - 4 + 8)$

Millman-Klopfenstein Predictor: [277]

3. (4,4) $\left(\frac{1}{1}\right)(-.29 - 15.39 + 12.13 + 4.55) + \left(\frac{1}{1}\right)(0 + 2.27 + 6.65 + 13.91 + 0.69)$

Craine-Klopfenstein Predictor: [95]

4. (4,4) $\left(\frac{1}{1}\right)(1.547 - 1.867 + 2.017 - 0.6973) + \left(\frac{1}{1}\right)(0 + 2.002 - 2.031 + 1.818 - 0.7143)$

Hermite Extrapolation Predictors: [556]

5. (3,2) $\left(\frac{1}{1}\right)(-4 + 5) + \left(\frac{1}{1}\right)(0 + 4 + 2)$
6. (4,3) $\left(\frac{1}{1}\right)(-9 + 9 + 1) + \left(\frac{1}{1}\right)(0 + 6 + 6)$
7. (5,3) $\left(\frac{1}{1}\right)(-18 + 9 + 10) + \left(\frac{1}{1}\right)(0 + 9 + 18 + 3)$
8. (7,4) $\left(\frac{1}{3}\right)(-128 - 108 + 198 + 47) + \left(\frac{1}{1}\right)(0 + 16 + 72 + 48 + 4)$
9. (9,5) $\left(\frac{1}{6}\right)(-475 - 1400 + 600 + 1150 + 131) + \left(\frac{1}{1}\right)(0 + 25 + 200 + 300 + 100 + 5)$

Nystrom Predictors: [556]

10. (2,1) $(\frac{1}{1})(0 + 1) + (\frac{1}{1})(0 + 2)$

11. (3,2) $(\frac{1}{1})(0 + 1) + (\frac{1}{1})(0 + 2 + 0)$

12. (4,3) $(\frac{1}{1})(0 + 1) + (\frac{1}{3})(0 + 7 - 2 + 1)$

13. (5,4) $(\frac{1}{1})(0 + 1) + (\frac{1}{3})(0 + 8 - 5 + 4 - 1)$

14. (6,5) $(\frac{1}{1})(0 + 1) + (\frac{1}{90})(0 + 269 - 266 + 294 - 146 + 29)$

Same Predictors from method of undetermined coefficient: [556]

15. (5,4) $(\frac{1}{1})(0 + 0 + 1) + \frac{1}{8}(0 + 21 - 9 + 15 - 3)$

16. (5,4) $(\frac{1}{1})(0 + 0 + 0 + 1) + \frac{1}{24}(0 + 9 - 32 + 64)$

17. (5,3) $(\frac{1}{1})(-9 + 9 + 1) + (\frac{1}{1})(0 + 6 + 6)$

18. (5,3) $(\frac{1}{1})(-8 + 9) + (\frac{1}{3})(0 + 17 + 14 - 1)$

19. (5,3) $(\frac{1}{1})(-7 + 9 - 1) + (\frac{1}{3})(0 + 16 + 10 - 2)$

20. (4,3) $(\frac{1}{1})(-54 + 45 + 10) + (\frac{1}{1})(0 + 24 + 42)$

21. (4,2) $(\frac{1}{1})(-4 + 5) + (\frac{1}{1})(0 + 4 + 2)$

22. (5,4) $(\frac{1}{1})(-1 + 0 + 1 + 1) + (\frac{1}{1})(0 + 3 + 0 + 3)$

Schoen Predictors: [461]

23. (5,5) $(\frac{1}{1})(-0.01745885 + 1.29864818 - 0.13318934 - .14799999)$

$+ (\frac{1}{1})(0 + 3.0001344 - 3.05980708 + 2.98844518 - 1.66187410$

$+ 0.32137111)$

24. (6,6) $(\frac{1}{1})(-.92790378 + 2.05094161 + .54664912 - 0.67100875$
$+ 0.0005296 + 0.00126884) + (\frac{1}{1})(0 + 3.60328436$
$- 3.60284254 + 5.40345116 - 4.77344249 + 1.80481406$
$- 0.29749495)$

Krough Predictors: [556]

25. (4,4)(A) $(\frac{1}{2})(1 + 1) + (\frac{1}{48})(0 + 119 - 99 + 69 - 17)$

26. (4,4)(B) $(\frac{1}{7})(4 + 3) + (\frac{1}{42})(0 + 103 - 88 + 61 - 15)$

27. (5,5) $(\frac{1}{31})(-1 + 32) + (\frac{1}{7440})(0 + 22{,}321 - 21{,}774 + 24{,}216$
$- 12{,}034 + 2391)$

28. (6,6) $(\frac{1}{12})(-11 + 23) + (\frac{1}{17{,}280})(0 + 62{,}248 - 62{,}255 + 101{,}430$
$- 76{,}490 + 30{,}545 - 5079)$

29. (7,7) $(\frac{1}{10})(-21 + 31) + (\frac{1}{604{,}800})(0 + 2{,}578{,}907 - 2{,}454{,}408$
$+ 5{,}615{,}199 - 5{,}719{,}936 + 3{,}444{,}849 - 1{,}149{,}048$
$+ 164{,}117)$

Adams-Bashforth Predictors: [556]

30. (2,2) $(\frac{1}{1})(1) + (\frac{1}{2})(0 + 3 - 1)$

31. (3,3) $(\frac{1}{1})(1) + (\frac{1}{12})(0 + 23 - 16 + 5)$

32. (4,4) $(\frac{1}{1})(1) + (\frac{1}{24})(0 + 55 - 59 + 37 - 9)$

33. (5,5) $(\frac{1}{1})(1) + (\frac{1}{720})(0 + 1901 - 2774 + 2616 - 1274 + 251)$

34. (6,6) $(\frac{1}{1})(1) + (\frac{1}{1440})(0 + 4277 - 7923 + 9982 - 7298 + 2887 - 475)$

35. (7,7) $(\frac{1}{1})(1) + (\frac{1}{60,480})(0 + 198,721 - 447,288 + 705,549 - 688,256$
$+ 407,139 - 134,472 + 19,087)$

36. (8,8) $(\frac{1}{1})(1) + (\frac{1}{120,960})(0 + 434,241 - 1,162,169 + 2,183,877$
$- 2,664,477 + 2,102,243 - 1,041,723 + 295,767$
$- 36,799)$

Milne Corrector: [556]

37. (4,2) $(\frac{1}{1})(0 + 1) + (\frac{1}{3})(1 + 4 + 1)$

Hamming Corrector: [556]

38. (4,2) $(\frac{1}{8})(9 + 0 - 1) + (\frac{1}{8})(3 + 6 - 3)$

Milne-Reynolds Corrector: [361]

39. (5,3) $(\frac{1}{8})(1 + 7) + (\frac{1}{192})(65 + 243 + 51 + 1)$

Correctors from method of undetermined coefficients: [556]

40. (4,2) $(\frac{1}{5})(4 + 1) + (\frac{1}{5})(2 + 4)$

41. (5,3) $(\frac{1}{1})(0 + 0 + 1) + (\frac{1}{8})(3 + 9 + 9 + 3)$

42. (5,3) $(\frac{1}{3})(1 + 1 + 1) + (\frac{1}{36})(13 + 39 + 15 + 5)$

43. (5,3) $(\frac{1}{2})(1 + 1) + (\frac{1}{48})(17 + 51 + 3 + 1)$

44. (5,3) $(\frac{1}{3})(2 + 1) + (\frac{1}{72})(25 + 91 + 43 + 9)$

45. (5,3) $(\frac{1}{17})(9 + 9 - 1) + (\frac{1}{17})(6 + 18)$

46. (5,3) $(\frac{1}{9})(9 + 1 - 1) + (\frac{1}{27})(10 + 22 - 8)$

47. (5,3) $(\frac{1}{7})(9 - 1 - 1) + (\frac{1}{21})(8 + 14 - 10)$

48. (5,3) $(\frac{1}{31})(45 - 9 - 5) + (\frac{1}{31})(12 + 18 + 18)$

49. (5,3) $(\frac{1}{5})(9 - 3 - 1) + (\frac{1}{5})(2 + 2 - 4)$

Norsett A(α) stable Correctors: [389]

50. (4,3) $(\frac{1}{11})(18 - 9 + 2) + (\frac{1}{11})(6)$

51. (5,4) $(\frac{1}{25})(48 - 36 + 16 - 3) + (\frac{1}{25})(12)$

52. (6,5) $(\frac{1}{137})(300 - 300 + 200 - 75 + 12) + (\frac{1}{137})(60)$

53. (7,6) $(\frac{1}{147})(360 - 450 + 400 - 225 + 72 - 10) + (\frac{1}{147})(60)$

Adams-Moulton Correctors: [422]

54. (2,1) $(\frac{1}{1})(1) + (\frac{1}{2})(1 + 1)$

55. (3,2) $(\frac{1}{1})(1) + (\frac{1}{12})(5 + 8 - 1)$

56. (4,3) $(\frac{1}{1})(1) + (\frac{1}{24})(9 + 19 - 5 + 1)$

57. (5,4) $(\frac{1}{1})(1) + (\frac{1}{720})(251 + 646 - 264 + 106 - 19)$

58. (6,5) $(\frac{1}{1})(1) + (\frac{1}{1440})(475 + 1427 - 798 + 482 - 173 + 27)$

59. (7,6) $(\frac{1}{1})(1) + (\frac{1}{60,480})(19,087 + 65,112 - 46,461 + 37,504 - 20,211 + 6312 - 863)$

60. (8,7) $(\frac{1}{1})(1) + (\frac{1}{120,960})(36,799 + 139,849 - 121,797 + 123,133 - 88,547 + 41,499 - 11,351 + 1375)$

61. (9,8) $(\frac{1}{1})(1) + (\frac{1}{3,628,800})(1,070,017 + 4,467,094 - 4,604,594$
$+ 5,593,358 - 5,033,120 + 3,146,338 - 1,291,214$
$+ 312,874 - 339,533)$

62. (10,9) $(\frac{1}{1})(1) + (\frac{1}{7,257,600})(2,082,753 + 2,449,717 - 11,271,304$
$+ 16,002,320 - 17,283,646 + 13,510,082 - 7,394,032$
$+ 2,687,864 - 583,435 + 57,281)$

Fehlberg Correctors: [422]

63. (3,2) $(\frac{1}{5})(4 + 1) + (\frac{1}{5})(2 + 4)$

64. (4,3) $(\frac{1}{17})(9 + 9 - 1) + (\frac{1}{17})(6 + 18)$

65. (5,4) $(\frac{1}{27})(16 + 0 + 0 + 11) + (\frac{1}{9})(3 + 10 + 0 + 6 + 1)$

66. (6,5) $(\frac{1}{21,319})(0 + 0 + 13,300 + 8775 - 756) + (\frac{1}{21,319})(6720$
$+29,700 + 13,500 + 21,300)$

67. (7,6) $(\frac{1}{1000})(243 + 0 + 125 + 0 + 0 + 632) + (\frac{1}{400})(120 + 567$
$+ 0 + 600 + 0 + 405 + 72)$

68. (8,7) $(\frac{1}{82,490,048})(0 + 0 + 28,107,625 + 0 + 29,545,048$
$+ 2,783,200 - 1,994,625)$
$+ (\frac{1}{82,490,048})(24,299,520 + 126,015,750$
$+ 21,168,000 + 0 + 70,634,970)$

Wesson Correctors: [422]

69. (3,2) $(\frac{1}{2})(1 + 1) + (\frac{1}{8})(3 + 8 + 1)$

70. (3,2) $(\frac{1}{25})(9 + 16) + (\frac{1}{300})(109 + 328 + 55)$

71. (5,4) $(\frac{1}{16})(1 + 0 + 0 + 15) + (\frac{1}{11,520})(3611 + 16,006 + 5496 + 15,466 + 3341)$

72. (5,4) $(\frac{1}{16})(8 + 2 + 1 + 5) + (\frac{1}{2304})(767 + 2638 + 168 + 1282 + 185)$

73. (5,5) $(\frac{1}{1})(1) + (\frac{1}{2,440,080})(859,838 + 2,143,299 - 802,706 + 267,244 - 18,396 - 9199)$

74. (8,8) $(\frac{1}{1})(1) + (\frac{1}{3,628,800})(1,147,591 + 3,846,502 - 2,432,522 + 1,251,214 + 397,060 - 1,197,806 + 880,858 - 307,718 + 43,621)$

Rodabaugh-Wesson Correctors: [422]

75. (5,4) $(\frac{1}{16})(1 + 2 + 4 + 9) + (\frac{1}{11,520})(3703 + 15,518 + 6168 + 10,898 + 1873)$

76. (7,6) $(\frac{1}{64})(1 + 2 + 4 + 8 + 16 + 33) + (\frac{1}{430,080})(128,627 + 642,168 + 130,167 + 693,632 + 143,137 + 399,240 + 61,469)$

77. (8,8) $(\frac{1}{1})(1) + (\frac{1}{3,628,800})(1,111,267 + 413,709 - 3,449,594 + 3,285,358 - 2,145,620 + 836,338 - 136,214 - 17,126 + 7297)$

78. (9,8) $(\frac{1}{2,560,016})(9784 + 20,133 + 41,040 + 79,775 + 159,816 + 319,691 + 639,792 + 1,289,985) + (\frac{1}{2,560,016})(725,340 + 4,150,740 - 280,710 + 6,541,620 - 1,808,250 + 5,630,940 + 244,290 + 2,458,620 + 345,330)$