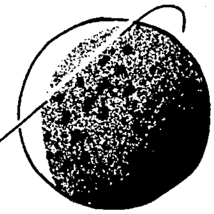
2
m1X

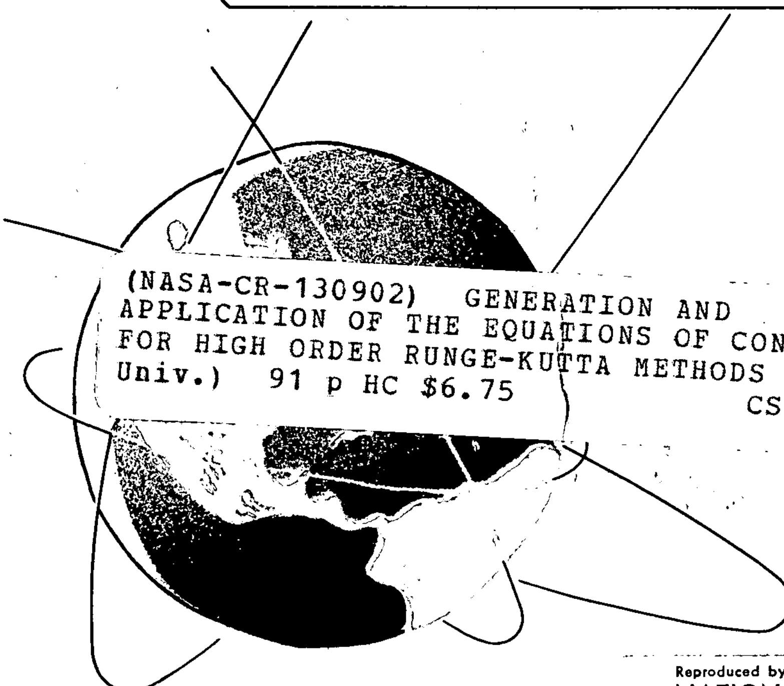


GENERATION AND APPLICATION OF THE EQUATIONS OF CONDITION FOR HIGH ORDER RUNGE-KUTTA METHODS

BY

DENNIS CLYDE HALEY

AMRL 1042 MAY, 1972



(NASA-CR-130902) GENERATION AND APPLICATION OF THE EQUATIONS OF CONDITION FOR HIGH ORDER RUNGE-KUTTA METHODS (Texas Univ.) 91 p HC $6.75 CSCL 12A

N73-24613

Unclas
G3/19 04057

APPLIED MECHANICS RESEARCH LABORATORY
THE UNIVERSITY OF TEXAS AT AUSTIN AUSTIN, TEXAS

91

GENERATION AND APPLICATION OF THE EQUATIONS OF CONDITION FOR HIGH ORDER RUNGE-KUTTA METHODS

Dennis Clyde Haley
The University of Texas at Austin
Austin, Texas

AMRL 1042
May, 1972

Applied Mechanics Research Laboratory
The University of Texas at Austin
Austin, Texas

This report was prepared under

Grant No. NAS 8-27931

for the

National Aeronautics and Space Administration
Marshall Space Flight Center

by the

Applied Mechanics Research Laboratory
The University of Texas at Austin

Austin, Texas

under the direction of

Dale G. Bettis
Assistant Professor

and

Bob E. Schutz
Assistant Professor

ACKNOWLEDGMENTS

The author would like to express his appreciation to Dr. D. G. Bettis and Dr. B. E. Schutz for serving as his supervising professors and for guiding the research presented in this study. He would also like to express his gratitude to Dr. J. Waldvogel, Mr. T. W. Feagin, and Mr. W. H. Moore for their helpful suggestions and contributions. The author would especially like to thank his wife for her support and encouragement throughout the writing of this paper.

This research was supported in part by the National Aeronautics and Space Administration Marshall Space Flight Center under Grant No. NAS 8-27931. Financial support for graduate study from the National Science Foundation is also gratefully acknowledged.

March 20, 1972

ABSTRACT

The purpose of this thesis is to develop the equations of condition necessary for determining the coefficients for Runge-Kutta methods used in the solution of ordinary differential equations. The equations of condition are developed for Runge-Kutta methods of order four through order nine. Once developed, these equations are used in a comparison of the local truncation errors for several sets of Runge-Kutta coefficients for methods of order three up through methods of order eight.

TABLE OF CONTENTS

LIST OF TABLES

CHAPTER I

INTRODUCTION

1.1 Description of the Problem

The purpose of this thesis was to develop the equations of condition necessary to determine the coefficients for Runge-Kutta methods used in the solution of ordinary differential equations. The equations of condition were developed for Runge-Kutta methods of order four through order nine. Once developed, the equations were used in a comparison of local truncation errors for several sets of Runge-Kutta coefficients for methods of order three through order eight. The equations of condition were generated by the computer using the algebraic manipulation language SYMBAL[3] and the CDC 6600/6400 computer system at the University of Texas at Austin.

1.2 Motivation

Numerical solutions to ordinary differential equations play a large role in the study of science and engineering. The integration of spacecraft trajectories in orbital mechanics, for example, requires accurate and efficient numerical integration techniques. The Runge-Kutta type methods for solving systems of ordinary differential equations are widely used because of their simplicity of application. Any studies of existing Runge-Kutta methods, or attempts to produce new or better Runge-Kutta methods will result in a better understanding of the methods, and will lead to the development of more efficient techniques for solving the ordinary differential equations of science and engineering.

CHAPTER II

GENERATION OF THE EQUATIONS OF CONDITION FOR RUNGE-KUTTA COEFFICIENTS WITH TAYLOR SERIES EXPANSIONS

2.1 Description of the Method

Consider the ordinary differential equation

$$\frac{dy}{dx} = y'(x) = f(x, y) \tag{1}$$

with initial conditions $y(x_o) = y_o$, where y and f may be vectors, y is the dependent variable, and x is the independent variable. A solution of the form $y = y(x)$ which satisfies the initial conditions and Equation (1) is desired. The existence and uniqueness of a solution is assumed since there exist theorems[8] which state that if $f(x, y)$ is sufficiently well behaved near a point (x, y), then Equation (1) has a solution that passes through the point and is unique in a neighborhood of the point.

A solution of Equation (1) can be found using a Taylor series expansion of y about $y = y_o$ in the form

$$y(x_o + \gamma h) = y_o + h\gamma y_o' + \frac{h^2\gamma^2}{2!} y_o'' + \frac{h^3\gamma^3}{3!} y_o''' + \ldots. \tag{2}$$

for any γ for which the series converges. This technique requires, however, that the derivatives of $y(x)$ be known up to some desired order. Since $f(x, y)$ is a function of both x and y, these derivatives may become quite complicated. It is convenient to introduce the following notation:

$$\begin{aligned} y' &= f^{[0]} = f \quad, \\ y'' &= f^{[1]} \quad, \\ y''' &= f^{[2]} \quad, \\ &\vdots \end{aligned}$$

where $$f^{[1]} = f_x + f_y f \quad ,$$

where f_x and f_y represent partial derivatives of f(x, y) with respect to x and y respectively. Successive derivatives of y(x) can be generated using

$$f^{[j+1]} = f_x^{[j]} + f^{[j]} f.$$

If the function f(x, y) is complicated, this procedure can become prohibitively difficult.

Runge[17] was the first to point out that it was possible to avoid the successive differentiation required in the Taylor series solution while still preserving the accuracy. Runge bypassed the derivatives in the Taylor series solution by using evaluations of the function f(x, y) within the interval (x_o, y_o) to $(x_o + h, y(x_o + h))$. Runge's ideas were applied to first order differential equations in a more accurate form by Heun[7] and Kutta[10], and extended to second order differential equations by Nystrom[14]. Zurmuhl[20] continued the extension to nth order.

Considering only the first order system of differential equations, the problem formulation now becomes

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

$$y(x_o) = y_o \quad ,$$

with $$f_o = f(x_o, y_o), \text{ and}$$

$$f_k = f(x_o + \alpha_k h, y_o + h \sum_{\lambda=0}^{k-1} \beta_{k\lambda} f_\lambda), \; k = 1, 2, 3, \ldots, n,$$

where n + 1 equals the number of function evaluations required in the interval. The solution is then given by

$$Y(x) = y_o + h \sum_{k=0}^{n} c_k f_k + O(h^{m+1}) \quad , \tag{3}$$

where m is the order to which the Runge-Kutta formula agrees with the Taylor series, and h is the integration step-size which implies a value $\gamma = 1$ in Equation (2).

In order to obtain Y(x) from the Runge-Kutta scheme, it is necessary to determine the coefficients α, β, and c in such a way that the Runge-Kutta solution is equivalent to the Taylor series solution to some order m. In order to accomplish this, the common approach in the past has been to equate the expression for $y(x_o + h)$ obtained from the Taylor series in Equation (2) and the expression for Y(x) obtained from the Runge-Kutta formulation given in Equation (3). From this equivalence the coefficients of corresponding powers of h are compared yielding a system of nonlinear algebraic equations, which are referred to as equations of condition, from which the unknown coefficients may be determined for the classical Runge-Kutta formula of order m.

For the solution of this set of equations of condition to yield coefficients which produce a method of order m, there is a minimum number of function evaluations required for that order. For example, since a seventh order method requires at least nine function evaluations, a method which has eight function evaluations can at best be a sixth order method although only seven function evaluations are actually required for a sixth order method. Since many of the existing methods use more than the minimum number of function evaluations for a given order, the number of function evaluations used for each order in this study was chosen equal to that of the method using the most function evaluations for that order. Table 1 shows the minimum number of function evaluations, the number of function evaluations used in this study, and the number of new equations of condition expected for methods of order three through order nine.

Table 1.

Number of Function Evaluations and Equations of Condition

Order	Min. No. of Func. Eval. Reqd.	No. of Func. Eval. Used	No. New Eqs. Expected
3	3	4	2
4	4	5	4
5	6	6	9
6	7	8	20
7	9	11	48
8	11	13	115
9	--	17	286

It will be advantageous to give an example to illustrate this procedure, which, although basically simple and straightforward, is extremely tedious. Consider the development of the equations of condition for a Runge-Kutta formula of third order with three function evaluations.

The Taylor series solution becomes:

$$y(x_o + h) = y_o + hy'_o + \frac{h^2}{2} y''_o + \frac{h^3}{6} y'''_o + O(h^4) \quad ,$$

where

$$y'_o = f_o \quad ,$$

$$y''_o = (f_x + f_y f)_o \quad ,$$

$$y'''_o = (f_{xx} + 2ff_{xy} + f^2 f_{yy} + f_x f_y + ff_y{}^2)_o \quad .$$

By expanding the function f_k in a Taylor series of two variables, the Runge-Kutta solution yields:

$$Y(x) = y_o + h(c_o f_o + c_1 f_1 + c_2 f_2) + O(h^4) \quad ,$$

where $\quad f_o = f_o \quad ,$

$$f_1 = f(x_o + \alpha_1 h,\ y_o + h\beta_{10}f_o)$$
$$= f_o + \alpha_1 hf_x + h\beta_{10}f_y f + \tfrac{1}{2}\alpha_1^2 h^2 f_{xx} + \tfrac{1}{2}\beta_{10}^2 h^2 f^2 f_{yy} + O(h^3),$$
$$f_2 = f(x_o + \alpha_2 h,\ y_o + h(\beta_{20}f_o + \beta_{21}f_1))$$
$$= f_o + \alpha_2 hf_x + h[\beta_{20}f + \beta_{21}(f + \alpha_1 hf_x + h\beta_{10}f_y f)]f_y$$
$$+ \tfrac{1}{2}\alpha_2^2 h^2 f_{xx} + \tfrac{1}{2}h^2[\beta_{20}f_o + \beta_{21}(f_o + \alpha_1 hf_x)]^2 f_{yy}$$
$$+ \alpha_2 h^2[\beta_{20}f + \beta_{21}(f_o + \alpha_1 hf_x)]f_{xy} + O(h^3).$$

Setting $y(x_o + h)$ equal to $Y(x)$ gives:

$$y_o + hf_o + h^2[\tfrac{1}{6}(f_x + f_y f)] + h^3[\tfrac{1}{6}(f_{xx} + 2f_{xy}f + f^2 f_{yy} + f_x f_y + ff_y^2)] + O(h^4)$$
$$= y_o + h\{c_o f_o + c_1[f_o + h(\alpha_1 f_x + \beta_{10}f_y f) + \tfrac{1}{2}h^2(\alpha_1^2 f_{xx} + \beta_{10}^2 f^2 f_{yy})$$
$$+ h(\alpha_1\beta_{10}ff_{xy})] + c_2[f_o + h(\alpha_2 f_x + \beta_{20}ff_y + \beta_{21}f_y(f + \alpha_1 hf_x$$
$$+ \beta_{10}f_y fh)) + \frac{1}{2}h^2(\alpha_2^2 f_{xx} + f^2 f_{yy}(\beta_{20} + \beta_{21})^2) + h^2(\alpha_2\beta_{20}ff_{xy}$$
$$+ \alpha_2\beta_{21}ff_{xy})]\} + O(h^4).$$

Comparing similar powers of h:

h^0 $\qquad y_o = y_o$;

h^1 $\qquad f_o = c_o f_o + c_1 f_o + c_2 f_o$;

h^2 $\qquad \frac{1}{2}f_x = c_1\alpha_1 f_x + c_2\alpha_2 f_x$,

$$\frac{1}{2}f_y f = c_1\beta_{10}f_y f + c_2(\beta_{20} + \beta_{21})\, ff_y \quad ;$$

h^3 $\qquad \frac{1}{6}f_{xx} = \frac{1}{2}(\alpha_1^2 c_1 + \alpha_2^2 c_2)f_{xx}$,

$$\frac{1}{6}f^2 f_{yy} = \frac{1}{2}(c_1\beta_{10}^2 + c_2(\beta_{20} + \beta_{21})^2)f^2 f_{yy} \quad ,$$

$$\frac{1}{6}f_x f_y = c_2\beta_{21}\alpha_1 f_x f_y \quad ,$$

$$\frac{1}{6}f_y^2 f = c_2\beta_{21}\beta_{10}f_y^2 f \quad ,$$

$$\frac{1}{3}f_{xy}f = [c_2\alpha_2(\beta_{20} + \beta_{21}) + c_1\alpha_1\beta_{10}]f_{xy}f \quad .$$

An examination of these equations reveals that

$$\alpha_1 = \beta_{10} \text{ , and}$$

$$\alpha_2 = \beta_{20} + \beta_{21} .$$

In general, this requirement can be stated

$$\alpha_i = \sum_{j=0}^{i-1} \beta_{ij} .$$

The equations of condition for the third order Runge-Kutta formula then become:

$$1 = c_o + c_1 + c_2 \quad ,$$

$$\frac{1}{2} = c_1\alpha_1 + c_2\alpha_2 \quad ,$$

$$\frac{1}{3} = c_1\alpha_1^2 + c_2\alpha_2^2 \quad ,$$

$$\frac{1}{6} = c_2\alpha_1\beta_{21} \quad .$$

By extending this procedure to higher orders it is observed that as the order and number of function evaluations increase, the expressions for $f^{[i]}$ in the Taylor series and f_i in the Runge-Kutta formula become more and more complex and grow enormously in length.

2.2 Computational Procedure and Limitations

In order to generate the equations of condition for Runge-Kutta methods, the computer and the algebraic manipulation language, SYMBAL, were employed. From the example just given it is observed that the generation of the equations of condition for even the fourth and fifth order methods is an extremely tedious task. Therefore, the computational power of the computer was used to develop these equations, beginning with the fourth order method, in hopes of eventually developing the equations of condition for high order Runge-Kutta methods.

The SYMBAL (SYMBolic ALgebra) language used was developed by M. E. Engeli[4] and is implemented on the CDC 6600 at The University of Texas at Austin and at the Swiss Federal Institute of Technology in Zürich. The language is a generalization of ALGOL 60 with the manipulation of unrestricted algebraic expressions of which numbers represent only a very special case. Although the language SYMBAL may not be widely available, it is similar to several other algebraic manipulation languages in use which may be more familiar to the reader.

The original program used in this study is shown in Appendix I and appears in the SYMBAL manual[4] where, as an example, the fourth order equations of condition are developed. This program proceeds exactly as in the third order example of the previous section. As a result, the Taylor series expansions are in two variables and the requirement that $\alpha = \Sigma\beta$ was not included. The first revision of this program, Revision 1 shown in Appendix I, incorporates an autonomous differential equation to avoid the Taylor series expansions in two variables, i.e., the right-hand side of the differential equations contain only the dependent variable. The fact that $\alpha = \Sigma\beta$ was also incorporated into this program. Table 2 in the next section gives the results of this change.

With the increased efficiency gained by the first revision for the fourth order Runge-Kutta equations of condition, efforts were then directed toward generating the fifth order equations of condition. The fifth order method requires six function evaluations and all expansions were carried out to include fifth order terms, thus increasing the complexity of the problem considerably over that of the fourth order. With the increased complexity of computation, problems of inefficiency in the program became apparent. Due to the size of the expressions and to the large number of intermediate calculations, storage problems were encountered. When it became evident that this would be a serious problem, the program was divided into two parts: (1) $y(x_o + h)$ and $Y(x)$ were calculated as described earlier, and (2) like powers of h in the two resulting expressions were compared to yield the equations of condition.

The major area of inefficiency in the program was found to be the calculation of unnecessary terms in many instances which, after multiplications or substitutions, led to expressions containing higher order terms in h than were needed. The program at this stage was set up to develop all needed expressions to fifth order, then multiplications and substitutions were carried out and the resulting expressions truncated to fifth order. A considerable effort was made to minimize the calculation of these extra terms so that only those needed to make the final result fifth order were retained. The details of this work will not be discussed, but a copy of this program, Revision 2, is shown in Appendix I. Attempts at further revisions of this basic program did not lead to substantial improvement. The results of generating the equations of condition for the fifth order are presented in Table 3 of the next section.

From the results shown in Table 3 it is obvious that this method would not produce the fifth order equations of condition. As a final attempt to determine the fifth order results with this method, efforts were made to use magnetic tapes to help decrease storage and to store expressions for future use. However, the SYMBAL compiler has very limited file manipulation capabilities. It was possible to store results on tape, but there are no provisions for reading information into a SYMBAL program from tape so that the usefulness of this effort was considerably reduced.

Because this method was using the maximum allowable storage and because relatively long computing times were encountered at the fifth order level, it was evident that the series expansion approach was not practical, especially since the equations of condition were sought for high order formulas (eighth, ninth, tenth, etc.). Although this method failed to achieve the desired goal, it did provide much insight into the capabilities and limitations of the SYMBAL language.

2.3 Summary of Results

The results of the fourth order programs are given in Table 2. These results are for a complete set of equations with four function evaluations. Table 3 gives the results for the fifth order Runge-Kutta method with six function evaluations. It should be remembered that the fifth order program was divided into two sections, and Table 3 gives only the results for the first half of the program which was to calculate $y(x_o + h)$ and $Y(x)$. Although these expressions were finally obtained and stored on tape, the actual set of equations of condition was not obtained. The program, Revision 1, used in the fifth order study was the same as that used in the fourth order study with only the order and number of function evaluations changed.

Table 2.

Fourth Order Results

Program	CDC 6600		No. of Eqs.
	Storage Reqd.	Run Time	
Original	77000_8	30.4 sec.	19
Revision 1	77000_8	10.8 sec.	7

Table 3.

Fifth Order Results

Program	CDC 6600		$y(x_o + h)$, $Y(x)$
	Storage Reqd.	Run Time	
Revision 1	220000_8 (max.)	500^+ sec.	not obtained
Revision 2	220000_8 (max.)	97 sec.	obtained

CHAPTER III

GENERATION OF THE EQUATIONS OF CONDITION FOR RUNGE-KUTTA COEFFICIENTS WITH THE METHOD OF E. BAYLIS SHANKS

After the failure of the Taylor series approach discussed in Chapter II, an alternative method was sought. Since most of the problems in the Taylor series approach stemmed from the large expansions of $f^{[i]}$ and f_i, it was natural to turn to the work of E. Baylis Shanks[19]. Shanks has developed a method which generates the desired equations of condition without carrying out the classical expansions and comparing coefficients of similar powers of h, as was done in Chapter II.

3.1 Description of the Method

Again the system under consideration is

$$y' = f(x, y) \ , \quad y(x_o) = y_o \quad ,$$

where the solution is given by

$$Y(x) = y_o + h \sum_{i=0}^{n} c_i f_i \qquad ,$$

with

$$f_i = f(x_o + \alpha_i h \ , \quad y_o + h \sum_{j=0}^{i-1} \beta_{ij} f_j).$$

The parameters to be determined are again α, β, and c.

It can be shown[19] that a necessary and sufficient condition for the Taylor series for $y(x_o + h)$ and the expansion of the solution Y(x) to agree through terms in h^m is

$$(f^{[k-1]})_o = k \sum_{i=0}^{n} c_i (f_i{}^{[k-1]})_o, \ k = 1, 2, \ldots, m-1, \tag{4}$$

where $f^{[k]}$ was defined in Chapter II, and $f_i{}^{[k]}$ denotes the kth derivative of f_i as expanded about $(x_o + \alpha_i h, \ y_o + \sum_{j=0}^{i-1} \beta_{ij} f_j)$. Shanks states and proves several theorems in order to arrive at expressions for the derivatives $(f^{[k-1]})_o$ and $(f_i^{[k-1]})_o$ in Equation (4). He shows that the left-hand side of Equation (4) may be represented by

$$f^{[k]} = \Sigma_t \pi_t^{[k]} \tag{5}$$

where the factor $\pi_t^{[k]}$ is a product of integers and partial derivatives.

For an expression for $f_i^{[k]}$ Shanks first defines a quantity $Q_{i_1}^{[1]} \equiv \alpha_i$, and he then develops inductively that the general term, $Q_{i_t}^{[k]}$ is

$$\alpha_i^{i_1}\Big(\prod_{r=1}^{i_2}2\sum_{j=1}^{i-1}\beta_{ij}Q_{j_{t_r}}^{[1]}\Big)\ldots\Big(\prod_{r=1}^{i_k}k\sum_{j=1}^{i-1}\beta_{ij}Q_{j_{t_r}}^{[k-1]}\Big), \tag{6}$$

where the induction is on k, $k = i_1 + 2i_2 + \ldots + ki_k$, and where the subscripts t, t_r are used to number distinct terms defined for each fixed k. As an example consider $k = 2$ where either:

$$i_1 = 2,\ i_2 = 0,\ t = 1,$$
$$Q_{i_1}^{[2]} = \alpha_i^2;$$

or

$$i_1 = 0,\ i_2 = 1,\ t = 2,$$
$$Q_{i_2}^{[2]} = 2\sum_{j=1}^{i-1}\beta_{ij}\alpha_j\ .$$

It is then stated and proven[19] that $f_j^{[k]}$ can be given as

$$f_j^{[k]} = \Sigma_t Q_{j_t}^{[k]}\pi_t^{[k]}\ . \tag{7}$$

From Equations (4), (5), and (7) the necessary and sufficient condition can now be written:

$$\Sigma_t\pi_t^{[k]} = (k+1)\sum_{i=0}^{n} c_i\Sigma_t Q_{i_t}^{[k]}\pi_t^{[k]}\ ,$$

or, by rearranging this expression,

$$\Sigma_t[1 - (k + 1)\sum_{i=1}^{n} c_i Q_{i_t}^{[k]}]\pi_t^{[k]} = 0 \quad . \tag{8}$$

From Equation (8) the sufficient condition then becomes:

$$1 = (k + 1)\sum_{i=1}^{n} c_i Q_{i_t}^{[k]}, \; k = 1, 2, \ldots, m\text{-}1. \tag{9}$$

The reader is referred to the original paper by Shanks[19] for a complete treatment of the theorems and proofs leading to Equation (9). Equation (9) gives the sufficient conditions to be satisfied for a given order and these equations can be completely determined when the $Q_{i_t}^{[k]}$ expressions are known.

For comparison with the Taylor series approach, the third order equations will be developed using Shanks' method. From Equation (4), with k = 1, it is seen that:

$$f_o = \sum_{i=0}^{n} c_i f_{i_o} \quad ,$$

but since $f_{i_o} = f_o$

$$1 = \sum_{i=0}^{n} c_i \quad .$$

This equation always occurs and is thus assumed for simplicity from this point on since it is not produced by Equation (9). From Equations (6) and (9), for k = 1:

$$Q_{i_1}^{[1]} = \alpha_i \quad ,$$

$$1 = 2\sum_{i=1}^{2} c_i \alpha_i \quad ;$$

and for k = 2,

$$Q_{i_1}^{[2]} = \alpha_i^2 \quad , \qquad Q_{i_2}^{[2]} = 2\sum_{j=1}^{1} \beta_{i1}\alpha_1 \quad ,$$

$$1 = 3\sum_{i=1}^{2} c_i \alpha_1^2 \quad , \qquad 1 = 3\sum_{i=2}^{2} c_i 2\sum_{j=1}^{1} \beta_{i1}\alpha_1 \quad .$$

The equations of condition are then:

$$1 = c_o + c_1 + c_2 \quad ,$$

$$\frac{1}{2} = c_1\alpha_1 + c_2\alpha_2 \quad ,$$

$$\frac{1}{3} = c_1\alpha_1^2 + c_2\alpha_2^2 \quad ,$$

$$\frac{1}{6} = c_2\beta_{21}\alpha_1 \quad ,$$

which are identical to the equations obtained in the example in Chapter II. The series expansions have been eliminated and the computational complexity greatly reduced.

3.2 Computational Procedure and Limitations

A SYMBAL program was written to produce the Q expressions, from which the equations of condition follow immediately. This program is shown in Appendix I as Shanks' Method and includes the output for the fifth order equations of condition. It was found that Shanks' method was particularly well suited for computer implementation.

Equations of condition were generated for the fourth, fifth, sixth, and seventh order Runge-Kutta methods without difficulty. The number of function evaluations used in each order is given in Table 1 in section 2.1. The results of this program are given in Table 4 in section 3.4. From Table 4 the time required for successive orders is seen to grow rapidly. The time that would be required for an eighth order run was estimated at about eight minutes. However, it was not possible to run this program because of what appeared to be a limit on the size of an array or the number of variables used. The program could not be compiled, and was terminated during the setting up of the Q array which is a triangular array with three indices of the form Q[8, 13, 297]. An "inventory overflow" diagnostic was given and all efforts to compile the eighth order program failed.

3.3 Generation of Reduced Systems of Equations of Condition for High Orders

Up to this point the goal had been the generation of the complete set of equations of condition for each order of Runge-Kutta method. However, to solve the resulting equations for the α, β, and c coefficients, the number of equations must be reduced for the higher order methods since the system is overdetermined, i.e., more equations than unknowns. The usual procedure in solving the equations of condition is to generate the complete set of equations and then make certain assumptions to reduce the number of equations. The assumptions that are usually made arise naturally from a quadrature approach[16] and they make several of the original equations identical, thus reducing the number of equations in the system.

The assumptions used in this study were:

$$\Sigma\beta\alpha = \frac{1}{2}\alpha^2 ,$$

$$\Sigma\beta\alpha^2 = \frac{1}{3}\alpha^3 ,$$

$$\Sigma\beta\alpha^3 = \frac{1}{4}\alpha^4 . \tag{10}$$

These assumptions were incorporated into the calculation of the Q expressions which then led to the generation of a reduced system of equations directly. This program is shown in Appendix I as Shanks' Revision 1.

The assumptions that reduce the number of equations of condition, and the number of these assumptions to be made to minimize the number of function evaluations required are discussed by Curtis[3], who concludes that four assumptions of the type given by Equation (10) will produce an eighth order method with eleven function evaluations. Since the present study is concerned with generating the equations of condition, it was decided to use only the three assumptions given in Equation (10) solely on the basis of ease of computation of the eighth and ninth order equations of condition. The eighth and ninth order equations of condition were then generated in reduced form without difficulty. The results are given in Table 5 in section 3.4.

With this technique it is now feasible to go to the tenth and higher order methods by making additional assumptions, but this was not done since it was desired to use the equations generated up to this point to make comparisons of local truncation errors. This comparison is discussed in Chapter IV. It is stressed, however, that the equations of condition can be computer generated for higher order Runge-Kutta methods.

3.4 Summary of Results

The results for the fourth, fifth, sixth, seventh, and eighth order methods using the Shanks method are given in Table 4. These are complete sets of equations, but the number of equations given for each order in the table is the number of new equations due only to that order. Table 5 gives the results for the reduced eighth and ninth order sets of equations of condition. The number of equations given in Tables 4 and 5 do not always agree with the number of equations expected for the higher orders since the SYMBAL program generates some identical equations for the higher orders.

Table 4.

Results from Shanks' Method

Order	CDC 6600		No. of New Eqs.
	Storage Reqd.	Run Time	
4	77000_8	1.4 sec.	4
5	77000_8	3.8 sec.	9
6	77000_8	12.0 sec.	20
7	150000_8	75.5 sec.	49
8	220000_8	Not Compiled	117

Table 5.

Results from Revised Shanks' Method

Order	CDC 6600		No. of New Eqs.	
	Storage Reqd.	Run Time	Orig.	Reduced
8	77000_8	12.0 sec.	117	8
9	120000_8	70.0 sec.	297	16

CHAPTER IV

CALCULATION AND COMPARISON OF LOCAL TRUNCATION ERROR COEFFICIENTS

4.1 Description of the Computational Procedure

As a means of comparing sets of coefficients for Runge-Kutta methods of a given order, the coefficients of the local truncation error terms can be evaluated. From the development of the equations of condition, it is observed that these equations arise as multiplicative factors of the partial derivatives that appear in the expansions of the solutions $y(x_o + h)$ and $Y(x)$, as given in Chapter II. For these two solutions to agree through some order m, the equations of condition arising from comparing terms in h up through order m must be satisfied exactly. The leading term of the truncation error consists of the partial derivatives multiplied by numerical factors. These factors are simply the equations of condition that arise from the h^{m+1} terms in the expansions, and are explicit functions of the coefficients of the mth order Runge-Kutta method. More explicitly, these truncation error terms are of the form

$$TE(i)[\ldots]_i h^{m+1} ,$$

where TE(i) represents the truncation error coefficient and $[\ldots]_i$ represents the partial derivatives which are multiplied by that particular TE(i). The index i numbers the distinct terms in the h^{m+1} portion of the expansion. By evaluating these multiplication factors, or truncation error coefficients, and by comparing the results with those obtained from using other sets of Runge-Kutta coefficients for the same order method, an "optimum" set of coefficients for use with a given order Runge-Kutta method can be determined by determining the set that has the lowest values for these truncation error coefficients.

The equations of condition were generated in complete form for orders four through seven. These equations were used to produce the truncation error coefficients for the third through the sixth order methods. For the eighth and ninth order methods, it was necessary to generate reduced sets of equations of condition, and these were then used to obtain the truncation error coefficients for the seventh and eighth order methods.

The truncation error coefficients for several sets of Runge-Kutta coefficients are given for methods of order three through methods of order eight in Table 7 through Table 12 in the next section.

4.2 Comparison of Truncation Error Coefficients

Several sets of Runge-Kutta coefficients for methods of order three through eight were used in this comparison. These coefficients are given in Appendix II for reference. Several points concerning notation should be emphasized for these sets of coefficients. The Fehlberg[5, 6] methods are given as "Fehlberg m(m + 1)" where m is the order of the method, with a solution of order m + 1 also being calculated to use with an automatic step-size control. The difference in the two solutions is used as an estimate of the truncation error made during that step. The new step-size is then based on this error and the desired accuracy. When this step-size control is used, the Fehlberg methods require extra function evaluations to produce the solution to both orders. The advantage of the automatic step-size, however, makes these methods desirable, especially if the system of differential equations is such that a rapidly varying step-size is required. Since these methods can be used to give a solution of order m or of order m + 1, they are shown in the comparisons for both orders. Table 6 shows the Fehlberg methods with the number of function evaluations required for each order solution with and without the automatic step-size control.

The notation for the Butcher[2] and Shanks[19] methods are of the form "Shanks (m - n)" where m denotes the order of the method, and n denotes the number of function evaluations required per step. It should be noted also that some of these methods by Butcher and Shanks are only approximately of order m. They were developed such that the number of function evaluations used is less than that normally needed for a true mth order method. For this reason the Shanks (6-6) is compared as both a fifth and a sixth order method, and the Shanks (7-7) is compared as both a sixth and a seventh order method. Methods which are approximations of this type are denoted with an asterisk in the tables.

A large number of fifth order methods fall within a family of methods known as the Newton-Cotes Family. Only three of these forms are given although several others exist. The "UT" coefficients[1] are sets of coefficients developed at The University of Texas at Austin along the Fehlberg 4(5) format so they appear in both the fourth and fifth order comparisons.

The truncation error coefficients are denoted by TE(i) and are given in Table 7 through Table 12 for the third through eighth order methods. The first row in each table gives the number of function evaluations required (F.E.R.) per step for each set of coefficients.

Table 6.

Function Evaluations Required
for the Fehlberg Methods

Method	Order of Solution	No. of Function Evaluations Reqd.	
		With Step-Size Control	Without Step-Size Control
Fehlberg 4(5)	4	6	5
Fehlberg 4(5)	5	6	6
Fehlberg 5(6)	5	8	6
Fehlberg 5(6)	6	8	8
Fehlberg 6(7)	6	10	8
Fehlberg 6(7)	7	10	10
Fehlberg 7(8)	7	13	11
Fehlberg 7(8)	8	13	13
Fehlberg 8(9)	8	17	15

TABLE 7

COMPARISON OF THIRD ORDER METHODS

	FEHLBERG 3(4)	CLASSICAL	RALSTON OPTIMUM	HEUN	NYSTROM
F.E.R.	4	3	3	3	3
TE(1)	.002339	0.000000	.020833	.027777	.027777
TE(2)	.002339	.083333	0.000000	.027777	.027777
TE(3)	0.000000	0.000000	0.000000	.083333	.083333
TE(4)	.026315	.250000	.250000	.250000	.250000

TABLE 8

COMPARISON OF FOURTH ORDER METHODS

	FEHLBERG 3(4)	FEHLBERG 4(5)-1	FEHLBERG 4(5)-2	CLASSICAL
F.E.R.	5	5	5	4
TE(1)	.000249	.000057	.000020	.000347
TE(2)	.001498	.000347	.000120	.002083
TE(3)	.001388	.000231	.000080	.002083
TE(4)	.005774	.002083	.001282	.008333
TE(5)	.000999	.000231	.000080	.001388
TE(6)	.002997	.000694	.000240	.004166
TE(7)	.001388	.000231	.000080	.002083
TE(8)	.005774	.002083	.001282	.008333
TE(9)	.000749	.000173	.000060	.006250

TABLE 8 CONT.

COMPARISON OF FOURTH ORDER METHODS

F.E.R.	KUNTZMANN OPT. 4	KUTTA 4	SARAFYAN 4	UT 1 5
TE(1)	.000277	.000154	.000347	0.000000
TE(2)	0.000000	.001388	.002083	.000001
TE(3)	0.000000	.001398	.002083	0.000000
TE(4)	.008333	.008333	.008333	.000006
TE(5)	.001111	.000617	.001388	0.000000
TE(6)	0.000000	.002777	.004166	.000001
TE(7)	0.000000	.001388	.002083	0.000000
TE(8)	.008333	.008333	.008333	.000006
TE(9)	.003409	.002777	.001041	0.000000

TABLE 8 CONT.

COMPARISON OF FOURTH ORDER METHODS

F.E.R.	UT 2 5	UT 3 5	SHANKS(4-4) 4
TE(1)	.000183	.000200	.000006
TE(2)	.001103	.001204	0.000000
TE(3)	.000735	.000802	.008125
TE(4)	.001768	.001698	.008333
TE(5)	.000735	.000803	.000027
TE(6)	.002206	.002409	0.000000
TE(7)	.000735	.000803	.008125
TE(8)	.001767	.001698	.008333
TE(9)	.000551	.000602	.000020

TABLE 9

COMPARISON OF FIFTH ORDER METHODS

F.E.R.	FEHLBERG 4(5)-1 6	FEHLBERG 4(5)-2 6	FEHLBERG5(6) 6	SARAFYAN 6	SHANKS(5-5)* 5
TE(1)	.001157	.002483	0.000000	.001111	.000416
TE(2)	.001157	.002483	0.000000	.001111	.000416
TE(3)	.001157	.002483	0.000000	.005833	.004156
TE(4)	.006250	.014743	0.000000	.015000	.004156
TE(5)	.001851	.005448	.000740	.010000	.013333
TE(6)	.001851	.005448	.000740	.010000	.013333
TE(7)	.001851	.005448	0.000000	.023333	.033303
TE(8)	.016666	.037179	.011111	.076666	.033333
TE(9)	.001157	.002483	0.000000	.001111	.000416
TE(10)	.006944	.011217	0.000000	.027777	.002083
TE(11)	.006944	.011217	0.000000	.027777	.002083
TE(12)	.006944	.011217	0.000000	.020833	.020805
TE(13)	.020833	.054487	0.000000	.125000	.020835
TE(14)	.013888	.022435	.003703	.083333	.066666
TE(15)	.013888	.022435	.003703	.083333	.066666
TE(16)	.013888	.022435	0.000000	.083333	.166611
TE(17)	.041666	.108974	.055555	.416666	.166666
TE(18)	.006944	.011217	0.000000	.027777	.002083
TE(19)	.001157	.002483	0.000000	.005833	.004156
TE(20)	.006250	.014743	0.000000	.015000	.004166

TABLE 9 CONT.

COMPARISON OF FIFTH ORDER METHODS

	BUTCHER 1	BUTCHER 2	NEWTON -COTES FAMILY		
			LAWSON	LUTHER	BUTCHER
F.E.R.	6	6	6	6	6
TE(1)	.001111	.001111	0.000000	0.000000	0.000000
TE(2)	.001111	.001111	0.000000	0.000000	0.000000
TE(3)	.003333	.003333	.003645	.008333	.000260
TE(4)	.003333	.003333	.002604	.016666	.001041
TE(5)	.006666	.006666	.004166	.004166	.004166
TE(6)	.006666	.006666	.004166	.004166	.004166
TE(7)	.013333	.053333	.002083	.008333	.002083
TE(8)	.033333	.033333	.014583	.004166	.004166
TE(9)	.001111	.001111	0.000000	0.000000	0.000000
TE(10)	.005555	.005555	0.000000	.010416	0.000000
TE(11)	.005555	.005555	0.000000	.010416	0.000000
TE(12)	.016666	.016666	.018229	.020833	.001302
TE(13)	.016666	.016666	.013020	.010416	.005208
TE(14)	.033333	.033333	.020833	.041666	.020833
TE(15)	.033333	.033333	.020833	.041666	.020833
TE(16)	.066666	.266666	.010416	.333333	.010416
TE(17)	.166666	.166666	.010416	.541666	.020833
TE(18)	.005555	.005555	0.000000	.010416	0.000000
TE(19)	.003333	.003333	.003645	.008333	.000260
TE(20)	.003333	.003333	.002604	.016666	.001041

TABLE 9 CONT.

COMPARISON OF FIFTH ORDER METHODS

F.E.R.	NYSTROM 6	SHANKS(6-6) 6	UT 1 6	UT 2 6	UT 3 6
TE(1)	.002222	.000222	.003307	.000624	.001244
TE(2)	.002222	.000222	.003309	.000624	.001244
TE(3)	.001111	.000083	.003308	.000624	.001244
TE(4)	.006666	0.000000	.016608	.007175	.002559
TE(5)	.006666	0.000000	.006644	.005665	.001699
TE(6)	.006666	0.000000	.006644	.005664	.001699
TE(7)	0.000000	.000222	.006644	.005664	.001699
TE(8)	.033333	0.000000	.033573	.024958	.006788
TE(9)	.002222	.000222	.003310	.000624	.001245
TE(10)	.016666	.001111	.016412	.007907	.008195
TE(11)	.016666	.001111	.016412	.007907	.008195
TE(12)	0.000000	.000416	.016412	.007907	.008195
TE(13)	.083333	0.000000	.082078	.009356	.008622
TE(14)	.033333	0.000000	.032839	.015815	.016393
TE(15)	.033333	0.000000	.032840	.015815	.016393
TE(16)	0.000000	.001111	.032839	.015815	.016393
TE(17)	.166666	0.000000	.164173	.018713	.036957
TE(18)	.166666	.001111	.016411	.007906	.008195
TE(19)	.001111	.000083	.003309	.000624	.006226
TE(20)	.006666	0.000000	.016609	.007175	.002559

TABLE 10

COMPARISON OF SIXTH ORDER METHODS

F.E.R.	FEHLBERG 5(6) 8	FEHLBERG 6(7) 8	BUTCHER(6-7) 7
TE(1)	.000402	0.000000	0.000000
TE(2)	.000402	0.000000	.070820
TE(3)	.000472	0.000000	.070820
TE(4)	.001587	0.000000	.150000
TE(5)	.000105	0.000000	.062400
TE(6)	.000105	0.000000	.265189
TE(7)	.000634	0.000000	.265189
TE(8)	.011746	0.000000	3.324834
TE(9)	.000402	0.000000	1.133327
TE(10)	.001005	.000450	.054035
TE(11)	.001005	.000450	.444664
TE(12)	.001190	.000450	.444664
TE(13)	.003968	.000450	3.616154
TE(14)	.001587	.011033	.393157
TE(15)	.001587	.011033	2.936067
TE(16)	.001587	.009095	2.936067
TE(17)	.001587	.003279	3.111607
TE(18)	.001005	.000450	6.304039
TE(19)	.000472	0.000000	1.133327
TE(20)	.001587	0.000000	.841666
TE(21)	.002412	0.000000	.041972
TE(22)	.002412	0.000000	.466975
TE(23)	.002857	0.000000	.466975
TE(24)	.009523	0.000000	3.637962
TE(25)	.000634	0.000000	.413732
TE(26)	.000634	0.000000	2.956393
TE(27)	.003809	0.000000	2.956393
TE(28)	.070476	0.000000	3.159325
TE(29)	.002412	0.000000	6.841975
TE(30)	.006031	.002705	.327738
TE(31)	.006031	.002705	2.237874
TE(32)	.007142	.002705	2.237874
TE(33)	.023809	.002705	2.497519
TE(34)	.009523	.066203	1.931866
TE(35)	.009523	.066203	2.041005
TE(36)	.009523	.054571	2.041005
TE(37)	.009523	.019677	2.622023
TE(38)	.006031	.002705	30.831620
TE(39)	.002857	0.000000	6.841975
TE(40)	.009523	0.000000	9.587962
TE(41)	.000105	0.000000	1.027839
TE(42)	.000105	0.000000	1.209634
TE(43)	.000634	0.000000	1.209634
TE(44)	.011746	0.000000	4.835945
TE(45)	.000402	0.000000	17.070820
TE(46)	.000809	0.000000	1.133327
TE(47)	.005809	0.000000	.841666
TE(48)	.005809	0.000000	.841666
TE(49)	.069142	0.000000	235.207200

TABLE 1C CONT.

COMPARISON OF SIXTH ORDER METHODS

F.E.R.	BUTCHER(6-8) 8	SHANKS(6-6)* 6	SHANKS(7-7) 7
TE(1)	.000661	.000639	.000033
TE(2)	.000661	.000639	.000033
TE(3)	.001587	.000471	.000459
TE(4)	.004365	.000698	.000208
TE(5)	.000264	.003746	.002155
TE(6)	.000264	.003746	.002155
TE(7)	.001587	.006783	.003883
TE(8)	.007142	.007301	.003968
TE(9)	.000661	.000637	.000033
TE(10)	.001653	.000636	.000497
TE(11)	.001653	.000649	.000496
TE(12)	.003968	.000584	.000022
TE(13)	.010912	.001209	.000296
TE(14)	.005291	.012624	.000661
TE(15)	.005291	.012697	.000661
TE(16)	.023809	.022432	.001818
TE(17)	.079365	.023823	.001557
TE(18)	.001653	.000649	.000496
TE(19)	.001587	.000469	.000459
TE(20)	.004365	.000698	.000208
TE(21)	.003968	.002501	.000196
TE(22)	.003968	.002501	.000198
TE(23)	.009523	.003330	.001092
TE(24)	.026190	.004190	.000203
TE(25)	.001587	.022478	.012935
TE(26)	.001587	.022478	.012930
TE(27)	.009523	.042031	.023299
TE(28)	.042857	.043809	.023809
TE(29)	.003968	.002501	.000198
TE(30)	.009920	.002857	.002975
TE(31)	.009920	.002857	.002976
TE(32)	.023809	.005892	.008184
TE(33)	.065476	.007142	.007008
TE(34)	.031746	.076197	.003968
TE(35)	.031746	.076190	.003968
TE(36)	.142857	.141190	.010912
TE(37)	.476190	.142857	.009344
TE(38)	.009920	.002857	.002976
TE(39)	.009523	.003330	.001092
TE(40)	.026190	.004190	.000203
TE(41)	.000264	.003746	.002155
TE(42)	.000264	.003746	.002155
TE(43)	.001587	.006783	.003883
TE(44)	.007142	.007301	.003968
TE(45)	.000661	.000639	.000033
TE(46)	.000432	.010800	.000862
TE(47)	.006493	.013190	.000602
TE(48)	.006493	.013190	.000602
TE(49)	.039069	.016190	.000281

TABLE 11

COMPARISON OF SEVENTH ORDER METHODS

	FEHLBERG 6(7)	FEHLBERG 7(8)	SHANKS(7-7)*	SHANKS(7-9)
F.E.R.	10	11	7	9
TE(1)	.000025	0.000000	.000115	0.000000
TE(2)	.000450	0.000000	.002515	.000055
TE(3)	.000077	.000066	.001391	.000183
TE(4)	0.000000	.000551	.004629	.000110
TE(5)	.000180	0.000000	.000578	0.000000
TE(6)	.003156	0.000000	.014138	.000385
TE(7)	.000541	.000462	.011111	.001286
TE(8)	0.000000	.003858	.011574	.000771

TABLE 12

COMPARISON OF EIGHTH ORDER METHODS

F.E.R.	FEHLBERG 7(8) 13	FEHLBERG 8(9) 15	SHANKS(8-10)* 10	SHANKS(8-12) 12	CURTIS 11
TE(1)	.000025	0.000000	.000025	.000025	.000027
TE(2)	.000047	.000001	.001492	.000029	.000344
TE(3)	.000088	.000001	.001620	.000029	.000057
TE(4)	.000211	.000001	.000492	.000289	.001762
TE(5)	.000090	0.000000	.000154	.000090	.000100
TE(6)	.000166	.000001	.003182	.000102	.000079
TE(7)	.000077	.000002	.000308	.000308	.000196
TE(8)	.001189	.000539	.005208	.002642	.000806
TE(9)	.000205	.000001	.000205	.000205	.000228
TE(10)	.000379	.000002	.005327	.000235	.002748
TE(11)	.000705	.000002	.012962	.000264	.000451
TE(12)	.001690	.000012	.003943	.002314	.014116
TE(13)	.000720	.000007	.001234	.000720	.000790
TE(14)	.001328	.000006	.017319	.000823	.000648
TE(15)	.000617	.000018	.000247	.002469	.001594
TE(16)	.009516	.004406	.041664	.021141	.006500

CHAPTER V

CONCLUSIONS AND RECOMMENDATIONS

5.1 Conclusions from the Comparison of the Truncation Error Coefficient

In Chapter IV it was stated that the "optimum" set of coefficients for a Runge-Kutta method of a particular order could be established by determining the set of Runge-Kutta coefficients with the lowest values for the truncation error coefficients. Although this is the dominant factor in comparing various sets of Runge-Kutta coefficients, the reader should also bear in mind the number of function evaluations required by each set of coefficients. It was shown in Table 6 that the Fehlberg coefficients require more function evaluations for the mth order method when the automatic step-size control is used. The disadvantage of more function evaluations is usually offset by an efficient choice of step-size, which in turn reduces the number of steps required. Several of the Shanks and Butcher methods minimize the number of function evaluations but do not incorporate a step-size control. For a rapidly varying function, a poor choice of step-size with these methods could result in many more steps being taken than necessary, and the resulting total number of function evaluations being correspondingly large.

Since the system of differential equations to be solved determines the complexity of the function evaluations and the step-size that can be taken, the matter of choosing the best set of Runge-Kutta coefficients becomes problem dependent. For this reason no absolute optimum set of coefficients can be given. If, however, certain classes of differential equations are considered, some conclusions can be drawn from the comparison made in Chapter IV.

Three types of systems of differential equations will be considered: (1) systems with rapidly varying functions; (2) systems with slowly varying functions; and (3) systems requiring complicated function evaluations. For systems of differential equations with rapidly varying functions, it is desirable and necessary to incorporate some type of variable step control. Efficient determination and correction of the step-size is essential if the method is to progress efficiently to a solution. The sets of Runge-Kutta coefficients recommended for each order method for a system of differential equations with rapidly varying functions are given in Table 13. If, on the other hand, the system of differential equations has only slowly varying functions, an automatic step-size control is not a necessity. The recommendations for problems of this type are given in Table 14. Finally, if the system of differential equations is such that the function evaluations become quite complicated, a method which minimizes the number of function evaluations while still producing small truncation errors is required. Table 15 gives the sets of coefficients recommended for this type of problem.

Table 13.

Recommendations for Rapidly Varying Functions

Order	Recommended Method	No. of Function Evaluations Reqd. with Step-Size Control
3	Fehlberg 3(4)	5
4	UT 1	6
5	Fehlberg 5(6)	8
6	Fehlberg 6(7)	10
7	Fehlberg 6(7)	10
8	Fehlberg 8(9)	17

Table 14.

Recommendations for Slowly Varying Functions

Order	Recommended Method	Number of Function Evaluations Required
3	Fehlberg 3(4)	4
4	UT 1	5
5	Fehlberg 5(6)	6
6	Shanks (7-7)	7
7	Fehlberg 6(7)	10
8	Fehlberg 7(8)	13

Table 15.

Recommendations for Complicated Function Evaluations

Order	Recommended Method	Number of Function Evaluations Required
3	Ralston Optimum	3
4	Kuntzmann Optimum	4
5	Shanks (5-5)	5
6	Shanks (6-6)	6
7	Shanks (7-9)	9
8	Curtis	11

5.2 Topics for Future Study

With the availability of the equations of condition, it is possible to attempt to produce explicit Runge-Kutta methods of the ninth, tenth, and even possibly higher orders. However, the solution of the equations of condition for these high order methods will be very difficult. It is hoped that a computer process can be developed to solve these extremely large systems of nonlinear algebraic equations. Aside from producing new methods, the equations of condition can be used to improve the lower order methods. The UT 1 coefficients were successfully developed previously to optimize the fourth order Runge-Kutta method. If a computer process is developed to solve these equations of condition, then some form of optimizing process can be used to produce similar sets of coefficients for the higher order methods.

APPENDICES

APPENDIX I

Original SYMBAL Program

```
                                        SYMBAL VERSION 1.1E      (3/14/69)
                                        START PROGRAM            0. 0.  0
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
.     EQUATIONS OF CONDITION FOR RUNGE KUTTA METHOD                          .
. ≡BEGIN≡                                                                    .
.    ≡NEW≡ DX,DY,YN1,YN2,TS,ZZ,COEF;                                         .
.         DUMP := 5;                                                         .
.    M := 4;                                                                 .
.    P := 3;                                                                 .
.         MODE[4] := M;                                                      .
.    F := ←0: ≡FOR≡ I := 0:M-1 ≡DO≡ ←0:M-I-1:→→;                             .
.    TS := ≡FOR≡ I := 0:M-1 ≡SUM≡ ≡FOR≡ J := 0:M-1-I ≡SUM≡                   .
.         H*DX↑I*DY↑J*F[I,J]/FACT(I)/FACT(J);                                .
.    A := ←1:P:→;                                                            .
.    K := ←0:H*F[0,0],P:→;                                                   .
.    B := ← ≡FOR≡ I := 1:P ≡DO≡ ←0:I-1:→→;                                   .
.    ≡FOR≡ I := 1:P ≡DO≡ ≡BEGIN≡                                             .
.       S := ≡FOR≡ J := 0:I-1 ≡SUM≡ B[I,J]*K[J];                             .
.       K[I] := ≡S≡ DX:A[I]*H,DY:S $ TS;                                     .
.    ≡END≡;                                                                  .
.    C := ←0:P:→;                                                            .
.      YN1 := ≡FOR≡ I := 0:P ≡SUM≡ C[I]*K[I];                                .
.    Z := ←F[0,0],M:→;                                                       .
.    YN2 := H*Z[1]+( ≡FOR≡ I := 2:M ≡SUM≡ H↑I/FACT(I)*(Z[I] :=               .
.      ≡FOR≡ J := 0:I-2 ≡SUM≡ ≡FOR≡ N := 0:I-J-2 ≡SUM≡                       .
.       (F[J+1,N]+F[0,0]*F[J,N+1])* ≡D≡ F[J,N] $ Z[I-1]));                   .
.    ZZ := SPLIT POWERS (YN2-YN1,H);                                         .
.    ≡FOR≡ I := 1:M ≡DO≡ ≡FOR≡ J := 0:I-1 ≡DO≡ ≡FOR≡ N := 0:I-J-1 ≡DO≡       .
.      ZZ[I] := ≡S≡ F[J,N]:Q↑F[J,N] $ ZZ[I];                                 .
.    ≡FOR≡ I := 1:M ≡DO≡ ≡BEGIN≡                                             .
.      T := SPLIT POWERS (ZZ[I],G);                                          .
.      ≡FOR≡ J := 1:UPBOUND(T) ≡DO≡ ≡BEGIN≡                                  .
.        COEF := T[-J];                                                      .
.        T[J] := T[J];                                                       .
.      ≡END≡;                                                                .
.    ≡END≡;                                                                  .
.  ≡END≡;                                                                    .
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
                                        TOTAL SPACE AVAILABLE     012277B
                                        EXECUTE PROGRAM           0. 0.642
```

Revision 1

```
                                        SYMBAL VERSION 1.1E      (3/14/69)
                                        START PROGRAM            0. 0.  0
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
.  THE EQUATIONS OF CONDITION FOR A GENERAL RUNGE-KUTTA METHOD               .
.    FOR ANY ORDER RUNGE KUTTA FORMULA FOR AUTONOMOUS SYSTEM                 .
.≡BEGIN≡                                                                     .
.    ≡NEW≡ DX, DY, YN1, YN2, T , ZZ, COEF;                                   .
.    M := 4;                                                                 .
.    P := 3;                                                                 .
.                                              MODE[4] := M;                 .
. TAYLOR SERIES                                        ;                     .
.    F := ←0:M-1:→;                                                          .
.    TS := ≡FOR≡ I:= 0:M-1 ≡SUM≡ H*DY↑I*F[I]/FACT(I);                        .
. COMPUTATION OF THE TAYLOR SERIES FOR THE K:                                .
.    K := ←0:H*F[0],P:→;                                                     .
.    B := ←≡FOR≡ I:= 1:P ≡DO≡ ←0:I-1:→→;                                     .
.    ≡FOR≡ I:= 1:P ≡DO≡ ≡BEGIN≡                                              .
.        S := ≡FOR≡ J:= 0:I-1 ≡SUM≡ B[I,J]*K[J];                             .
.    K[I] := ≡S≡ DY:S $ TS ;                                                 .
.    ≡END≡;                                                                  .
. FINAL RUNGE KUTTA SUMMATION:                                               .
.    C := ←0:P:→;                                                            .
.    YN1 := ≡FOR≡ I := 0:P ≡SUM≡ C[I]*K[I];                                  .
.                                                                            .
.    Z := ←F[0],M:→;                                                         .
.    YN2 := H*Z[1] + (≡FOR≡ I := 2:M ≡SUM≡ H↑I/FACT(I)*(Z[I] :=              .
.          ≡FOR≡ J:= 0:I-2 ≡SUM≡ (F[0]*F[J+1])*≡D≡F[J] $ Z[I-1]));           .
. DIFFERENCE OF YN1 AND YN2 SET TO ZERO YIELDS EQUATIONS:                    .
.    ZZ := SPLIT POWERS(YN2-YN1,H);                                          .
.    ≡FOR≡ I:= 1:M ≡DO≡ ≡FOR≡ J := 0:I-1 ≡DO≡ ZZ[I] := ≡S≡ F[J]:Q↑F[J]       .
.          $ ZZ[I];                                                          .
. SEPARATION OF EQUATIONS ACCORDING TO PRODUCTS OF DIFFERENT F S:            .
.                                                    DUMP := 5;              .
.    ≡FOR≡ I := 1:M ≡DO≡ ≡BEGIN≡                                             .
.        T := SPLIT POWERS(ZZ[I],Q);                                         .
.        ≡FOR≡ J := 1:UPBOUND(T) ≡DO≡ ≡BEGIN≡                                .
.            COEF := T[-J];                                                  .
.            T[J] := T[J];                                                   .
.        ≡END≡;                                                              .
.    ≡END≡;                                                                  .
.≡END≡;                                                                      .
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
                                        TOTAL SPACE AVAILABLE     021277B
                                        EXECUTE PROGRAM           0. 0.067
```

Revision 2

```
                                        SYMBAL VERSION 1.1F      (3/14/69)
                                        START PROGRAM            0. 0.  0
 THE EQUATIONS OF CONDITION FOR A GENERAL RUNGE-KUTTA METHOD
≡BEGIN≡
    ≡NEW≡ DX, DY, YN1, YN2, TS, ZZ, COEF;

    M:=5;  P:=5;
      MODE[4] := M-1;
      MODE[5] := 0;      MODE[1] := M;
    F := *0:M-1:*;
      D := * 1:M-1 : *;
      G := * 1:M-1:*;
      TS := ≡FOR≡ I := 1:M-1 ≡SUM≡ H*D[I]*F[I]/FACT(I);
      TS := H*F[0]+TS;
 COMPUTATION OF THE TAYLOR SERIES FOR THE K:
    K := *0:H*F[0],P:*;
    B := *≡FOR≡ I:=  1:P ≡DO≡ *0:I-1:**;
    ≡FOR≡ I:= 1:P ≡DO≡ ≡BEGIN≡
          MODE[4] := M-2;
      G[I] := ≡FOR≡ J := 0:I-1 ≡SUM≡ B[I,J]*K[J];
      ≡FOR≡ N := 2:M-1 ≡DO≡ ≡BEGIN≡ MODE[4] := M-N-1;
      B[N] := G[1]*1 ;
          MODE[4] := M-2;
      G[N] := B[N]*N ;
      ≡END≡ ;
      MODE[4] := M-1;
   K[I] := ≡S≡ D[1]:G[1] , D[2]:G[2] , D[3]:G[3] ,
                           D[4]:G[4] $ TS ;
    ≡END≡;
 FINAL RUNGE KUTTA SUMMATION:
    C := *0:P:*;
    YN1 := ≡FOR≡ I := 0:P ≡SUM≡ C[I]*K[I];

    Z := *F[0],M:*;
    YN2 := H*Z[1] + (≡FOR≡ I := 2:M ≡SUM≡ H*I/FACT(I)*(Z[I] :=
        ≡FOR≡ J:= 0:I-2 ≡SUM≡ (F[0]*F[J+1])*≡D≡F[J] $ Z[I-1]));
≡END≡;
                                        TOTAL SPACE AVAILABLE     012277B
                                        EXECUTE PROGRAM          0. 0.616
```

Shanks' Method with
Fifth Order Output

```
                                        SYMBAL VERSION 1.1E      (3/14/69)
                                        START PROGRAM            0. 0.   0
    ≡BEGIN≡
     DUMP := 0 ;
     MODE[1] := 0 ;
     M:=5;
     N:=5;
     L:=⌐0:1,1,2,4,9,20,49,117,297,746,10:↓;
     A:=⌐1:N:↓;
     C:=⌐1:N:↓;
     B:=⌐≡FOR≡ I:= 1:N ≡DO≡ ⌐0:I-1:↓↓;
     Q:=⌐≡FOR≡ K := 1:M-1 ≡DO≡
        ⌐≡FOR≡ I := 1:N   ≡DO≡
        ⌐1:L[K]:↓↓↓;

     COMPUTE THE VALUES OF THE Q S   ;

     ≡FOR≡ I:=1:N ≡DO≡ Q[1,I,1] := A[I];
     ≡FOR≡ K:=2:M-1 ≡DO≡
        ≡BEGIN≡
        ≡FOR≡ I:=1:N ≡DO≡ ≡BEGIN≡
        ≡FOR≡ T:=1:L[K-1] ≡DO≡ ≡BEGIN≡
        Q[K,I,T] := A[I]*Q[K-1,I,T];
        Q[K,I,T+L[K-1]] := K*(≡FOR≡ J:=1:I-1
        ≡SUM≡ (B[I,J]*Q[K-1,J,T]) );
     ≡END≡;
      ≡IF≡ K<4 ≡THEN≡ ≡GOTO≡ QUIT;
      R:=L[K-3];  U:=2*L[K-1];
      ≡FOR≡ T:=U+1:U+L[K-2]-L[K-3] ≡DO≡ ≡BEGIN≡
      R:=R+1;
      Q[K,I,T]:=2*(≡FOR≡ J:=1:I-1 ≡SUM≡ B[I,J]*A[J])
          *Q[K-2,I,R];
                        ≡END≡;
     ≡IF≡ K<6 ≡THEN≡ ≡GOTO≡ QUIT;
     ≡FOR≡ V:=1:2 ≡DO≡ ≡BEGIN≡
       R:=L[K-4]; U:=T-1;
       ≡FOR≡ T:=U+1 : U+L[K-3]-L[K-4]-L[K-5]+L[K-6]
         ≡DO≡ ≡BEGIN≡
       R:=R+1;
       Q[K,I,T]:=3*(≡FOR≡ J:=1:I-1 ≡SUM≡
       B[I,J]*Q[2,J,V])*Q[K-3,I,R];
       ≡END≡ ; ≡END≡ ;
     ≡IF≡ K<8 ≡THEN≡ ≡GOTO≡ QUIT;
     ≡FOR≡ V:= 1:4 ≡DO≡ ≡BEGIN≡.
       R:=L[K-5]; U:=T-1;
       ≡FOR≡ T:= U+1 : U+L[K-4]-L[K-5]-L[K-6]+L[K-7]
          ≡DO≡ ≡BEGIN≡
       R:=R+1;
       Q[K,I,T]:=4*(≡FOR≡ J:=1:I-1 ≡SUM≡
       B[I,J]*Q[4,J,V])*Q[K-4,I,R];
       ≡END≡ ; ≡END≡ ;
 QUIT:
      ≡END≡ ;
       ≡END≡ ;
     ≡FOR≡ K := 1:M-1 ≡DO≡ ≡BEGIN≡
    ≡FOR≡ T := 1:L[K] ≡DO≡ ≡BEGIN≡
     DUMP := 0 ;
    ≡BEGIN≡
   X := ≡FOR≡ I := 1:N ≡SUM≡ C[I]*(Q[K,I,T]) ;
     DUMP:=5;
    O := 1/(K+1) - X ;
     DUMP := 0 ;
    ≡END≡ ;
     ≡END≡ ;
    ≡END≡ ;
   ≡END≡;
                                        TOTAL SPACE AVAILABLE    021315R
                                        EXECUTE PROGRAM          0. 0.955
0        :=  1/2 - C[1]*A[1] - C[2]*A[2] - C[3]*A[3] - C[4]*A[4]
           - C[5]*A[5];
                                                                 0. 2. 29
0        :=  1/3 - C[1]*A[1]↑2 - C[2]*A[2]↑2 - C[3]*A[3]↑2 - C[4]*A[4]↑2
           - C[5]*A[5]↑2;
                                                                 0. 2. 75
0        :=  1/3 - 2*B[2,1]*C[2]*A[1] - 2*C[3]*(B[3,1]*A[1]
           + B[3,2]*A[2]) - 2*C[4]*(B[4,1]*A[1] + B[4,2]*A[2]
           + B[4,3]*A[3]) - 2*C[5]*(B[5,1]*A[1] + B[5,2]*A[2]
           + B[5,3]*A[3] + B[5,4]*A[4]);
                                                                 0. 2.129
0        :=  1/4 - C[1]*A[1]↑3 - C[2]*A[2]↑3 - C[3]*A[3]↑3 - C[4]*A[4]↑3
           - C[5]*A[5]↑3;
                                                                 0. 2.174
0        :=  1/4 - 2*B[2,1]*C[2]*A[1]*A[2] - 2*C[3]*A[3]*(B[3,1]*A[1]
           + B[3,2]*A[2]) - 2*C[4]*A[4]*(B[4,1]*A[1] + B[4,2]*A[2]
           + B[4,3]*A[3]) - 2*C[5]*A[5]*(B[5,1]*A[1] + B[5,2]*A[2]
           + B[5,3]*A[3] + B[5,4]*A[4]);
                                                                 0. 2.229
0        :=  1/4 - 3*B[2,1]*C[2]*A[1]↑2 - 3*C[3]*(B[3,1]*A[1]↑2
           + B[3,2]*A[2]↑2) - 3*C[4]*(B[4,1]*A[1]↑2 + B[4,2]*A[2]↑2
           + B[4,3]*A[3]↑2) - 3*C[5]*(B[5,1]*A[1]↑2 + B[5,2]*A[2]↑2
           + B[5,3]*A[3]↑2 + B[5,4]*A[4]↑2);
                                                                 0. 2.287
0        :=  1/4 - 6*B[3,2]*B[2,1]*C[3]*A[1] - 3*C[4]*(2*B[4,2]*B[2,1]
           *A[1] + 2*B[4,3]*(B[3,1]*A[1] + B[3,2]*A[2]))
           - 3*C[5]*(2*B[5,2]*B[2,1]*A[1] + 2*B[5,3]*(B[3,1]*A[1]
           + B[3,2]*A[2]) + 2*B[5,4]*(B[4,1]*A[1] + B[4,2]*A[2]
           + B[4,3]*A[3])));
                                                                 0. 2.343
0        :=  1/5 - C[1]*A[1]↑4 - C[2]*A[2]↑4 - C[3]*A[3]↑4 - C[4]*A[4]↑4
           - C[5]*A[5]↑4;
                                                                 0. 2.391
0        :=  1/5 - 2*B[2,1]*C[2]*A[1]*A[2]↑2 - 2*C[3]*A[3]↑2*(B[3,1]*A[1]
           + B[3,2]*A[2]) - 2*C[4]*A[4]↑2*(B[4,1]*A[1] + B[4,2]*A[2]
           + B[4,3]*A[3]) - 2*C[5]*A[5]↑2*(B[5,1]*A[1] + B[5,2]*A[2]
           + B[5,3]*A[3] + B[5,4]*A[4]);
                                                                 0. 2.487
0        :=  1/5 - 3*B[2,1]*C[2]*A[1]↑2*A[2] - 3*C[3]*A[3]*(B[3,1]*A[1]↑2
           + B[3,2]*A[2]↑2) - 3*C[4]*A[4]*(B[4,1]*A[1]↑2 + B[4,2]*A[2]↑2
           + B[4,3]*A[3]↑2) - 3*C[5]*A[5]*(B[5,1]*A[1]↑2 + B[5,2]*A[2]↑2
           + B[5,3]*A[3]↑2 + B[5,4]*A[4]↑2);
                                                                 0. 2.547
0        :=  1/5 - 6*B[3,2]*B[2,1]*C[3]*A[1]*A[3] - 3*C[4]*A[4]*(2*B[4,2]
           *B[2,1]*A[1] + 2*B[4,3]*(B[3,1]*A[1] + B[3,2]*A[2]))
           - 3*C[5]*A[5]*(2*B[5,2]*B[2,1]*A[1] + 2*B[5,3]*(B[3,1]*A[1]
           + B[3,2]*A[2]) + 2*B[5,4]*(B[4,1]*A[1] + B[4,2]*A[2]
           + B[4,3]*A[3])));
                                                                 0. 2.606
0        :=  1/5 - 4*B[2,1]*C[2]*A[1]↑3 - 4*C[3]*(B[3,1]*A[1]↑3
```

```
         + B[3,2]*A[2]↑3) - 4*C[4]*(B[4,1]*A[1]↑3 + B[4,2]*A[2]↑3
         + B[4,3]*A[3]↑3) - 4*C[5]*(B[5,1]*A[1]↑3 + B[5,2]*A[2]↑3
         + B[5,3]*A[3]↑3 + B[5,4]*A[4]↑3);
                                                           0. 2.665
0      := 1/5 - 8*B[3,2]*B[2,1]*C[3]*A[1]*A[2] - 4*C[4]*(2*B[4,2]
         *B[2,1]*A[1]*A[2] + 2*B[4,3]*A[3]*(B[3,1]*A[1]
         + B[3,2]*A[2])) - 4*C[5]*(2*B[5,2]*B[2,1]*A[1]*A[2]
         + 2*B[5,3]*A[3]*(B[3,1]*A[1] + B[3,2]*A[2])
         + 2*B[5,4]*A[4]*(B[4,1]*A[1] + B[4,2]*A[2] + B[4,3]*A[3]));
                                                           0. 2.726
0      := 1/5 - 12*B[3,2]*B[2,1]*C[3]*A[1]↑2 - 4*C[4]*(3*B[4,2]*B[2,1]
         *A[1]↑2 + 3*B[4,3]*(B[3,1]*A[1]↑2 + B[3,2]*A[2]↑2))
         - 4*C[5]*(3*B[5,2]*B[2,1]*A[1]↑2 + 3*B[5,3]*(B[3,1]*A[1]↑2
         + B[3,2]*A[2]↑2) + 3*B[5,4]*(B[4,1]*A[1]↑2 + B[4,2]*A[2]↑2
         + B[4,3]*A[3]↑2));
                                                           0. 2.786
0      := 1/5 - 24*B[4,3]*B[3,2]*B[2,1]*C[4]*A[1]
         - 4*C[5]*(6*B[5,3]*B[3,2]*B[2,1]*A[1] + 3*B[5,4]*(2*B[4,2]
         *B[2,1]*A[1] + 2*B[4,3]*(B[3,1]*A[1] + B[3,2]*A[2])));
                                                           0. 2.833
0      := 1/5 - 4*C[2]*(B[2,1]*A[1])↑2 - 4*C[3]*(B[3,1]*A[1]
         + B[3,2]*A[2])↑2 - 4*C[4]*(B[4,1]*A[1] + B[4,2]*A[2]
         + B[4,3]*A[3])↑2 - 4*C[5]*(B[5,1]*A[1] + B[5,2]*A[2]
         + B[5,3]*A[3] + B[5,4]*A[4])↑2;
                                                           0. 2.893

                                    EXECUTION COMPLETE     0. 2.897
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

Shanks' Revision 1

```
                                       SYMBAL VERSION 1.1E      (3/14/69)
                                       START PROGRAM            0. 0.  0
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*   ≡BEGIN≡                                                               *
*    DUMP:= 0 ;                                                           *
*    MODE[1]:= 0 ;                                                        *
*     M:=8; N:=12;                                                        *
*    L:=↑0:1,1,1,1,1,2,4,8,16,32:↓;                                       *
*    A:=↑1:N:↓;                                                           *
*    C:=↑1:N:↓;                                                           *
*    B:=↑≡FOR≡ I := 1:N ≡DO≡ ↑0:I-1:↓↓;                                   *
*    ≡FOR≡ I:=3:N ≡DO≡ B[I,1]:=0;                                         *
*    ≡FOR≡ I:=4:N ≡DO≡ B[I,2]:=0;                                         *
*    ≡FOR≡ I:=5:N ≡DO≡ B[I,3]:=0;                                         *
*    Q:=↑ ≡FOR≡ K:=1:M-1 ≡DO≡                                             *
*     ↑ ≡FOR≡ I:= 1:N ≡DO≡                                                *
*     ↑1:L[K]:↓↓↓;                                                        *
*    ≡FOR≡ I:= 1:N ≡DO≡ Q[1,I,1]:=A[I];                                   *
*    ≡FOR≡ I:=1:N ≡DO≡ Q[2,I,1]:=A[I]↑2 ;                                 *
*    ≡FOR≡ I:=1:N ≡DO≡ Q[3,I,1]:=A[I]↑3;                                  *
*    ≡FOR≡ I:=1:N ≡DO≡ Q[4,I,1]:=A[I]↑4;                                  *
*    ≡FOR≡ K:=5:M-1 ≡DO≡ ≡BEGIN≡                                          *
*    ≡FOR≡ I:=1:N ≡DO≡ ≡BEGIN≡                                            *
*    ≡FOR≡ T:=1:L[K-1] ≡DO≡ ≡BEGIN≡                                       *
*     Q[K,I,T]:=A[I]*Q[K-1,I,T];                                          *
*     ≡END≡;  ≡FOR≡ T:=L[K-1]+1:L[K] ≡DO≡                                 *
*     Q[K,I,T]:=K*( ≡FOR≡ J:=1:I-1 ≡SUM≡ (B[I,J]*Q[K-1,J,T-L[K-1]]));    *
*    ≡END≡ ; ≡END≡ ;                                                      *
*    ≡FOR≡ K:=1:M-1 ≡DO≡ ≡BEGIN≡                                          *
*    ≡FOR≡ T:=1:L[K] ≡DO≡ ≡BEGIN≡                                         *
*    DUMP := 0 ;                                                          *
*    ≡BEGIN≡                                                              *
*    X:= ≡FOR≡ I:=1:N ≡SUM≡ C[I]*(Q[K,I,T]) ;                             *
*   FILE2 := ≡≡PUNCH≡≡ ;                                                  *
*     ≡IF≡ K=7 ≡THEN≡ DUMP:=5;                                            *
*    0:= 1/(K+1)-X ;                                                      *
*    ≡END≡; ≡END≡ ; ≡END≡ ; ≡END≡ ;                                       *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
                                       TOTAL SPACE AVAILABLE    122315B
                                       EXECUTE PROGRAM          0. 0.671
```

APPENDIX II

All of the coefficients used in this study are given in rational fraction form since the SYMBAL language uses only rational fractions for computational purposes. All of the coefficients with the exception of the Fehlberg 8(9), Curtis, UT 1, UT 2, and UT 3 sets were originally developed in rational fraction form. The exceptions were developed in decimal form. For this study, these decimal coefficients were taken to six decimal places and put over a denominator of one million to give the necessary rational fraction form.

THIRD ORDER COEFFICIENTS

FEHLBERG[6] 3(4)

```
C[0]:=79/490; C[1]:=0; C[2]:=2175/3626; C[3]:=2166/9065;
A[1]:=2/7; A[2]:=7/15; A[3]:=35/38; A[4]:=1; B[1,0]:=2/7;
B[2,0]:=77/900; B[2,1]:=343/900; B[3,0]:=805/1444; B[3,1]:=-77175/
54872; B[3,2]:=97125/54872;
```

CLASSICAL[11]

```
C[0]:=3/8 ; C[1]:=2/3 ; C[2]:=1/6 ;  A[1]:=1/2 ; A[2]:=1 ;
B[1,0]:=1/2; B[2,0]:=-1 ; B[2,1]:=2 ;
```

RALSTON OPTIMUM[15]

```
C[0]:=2/9; C[1]:=1/3; C[2]:=4/9; A[1]:=1/2; A[2]:=3/4;
B[1,0]:=1/2; B[2,0]:=0; B[2,1]:=3/4;
```

HEUN [7]

```
C[0]:=1/4; C[1]:=0; C[2]:=3/4; A[1]:=1/3; A[2]:=2/3;
B[1,0]:=1/3; B[2,0]:=0; B[2,1]:=2/3;
```

NYSTROM [14]

```
C[0]:=1/4; C[1]:=3/8;  C[2]:=3/8;  A[1]:=2/3; A[2]:=2/3;
B[1,0]:=2/3; B[2,0]:=0; B[2,1]:=2/3;
```

FOURTH ORDER COEFFICIENTS

FEHLBERG[6] 3(4)

```
C[0]:=229/1470; C[1]:=0; C[2]:=1125/1813; C[3]:=13718/81585;
C[4]:=1/18;
A[1]:=2/7; A[2]:=7/15; A[3]:=35/38; A[4]:=1; B[1,0]:=2/7;
B[2,0]:=77/900; B[2,1]:=343/900; B[3,0]:=805/1444; B[3,1]:=-77175/
54872; B[3,2]:=97125/54872;
                    B[4,0]:=79/490; B[4,1]:=0; B[4,2]:=2175/3626;
B[4,3]:=2166/9065;
```

FEHLBERG[5] 4(5)-1

```
C[0] := 1/9 ;  C[1] := 0 ; C[2] := 9/20 ; C[3] := 16/45 ;
C[4] := 1/12 ;
A[1] := 2/9 ; A[2] := 1/3 ; A[3] := 3/4 ; A[4] := 1 ;
B[1,0] := 2/9 ; B[2,0] := 1/12 ; B[2,1] := 1/4 ; B[3,0] := 69/128 ;
B[3,1] := -243/128 ; B[3,2] := 135/64 ; B[4,0] := -17/12 ;
B[4,1] := 27/4 ; B[4,2] := -27/5 ; B[4,3] := 16/15 ;
```

FEHLBERG[5] 4(5)-2

```
C[0] := 25   /216    ; C[1] := 0    ; C[2] := 1408  /2565   ;
C[3] := 2197  /4104    ; C[4] :=-1   /5    ; A[1] := 1   /4    ;
A[2] := 3   /8    ; A[3]:=12/13;  A[4]:=1;            B[1,0] := 1   /4 ;
B[2,0] := 3   /32    ; B[2,1] := 9   /32    ; B[3,0] := 1932 /2197   ;
B[3,1] := -7200 /2197 ; B[3,2] := 7296 /2197  ; B[4,0] := 439 /216 ;
B[4,1] := -8    ; B[4,2] := 3680 /513   ; B[4,3] := -845 /4104   ;
```

CLASSICAL[11]

```
C[0]:=1/6; C[1]:=1/3; C[2]:=1/3; C[3]:=1/6; A[1]:=1/2; A[2]:=1/2 ;
A[3]:=1; B[1,0]:=1/2; B[2,0]:=0; B[2,1]:=1/2; B[3,0]:=0; B[3,1]:=0;
B[3,2]:=1;
```

KUNTZMANN OPTIMUM[9]

```
C[0]:=55/360; C[1]:=125/360; C[2]:=125/360; C[3]:=55/360; A[1]:=2/5;
A[2]:=3/5; A[3]:=1; B[1,0]:=2/5; B[2,0]:=-3/20; B[2,1]:=3/4;
B[3,0]:=19/44; B[3,1]:=-15/44; B[3,2]:=40/44;
```

KUTTA[10]

```
C[0]:=1/8; C[1]:=3/8; C[2]:=3/8; C[3]:=1/8; A[1]:=1/3; A[2]:=2/3;
A[3]:=1; B[1,0]:=1/3; B[2,0]:=-1/3; B[2,1]:=1; B[3,0]:=1;
B[3,1]:=-1; B[3,2]:=1;
```

SARAFYAN[18]

```
C[0] := 1/6 ; C[1] := 0 ; C[2] := 2/3 ; C[3] := 1/6 ;
A[1] := 1/2 ; A[2] := 1/2 ; A[3] := 1 ;
A[4] := 2/3 ;
B[1,0] := 1/2 ; B[2,0] := 1/4 ; B[2,1] := 1/4 ; B[3,0] := 0 ;
B[3,1] := -1 ; B[3,2] := 2 ;
B[4,0] := 7/27 ; B[4,1] := 10/27 ; B[4,2] := 0 ; B[4,3] := 1/27 ;
```

UT 1[1]

```
C[0]:=124584/1000000; C[1]:=0; C[2]:=577732/1000000; C[3]:=
11167899/1000000; C[4]:=-10870216/1000000;
A[1]:=266000/1000000; A[2]:=398999/1000000; A[3]:=997475/1000000;
A[4]:=1;

B[1,0]:=266000/1000000; B[2,0]:=99749/1000000; B[2,1]:=299250/
1000000; B[3,0]:=997397/1000000; B[3,1]:=-3740204/1000000; B[3,2]:=
3740282/1000000; B[4,0]:=1018551/1000000; B[4,1]:=-3826725/1000000;
B[4,2]:=3810768/1000000; B[4,3]:=-2594/1000000;
```

UT 2[1]

C[0]:=125181/1000000; C[1]:=0; C[2]:=299910/1000000; C[3]:=437933/
1000000; C[4]:=136974/1000000;

A[1]:=203066/1000000; A[2]:=304600/1000000; A[3]:=620351/1000000;
A[4]:=1;

B[1,0]:=203066/1000000; B[2,0]:=76150/1000000; B[2,1]:=228450/
1000000; B[3,0]:=327625/1000000; B[3,1]:=-1016945/1000000;
B[3,2]:=1309671/1000000; B[4,0]:=-300307/1000000; B[4,1]:=2751166/
1000000; B[4,2]:=-2664664/1000000; B[4,3]:=1213805/1000000;

UT 3[1]

C[0]:=128196/1000000; C[1]:=0; C[2]:=275745/1000000;
C[3]:=454937/1000000; C[4]:=141120/1000000; A[1]:=200933/1000000;
A[2]:=301400/1000000; A[3]:=606170/1000000; A[4]:=1;
B[1,0]:=200933/1000000; B[2,0]:=75350/1000000;B[2,1]:=226050/1000000
; B[3,0]:=308207/1000000; B[3,1]:=-934792/1000000;
B[3,2]:=1232755/1000000; B[4,0]:=-232399/1000000;
B[4,1]:=2571843/1000000; B[4,2]:=-2609052/1000000;
B[4,3]:=1269608/1000000;

SHANKS(4-4)[19]

```
C[0]:=-179124/70092; C[1]:=200000/70092; C[2]:=40425/70092;
C[3]:=8791/70092; A[1]:=1/100; A[2]:=3/5; A[3]:=1; B[1,0]:=1/100;
B[2,0]:=-4278/245; B[2,1]:=4425/245; B[3,0]:=524746/8791;
B[3,1]:=-532125/8791; B[3,2]:=16170/8791;
```

FIFTH ORDER COEFFICIENTS

FEHLBERG[5] 4(5)-1

```
C[Ø]:=47/450; C[1]:=Ø; C[2]:=12/25; C[3]:=32/225; C[4]:=1/3Ø;
C[5]:=6/25;
A[1] := 2/9 ; A[2] := 1/3 ; A[3] := 3/4 ; A[4] := 1 ; A[5]:=5/6;
B[1,Ø] := 2/9 ; B[2,Ø] := 1/12 ; B[2,1] := 1/4 ; B[3,Ø] := 69/128 ;
B[3,1] := -243/128 ; B[3,2] := 135/64 ; B[4,Ø] := -17/12 ;
B[4,1] := 27/4 ; B[4,2] := -27/5 ; B[4,3] := 16/15 ;
                              B[5,Ø]:=65/432; B[5,1]:=-5/16; B[5,2]:=13/16;
B[5,3]:=4/27; B[5,4]:=5/144;
```

FEHLBERG[5] 4(5)-2

```
C[0]:=16/135; C[1]:=0; C[2]:=6656/12825; C[3]:=28561/56430;
C[4]:=-9/50; C[5]:=2/55;
A[1]:=1/4; A[2]:=3/8; A[3]:=12/13; A[4]:=1; A[5]:=1/2;
                                                B[1,0] := 1  /4 ;
B[2,0] := 3  /32   ; B[2,1] := 9  /32   ; B[3,0] := 1932 /2197  ;
B[3,1] := -7200 /2197 ; B[3,2] := 7296 /2197  ; B[4,0] := 439 /216 ;
B[4,1] := -8   ; B[4,2] := 3680 /513  ; B[4,3] := -845 /4104  ;
                                      B[5,0]:=-8/27; B[5,1]:=2;
B[5,2]:=-3544/2565; B[5,3]:=1859/4104; B[5,4]:=-11/40;
```

FEHLBERG[6] 5(6)

```
C[0] := 31 /384  ; C[1] := 0   ; C[2] := 1125 /2816  ;
C[3] := 9 /32  ; C[4] := 125 /768  ; C[5] := 5 /66  ; A[1] := 1 /6 ;
```

```
A[2] := 4 /15  ; A[3] := 2 /3  ; A[4] := 4 /5  ; A[5] := 1   ;
B[1,0] := 1 /6  ; B[2,0] := 4 /75  ; B[2,1] := 16 /75  ; B[3,0] :=
5 /6  ; B[3,1] := -8 /3  ; B[3,2] := 5 /2  ; B[4,0] := -8 /5  ;
B[4,1] := 144 /25  ; B[4,2] := -4   ; B[4,3] := 16 /25  ;
B[5,0] := 361 /320  ; B[5,1] := -18 /5  ; B[5,2] := 407 /128  ;
B[5,3] := -11 /80  ; B[5,4] := 55 /128  ;
```

SARAFYAN[18]

```
C[0]:=14/336; C[1]:=0; C[2]:=0; C[3]:=35/336;C[4]:=162/336;
C[5]:=125/336; A[1]:=1/2; A[2]:=1/2; A[3]:=1; A[4]:=2/3; A[5]:=2/10;
B[1,0]:=1/2; B[2,0]:=1/4; B[2,1]:=1/4; B[3,0]:=0; B[3,1]:=-1;
B[3,2]:=2; B[4,0]:=7/27; B[4,1]:=10/27; B[4,2]:=0; B[4,3]:=1/27;
B[5,0]:=28/625; B[5,1]:=-125/625; B[5,2]:=546/625; B[5,4]:=-378/
625; B[5,3]:=54/625;
```

SHANKS(5-5)[19]

```
C[0]:=105/1134; C[1]:=0; C[2]:=500/1134; C[3]:=448/1134;
C[4]:=81/1134; C[5]:=0; A[1]:=1/9000; A[2]:=3/10; A[3]:=3/4; A[4]:=1
; B[1,0]:=1/9000; B[2,0]:=-4047/10; B[2,1]:=4050/10; B[3,0]:=
20241/8; B[3,1]:=-20250/8; B[3,2]:=15/8; B[4,0]:=-931041/81;
B[4,1]:=931500/81; B[4,2]:=-490/81; B[4,3]:=112/81;
```

BUTCHER[21]

```
C[0]:=5/48; C[1]:=0; C[2]:=0; C[3]:=27/56; C[4]:=125/336;
C[5]:=1/24; A[1]:=1/5; A[2]:=2/5; A[3]:=1/3; A[4]:=4/5; A[5]:=1;
B[1,0]:=1/5; B[2,0]:=0; B[2,1]:=2/5; B[3,0]:=7/36; B[3,1]:=0;
B[3,2]:=5/36; B[4,0]:=0; B[4,1]:=0; B[4,2]:=4/5; B[4,3]:=0; B[5,0]
:=1/4; B[5,1]:=0; B[5,2]:=-35/4; B[5,3]:=54/7; B[5,4]:=25/14;
```

BUTCHER[2] 2

```
C[0]:=5/48; C[1]:=0; C[2]:=0; C[3]:=27/56; C[4]:=125/336; C[5]:=1/24
; A[1]:=-1/5; A[2]:=2/5; A[3]:=1/3; A[4]:=4/5; A[5]:=1; B[1,0]:=-1/
5; B[2,0]:=4/5; B[2,1]:=-2/5; B[3,0]:=7/36; B[3,1]:=0; B[3,2]:=5/
36; B[4,0]:=0; B[4,1]:=0; B[4,2]:=4/5; B[4,3]:=0; B[5,0]:=1/4;
B[5,1]:=0;                B[5,2]:=-35/4; B[5,3]:=54/7; B[5,4]:=25/14;
```

NEWTON-COTES, LAWSON[12]

```
C[0]:=7/90; C[1]:=0; C[2]:=32/90; C[3]:=12/90; C[4]:=32/90; C[5]:=
7/90; A[1]:=1/2; A[2]:=1/4; A[3]:=1/2; A[4]:=3/4; A[5]:=1;
B[1,0]:=1/2; B[2,0]:=3/16; B[2,1]:=1/16; B[3,0]:=0; B[3,1]:=0;
B[3,2]:=1/2; B[4,0]:=0; B[4,1]:=-3/16; B[4,2]:=6/16; B[4,3]:=9/16;
B[5,0]:=1/7; B[5,1]:=4/7; B[5,2]:=6/7; B[5,3]:=-12/7; B[5,4]:=8/7;
```

NEWTON-COTES, LUTHER[13]

```
C[0]:=7/90; C[1]:=0; C[2]:=7/90; C[3]:=32/90; C[4]:=12/90;
C[5]:=32/90; A[1]:=1; A[2]:=1; A[3]:=1/4; A[4]:=1/2; A[5]:=3/4;
B[1,0]:=1; B[2,0]:=1/2; B[2,1]:=1/2; B[3,0]:=14/64; B[3,1]:=5/64;
B[3,2]:=-3/64; B[4,0]:=-12/96; B[4,1]:=-12/96; B[4,2]:=8/96;
B[4,3]:=64/96; B[5,0]:=0; B[5,1]:=-9/64; B[5,2]:=5/64; B[5,3]:=
16/64; B[5,4]:=36/64;
```

NEWTON-COTES, BUTCHER[2]

```
C[0]:=7/90; C[1]:=0; C[2]:=32/90; C[3]:=12/90; C[4]:=32/90;
C[5]:=7/90; A[1]:=1/8; A[2]:=1/4; A[3]:=1/2; A[4]:=3/4; A[5]:=1;
B[1,0]:=1/8; B[2,0]:=0; B[2,1]:=1/4; B[3,0]:=1/2; B[3,1]:=-1;
B[3,2]:=1; B[4,0]:=3/16; B[4,1]:=0; B[4,2]:=0; B[4,3]:=9/16;
B[5,0]:=-5/7; B[5,1]:=4/7; B[5,2]:=12/7; B[5,3]:=-12/7; B[5,4]:=8/7;
```

NYSTROM[14]

```
C[0]:=23/192; C[1]:=0; C[2]:=125/192; C[3]:=0; C[4]:=-81/192;
C[5]:=125/192;  A[1]:=1/3; A[2]:=2/5; A[3]:=1; A[4]:=2/3; A[5]:=4/5;
B[1,0]:=1/3; B[2,0]:=4/25; B[2,1]:=6/25; B[3,0]:=1/4; B[3,1]:=-12/4;
B[3,2]:=15/4; B[4,1]:=6/81; B[4,1]:=90/81; B[4,2]:=-50/81; B[4,3]:=
8/81; B[5,0]:=6/75; B[5,1]:=36/75; B[5,2]:=10/75; B[5,3]:=8/75;
B[5,4]:=0;
```

SHANKS(6-6)[19]

```
C[0]:=198/3696; C[1]:=0; C[2]:=1225/3696; C[3]:=1540/3696;
C[4]:=810/3696; C[5]:=-77/3696; C[6]:=0; C[7]:=0; A[1]:=1/300;
A[2]:=1/5; A[3]:=3/5; A[4]:=14/15; A[5]:=1; B[1,0]:=1/300;
  B[2,0]:=-29/5; B[2,1]:=30/5; B[3,0]:=323/5; B[3,1]:=-330/5;
B[3,2]:=10/5; B[4,0]:=-510104/810; B[4,1]:=521640/810;
B[4,2]:=-12705/810; B[4,3]:=1925/810; B[5,0]:=-417923/77;
B[5,1]:=427350/77; B[5,2]:=-10605/77; B[5,3]:=1309/77;
B[5,4]:=-54/77;
```

UT 1[1]

```
C[0]:=124789/1000000; C[1]:=0; C[2]:=574915/1000000; C[3]:=11113447/
1000000; C[4]:=-10816403/1000000; C[5]:=3250/1000000;
A[1]:=266000/1000000; A[2]:=398999/1000000; A[3]:=997475/1000000;
A[4]:=1;             A[5]:=1/2;
B[1,0]:=266000/1000000; B[2,0]:=99749/1000000; B[2,1]:=299250/
1000000; B[3,0]:=997397/1000000; B[3,1]:=-3740204/1000000; B[3,2]:=
3740282/1000000; B[4,0]:=1018551/1000000; B[4,1]:=-3826725/1000000;
B[4,2]:=3810768/1000000; B[4,3]:=-2594/1000000;
B[5,0]:=-36199/1000000; B[5,1]:=956681/1000000; B[5,2]:=-519392/
1000000; B[5,3]:=8376573/1000000; B[5,4]:=-8277662/1000000;
```

UT 2[1]

```
C[0]:=78464/1000000; C[1]:=0; C[2]:=637651/1000000; C[3]:=
931108/1000000; C[4]:=103537/1000000; C[5]:=-750761/1000000;
A[1]:=203066/1000000; A[2]:=304600/1000000; A[3]:=620351/1000000;
A[4]:=1;  A[5]:=1/2;
B[1,0]:=203066/1000000; B[2,0]:=76150/1000000; B[2,1]:=228450/
1000000; B[3,0]:=327625/1000000; B[3,1]:=-1016945/1000000;
B[3,2]:=1309671/1000000; B[4,0]:=-300307/1000000; B[4,1]:=2751166/
1000000; B[4,2]:=-2664664/1000000; B[4,3]:=1213805/1000000;
B[5,0]:=293963/1000000; B[5,1]:=-687791/1000000; B[5,2]:=891098/
1000000; B[5,3]:=24998/1000000; B[5,4]:=-22269/1000000;
```

UT 3[1]

```
C[0]:=99466/1000000; C[1]:=0; C[2]:=397514/1000000; C[3]:=242604/
1000000; C[4]:=-72772/1000000; C[5]:=333186/1000000;
                                                A[1]:=200933/1000000;
A[2]:=301400/1000000; A[3]:=606170/1000000; A[4]:=1;
A[5]:=918110/1000000;
B[1,0]:=200933/1000000; B[2,0]:=75350/1000000;B[2,1]:=226050/1000000
; B[3,0]:=308207/1000000; B[3,1]:=-934792/1000000;
B[3,2]:=1232755/1000000; B[4,0]:=-232399/1000000;
B[4,1]:=2571843/1000000; B[4,2]:=-2609052/1000000;
B[4,3]:=1269608/1000000;
B[5,0]:=-59480/1000000; B[5,1]:=972686/1000000; B[5,2]:=-663913/
1000000; B[5,3]:=616247/1000000; B[5,4]:=52570/1000000;
```

SIXTH ORDER COEFFICIENTS

FEHLBERG[6] 5(6)

```
C[0]:=7/1408; C[1]:=0; C[2]:=1125/2816; C[3]:=9/32; C[4]:=125/768;
C[5]:=0; C[6]:=5/66; C[7]:=5/66;

                                                      A[1]  := 1 /6 ;
A[2]  := 4 /15   ; A[3]  := 2 /3   ; A[4]  := 4 /5   ; A[5]  := 1    ;
                                    A[6]:=0; A[7]:=1;
B[1,0]  := 1 /6   ; B[2,0]  := 4 /75   ; B[2,1]  := 16 /75   ; B[3,0]:=
5 /6   ; B[3,1]  := -8 /3   ; B[3,2]  := 5 /2   ; B[4,0]  := -8 /5   ;
B[4,1]  := 144 /25   ; B[4,2]  := -4     ; B[4,3]  := 16 /25   ;
B[5,0]  := 361 /320   ; B[5,1]  := -18 /5   ; B[5,2]  := 407 /128   ;
B[5,3]  := -11 /80   ; B[5,4]  := 55 /128   ;

                                                   B[6,0]:=-11/640;
B[6,1]:=0; B[6,2]:=11/256; B[6,3]:=-11/160; B[6,4]:=11/256;
B[6,5]:=0; B[7,0]:=93/640; B[7,1]:=-18/5; B[7,2]:=803/256;
B[7,3]:=-11/160; B[7,4]:=99/256; B[7,5]:=0; B[7,6]:=1;
```

FEHLBERG[6] 6(7)

```
C[0]:=77/1440; C[1]:=0; C[2]:=0; C[3]:=1771561/6289920; C[4]:=
32/105; C[5]:=243/2560; C[6]:=16807/74880; C[7]:=11/270;
A[1]:=2/33; A[2]:=4/33; A[3]:=2/11; A[4]:=1/2; A[5]:=2/3; A[6]:=6/7;
A[7]:=1; B[1,0]:=2/33; B[2,0]:=0; B[2,1]:=4/33; B[3,0]:=1/22;
B[3,1]:=0; B[3,2]:=3/22; B[4,0]:=43/64; B[4,1]:=0; B[4,2]:=-165/64;
B[4,3]:=77/32; B[5,0]:=-2383/486; B[5,1]:=0; B[5,2]:=1067/54;
B[5,3]:=-26312/1701; B[5,4]:=2176/1701; B[6,0]:=10077/4802;
B[6,1]:=0; B[6,2]:=-5643/686; B[6,3]:=116259/16807;

B[6,4]:=-6240/16807; B[6,5]:=1053/2401; B[7,0]:=-733/176;
B[7,1]:=0; B[7,2]:=141/8; B[7,3]:=-335763/23296; B[7,4]:=216/77;
B[7,5]:=-4617/2816; B[7,6]:=7203/9152;
```

BUTCHER(6-7)[2]

```
C[0]:=11/120; C[1]:=0; C[2]:=27/40; C[3]:=27/40; C[4]:=-4/15;

C[5]:=-4/15; C[6]:=11/120; A[1]:=1/3; A[2]:=2/3; A[3]:=1/3;

C[7]:=0;

A[4]:=1/2; A[5]:=1/2; A[6]:=1; B[1,0]:=1/3; B[2,0]:=0; B[2,1]:=2/3;

B[3,0]:=1/12; B[3,1]:=1/3; B[3,2]:=-1/12; B[4,0]:=-1/16;

B[4,1]:=9/8; B[4,2]:=-3/16; B[4,3]:=-3/8; B[5,0]:=0; B[5,1]:=9/8;

B[5,2]:=-3/8; B[5,3]:=-3/4; B[5,4]:=1/2; B[6,0]:=9/44; B[6,1]:=

-9/11; B[6,2]:=63/44; B[6,3]:=18/11; B[6,4]:=0; B[6,5]:=-16/11;
```

BUTCHER(6-8)[2]

```
C[0]:=41/840; C[1]:=216/840; C[2]:=0; C[3]:=27/840; C[4]:=272/840;
C[5]:=27/840; C[6]:=216/840; C[7]:=41/840; A[1]:=1/9; A[2]:=1/6;
A[3]:=1/3; A[4]:=1/2; A[5]:=2/3; A[6]:=5/6; A[7]:=1; B[1,0]:=1/9;
B[2,0]:=1/24; B[2,1]:=3/24; B[3,0]:=1/6; B[3,1]:=-3/6; B[3,2]:=4/6;
B[4,0]:=-5/8; B[4,1]:=27/8; B[4,2]:=-24/8; B[4,3]:=48/8; B[5,0]:=
221/9; B[5,1]:=-981/9; B[5,2]:=867/9; B[5,3]:=-102/9; B[5,4]:=1/9;
B[6,0]:=-783/48; B[6,1]:=678/48; B[6,2]:=-472/48; B[6,3]:=-66/48;
B[6,4]:=80/48; B[6,5]:=3/48; B[7,0]:=761/82; B[7,1]:=-2079/82;
B[7,2]:=1002/82; B[7,3]:=834/82; B[7,4]:=-454/82; B[7,5]:=-9/82;
B[7,6]:=72/82;
```

SHANKS(6-6)[19]

```
C[0]:=198/3696; C[1]:=0; C[2]:=1225/3696; C[3]:=1540/3696;
C[4]:=810/3696; C[5]:=-77/3696; C[6]:=0; C[7]:=0; A[1]:=1/300;

A[2]:=1/5; A[3]:=3/5; A[4]:=14/15; A[5]:=1; B[1,0]:=1/300;
  B[2,0]:=-29/5; B[2,1]:=30/5; B[3,0]:=323/5; B[3,1]:=-330/5;
B[3,2]:=10/5; B[4,0]:=-510104/810; B[4,1]:=521640/810;
B[4,2]:=-12705/810; B[4,3]:=1925/810; B[5,0]:=-417923/77;
B[5,1]:=427350/77; B[5,2]:=-10605/77; B[5,3]:=1309/77;
B[5,4]:=-54/77;
```

SHANKS(7-7)[19]

```
C[0]:=14/300; C[1]:=0; C[2]:=81/300; C[3]:=110/300; C[4]:=0;
C[5]:=81/300; C[6]:=14/300; C[7]:=0;
A[1]:=1/192; A[2]:=1/6; A[3]:=1/2; A[4]:=1; A[5]:=5/6; A[6]:=1;
B[1,0]:=1/192; B[2,0]:=-15/6; B[2,1]:=16/6; B[3,0]:=4867/186;
B[3,1]:=-5072/186; B[3,2]:=298/186; B[4,0]:=-19995/31; B[4,1]:=
20896/31; B[4,2]:=-1025/31; B[4,3]:=155/31; B[5,0]:=-469805/5022;
B[5,1]:=490960/5022; B[5,2]:=-22736/5022; B[5,3]:=5580/5022;
B[5,4]:=186/5022; B[6,0]:=914314/2604; B[6,1]:=-955136/2604;
B[6,2]:=47983/2604; B[6,3]:=-6510/2604; B[6,4]:=-558/2604;
B[6,5]:=2511/2604;
```

SEVENTH ORDER COEFFICIENTS

FEHLBERG[6] 6(7)

```
C[0]:=11/864; C[1]:=0; C[2]:=0; C[3]:=1771561/6289920; C[4]:=32/105;
C[5]:=243/2560; C[6]:=16807/74880; C[7]:=0; C[8]:=11/270; C[9]:=
11/270;

A[1]:=2/33; A[2]:=4/33; A[3]:=2/11; A[4]:=1/2; A[5]:=2/3; A[6]:=6/7;
A[7]:=1; A[8]:=0; A[9]:=1;

          B[1,0]:=2/33; B[2,0]:=0; B[2,1]:=4/33; B[3,0]:=1/22;
B[3,1]:=0; B[3,2]:=3/22; B[4,0]:=43/64; B[4,1]:=0; B[4,2]:=-165/64;
B[4,3]:=77/32; B[5,0]:=-2383/486; B[5,1]:=0; B[5,2]:=1067/54;
B[5,3]:=-26312/1701; B[5,4]:=2176/1701; B[6,0]:=10077/4802;
B[6,1]:=0; B[6,2]:=-5643/686; B[6,3]:=116259/16807;
B[6,4]:=-6240/16807; B[6,5]:=1053/2401; B[7,0]:=-733/176;
B[7,1]:=0; B[7,2]:=141/8; B[7,3]:=-335763/23296; B[7,4]:=216/77;
B[7,5]:=-4617/2816; B[7,6]:=7203/9152;

                         B[8,0]:=15/352; B[8,1]:=0; B[8,2]:=0;
B[8,3]:=-5445/46592; B[8,4]:=18/77; B[8,5]:=-1215/5632;
B[8,6]:=1029/18304; B[8,7]:=0; B[9,0]:=-1833/352; B[9,1]:=0;
B[9,2]:=141/8; B[9,3]:=-51237/3584; B[9,4]:=18/7; B[9,5]:=-729/512;
B[9,6]:=1029/1408; B[9,7]:=0; B[9,8]:=1;
```

FEHLBERG[6] 7(8)

```
C[0]:=41/840; C[1]:=0; C[2]:=0; C[3]:=0; C[4]:=0; C[5]:=34/105;
C[6]:=9/35; C[7]:=9/35; C[8]:=9/280; C[9]:=9/280; C[10]:=41/840;
A[1]:=2/27; A[2]:=1/9; A[3]:=1/6; A[4]:=5/12; A[5]:=1/2; A[6]:=5/6;
A[7]:=1/6; A[8]:=2/3; A[9]:=1/3; A[10]:=1; B[1,0]:=2/27; B[2,0]:=
1/36; B[2,1]:=1/12; B[3,0]:=1/24; B[3,1]:=0; B[3,2]:=1/8;
B[4,0]:=5/12; B[4,1]:=0; B[4,2]:=-25/16; B[4,3]:=25/16; B[5,0]:=
1/20; B[5,1]:=0; B[5,2]:=0; B[5,3]:=1/4; B[5,4]:=1/5; B[6,0]:=
 -25/108; B[6,1]:=0; B[6,2]:=0; B[6,3]:=125/108; B[6,4]:=-65/27;
B[6,5]:=125/54; B[7,1]:=31/300; B[7,1]:=0; B[7,2]:=0;
B[7,3]:=0; B[7,4]:=61/225; B[7,5]:=-2/9; B[7,6]:=13/900;
B[8,0]:=2; B[8,1]:=0; B[8,2]:=0; B[8,3]:=-53/6; B[8,4]:=704/45;
B[8,5]:=-107/9; B[8,6]:=67/90; B[8,7]:=3; B[9,0]:=-91/108; B[9,1]:=
0; B[9,2]:=0; B[9,3]:=23/108; B[9,4]:=-976/135; B[9,5]:=311/54;
B[9,6]:=-19/60; B[9,7]:=17/6; B[9,8]:=-1/12; B[10,0]:=2383/4100;
B[10,1]:=0; B[10,2]:=0; B[10,3]:=-341/164; B[10,4]:=4496/1025;
B[10,5]:=-301/82; B[10,6]:=2133/4100; B[10,7]:=45/82; B[10,8]:=
45/164; B[10,9]:=18/41;
```

SHANKS(7-7)[19]

```
C[0]:=14/300; C[1]:=0; C[2]:=81/300; C[3]:=110/300; C[4]:=0;
C[5]:=81/300; C[6]:=14/300; C[7]:=0; C[8]:=0; C[9]:=0; C[10]:=0;
A[1]:=1/192; A[2]:=1/6; A[3]:=1/2; A[4]:=1; A[5]:=5/6; A[6]:=1;
B[1,0]:=1/192; B[2,0]:=-15/6; B[2,1]:=16/6; B[3,0]:=4867/186;
B[3,1]:=-5072/186; B[3,2]:=298/186; B[4,0]:=-19995/31; B[4,1]:=
20896/31; B[4,2]:=-1025/31; B[4,3]:=155/31; B[5,0]:=-469805/5022;
B[5,1]:=490960/5022; B[5,2]:=-22736/5022; B[5,3]:=5580/5022;
B[5,4]:=186/5022; B[6,0]:=914314/2604; B[6,1]:=-955136/2604;
B[6,2]:=47983/2604; B[6,3]:=-6510/2604; B[6,4]:=-558/2604;
B[6,5]:=2511/2604;
```

SHANKS(7-9)[19]

```
C[0]:=110201/2140320; C[1]:=0; C[2]:=0; C[3]:=767936/2140320;
C[4]:=635040/2140320; C[5]:=-59049/2140320; C[6]:=-59049/2140320;
C[7]:=635040/2140320; C[8]:=110201/2140320; A[1]:=2/9; A[2]:=1/3;
A[3]:=1/2; A[4]:=1/6; A[5]:=8/9; A[6]:=1/9; A[7]:=5/6; A[8]:=1;
B[1,0]:=2/9; B[2,0]:=1/12; B[2,1]:=3/12; B[3,0]:=1/8    ; B[3,1]:=0;
B[3,2]:=3/8; B[4,0]:=23/216; B[4,1]:=0; B[4,2]:=21/216;
B[4,3]:=-8/216; B[5,0]:=-4136/729; B[5,1]:=0; B[5,2]:=-13584/729;
B[5,3]:=5264/729; B[5,4]:=13104/729; B[6,0]:=105131/151632;
B[6,1]:=0; B[6,2]:=302016/151632; B[6,3]:=-107744/151632;
B[6,4]:=-284256/151632; B[6,5]:=1701/151632; B[7,0]:=-775229/
1375920; B[7,1]:=0; B[7,2]:=-2770950/1375920; B[7,3]:=1735136/
1375920; B[7,4]:=2547216/1375920; B[7,5]:=81891/1375920; B[7,6]:=
328536/1375920; B[8,0]:=23569/251888; B[8,1]:=0; B[8,2]:=-122304/
251888; B[8,3]:=-20384/251888; B[8,4]:=695520/251888;
B[8,5]:=-99873/251888; B[8,6]:=-466560/251888; B[8,7]:=
241920/251888;
```

EIGHTH ORDER COEFFICIENTS

FEHLBERG[6] 7(8)

```
C[0]:=0; C[1]:=0; C[2]:=0; C[3]:=0; C[4]:=0; C[5]:=34/105; C[6]:=
9/35; C[7]:=9/35; C[8]:=9/280; C[9]:=9/280; C[10]:=0; C[11]:=41/840;
C[12]:=41/840;
A[1]:=2/27; A[2]:=1/9; A[3]:=1/6; A[4]:=5/12; A[5]:=1/2; A[6]:=5/6;
A[7]:=1/6; A[8]:=2/3; A[9]:=1/3; A[10]:=1; A[11]:=0; A[12]:=1;
                                          B[1,0]:=2/27; B[2,0]:=
1/36; B[2,1]:=1/12; B[3,0]:=1/24; B[3,1]:=0; B[3,2]:=1/8;
B[4,0]:=5/12; B[4,1]:=0; B[4,2]:=-25/16; B[4,3]:=25/16; B[5,0]:=
1/20; B[5,1]:=0; B[5,2]:=0; B[5,3]:=1/4; B[5,4]:=1/5; B[6,0]:=
-25/108; B[6,1]:=0; B[6,2]:=0; B[6,3]:=125/108; B[6,4]:=-65/27;
B[6,5]:=125/54; B[7,1]:=31/300; B[7,1]:=0; B[7,2]:=0;
B[7,3]:=0; B[7,4]:=61/225; B[7,5]:=-2/9; B[7,6]:=13/900;
B[8,0]:=2; B[8,1]:=0; B[8,2]:=0; B[8,3]:=-53/6; B[8,4]:=704/45;
```

```
B[8,5]:=-107/9; B[8,6]:=67/90; B[8,7]:=3; B[9,0]:=-91/108; B[9,1]:=
0; B[9,2]:=0; B[9,3]:=23/108; B[9,4]:=-976/135; B[9,5]:=311/54;
B[9,6]:=-19/60; B[9,7]:=17/6; B[9,8]:=-1/12; B[10,0]:=2383/4100;
B[10,1]:=0; B[10,2]:=0; B[10,3]:=-341/164; B[10,4]:=4496/1025;
B[10,5]:=-301/82; B[10,6]:=2133/4100; B[10,7]:=45/82; B[10,8]:=
45/164; B[10,9]:=18/41;

                                    B[11,0]:=3/205; B[11,1]:=0;
B[11,2]:=0; B[11,3]:=0; B[11,4]:=0; B[11,5]:=-6/41; B[11,6]:=-3/205;
B[11,7]:=-3/41; B[11,8]:=3/41; B[11,9]:=6/41; B[11,10]:=0;
B[12,0]:=-1777/4100; B[12,1]:=0; B[12,2]:=0; B[12,3]:=-341/164;
B[12,4]:=4496/1025; B[12,5]:=-289/82; B[12,6]:=2193/4100;
B[12,7]:=51/82; B[12,8]:=33/164; B[12,9]:=12/41; B[12,10]:=0;
B[12,11]:=1;
```

FEHLBERG[1] 8(9)

```
C[0]:=32256/1000000; C[1]:=0; C[2]:=0; C[3]:=0; C[4]:=0; C[5]:=0;
C[6]:=0; C[7]:=0; C[8]:=259837/1000000; C[9]:=92847/1000000;
C[10]:=164523/1000000; C[11]:=176659/1000000; C[12]:=239201/1000000;
C[13]:=3948/1000000; C[14]:=30726/1000000; A[1]:=443689/1000000;
A[2]:=665534/1000000; A[3]:=998301/1000000; A[4]:=315500/1000000;
A[5]:=505441/1000000; A[6]:=171428/1000000; A[7]:=828571/1000000;
A[8]:=665439/1000000; A[9]:=248783/1000000; A[10]:=109000/1000000;
A[11]:=891000/1000000; A[12]:=399500/1000000; A[13]:=600500/1000000;
A[14]:=1; B[1,0]:=443689/1000000; B[2,0]:=166383/1000000;
B[2,1]:=499150/1000000; B[3,0]:=249575/1000000; B[3,1]:=0;
B[3,2]:=748725/1000000; B[4,0]:=206618/1000000; B[4,1]:=0;
B[4,2]:=177078/1000000; B[4,3]:=-68197/1000000; B[5,0]:=109278/
1000000; B[5,1]:=0; B[5,2]:=0; B[5,3]:=4021/1000000; B[5,4]:=
392141/1000000;  B[6,0]:=98899/1000000; B[6,1]:=0; B[6,2]:=0;
B[6,3]:=3513/1000000; B[6,4]:=124760/1000000; B[6,5]:=-55745/
1000000; B[7,0]:=-368068/1000000; B[7,1]:=0; B[7,2]:=0; B[7,3]:=0;
```

```
B[7,4]:=-2227389/1000000; B[7,5]:=1374290/1000000; B[7,6]:=2049739/
1000000; B[8,0]:=45467/1000000; B[8,1]:=0; B[8,2]:=0; B[8,3]:=0;
B[8,4]:=0; B[8,5]:=325421/1000000; B[8,6]:=284766/1000000;
B[8,7]:=9783/1000000; B[9,0]:=60842/1000000; B[9,1]:=0; B[9,2]:=0;
B[9,3]:=0; B[9,4]:=0; B[9,5]:=-21184/1000000; B[9,6]:=195965/1000000
; B[9,7]:=-4274/1000000; B[9,8]:=17434/1000000; B[10,0]:=54059/
1000000; B[10,1]:=0; B[10,2]:=0; B[10,3]:=0; B[10,4]:=0; B[10,5]:=0;
B[10,6]:=110298/1000000; B[10,7]:=-1256/1000000; B[10,8]:=3679/
1000000; B[10,9]:=-57780/1000000; B[11,0]:=127324/1000000; B[11,1]:=
0; B[11,2]:=0; B[11,3]:=0; B[11,4]:=0; B[11,5]:=0; B[11,6]:=0;
B[11,7]:=114488/1000000; B[11,8]:=287730/1000000; B[11,9]:=509453/
1000000; B[11,10]:=-147996/1000000; B[12,0]:=-3652/1000000;
B[12,1]:=0; B[12,2]:=0; B[12,3]:=0; B[12,4]:=0; B[12,5]:=81629/
```

```
1000000; B[12,6]:=-386077/1000000; B[12,7]:=30862/1000000;
B[12,8]:=-58077/1000000; B[12,9]:=335986/1000000; B[12,10]:=410668/
1000000; B[12,11]:=-11840/1000000; B[13,0]:=-1237535/1000000;
B[13,1]:=0; B[13,2]:=0; B[13,3]:=0; B[13,4]:=0; B[13,5]:=-24430768/
1000000; B[13,6]:=547795/1000000; B[13,7]:=-4441386/1000000;
B[13,8]:=10013104/1000000; B[13,9]:=-14995773/1000000; B[13,10]:=
5894694/1000000; B[13,11]:=1738037/1000000; B[13,12]:=27512330/
1000000; B[14,0]:=-352608/1000000; B[14,1]:=0; B[14,2]:=0; B[14,3]:=
0; B[14,3]:=0; B[14,4]:=0; B[14,5]:=-183961/1000000; B[14,6]:=
  -655701/1000000; B[14,7]:=-390861/1000000; B[14,8]:=267946/
1000000; B[14,9]:=-1038302/1000000; B[14,10]:=1667232/1000000;
B[14,11]:=495519/1000000; B[14,12]:=1139400/1000000; B[14,13]:=
  51336/1000000;
```

SHANKS(8-10)[19]

```
C[0]:=41/840; C[1]:=0; C[2]:=0; C[3]:=27/840; C[4]:=272/840;
C[5]:=27/840; C[6]:=216/840; C[7]:=0; C[8]:=216/840; C[9]:=41/840;
A[1]:=4/27; A[2]:=2/9; A[3]:=1/3; A[4]:=1/2; A[5]:=2/3; A[6]:=1/6;
A[7]:=1; A[8]:=5/6; A[9]:=1;   B[1,0]:=4/27; B[2,0]:=1/18;
B[2,1]:=3/18; B[3,0]:=1/12; B[3,1]:=0; B[3,2]:=3/12; B[4,0]:=1/8;
B[4,1]:=0; B[4,2]:=0; B[4,3]:=3/8; B[5,0]:=13/54; B[5,1]:=0; B[5,2]
:=-27/54; B[5,3]:=42/54; B[5,4]:=8/54; B[6,0]:=389/4320; B[6,1]:=0;
B[6,2]:=-54/4320; B[6,3]:=966/4320; B[6,4]:=-824/4320;
B[6,5]:=243/4320; B[7,0]:=-231/20; B[7,1]:=0; B[7,2]:=81/20;
B[7,3]:=-1164/20; B[7,4]:=656/20; B[7,5]:=-122/20; B[7,6]:=40;
B[8,0]:=-127/288; B[8,1]:=0; B[8,2]:=18/288; B[8,3]:=-678/288;
B[8,4]:=456/288; B[8,5]:=-9/288; B[8,6]:=576/288; B[8,7]:=4/288;
B[9,0]:=1481/820; B[9,1]:=0; B[9,2]:=-81/820; B[9,3]:=7104/820;
B[9,4]:=-3376/820; B[9,5]:=72/820; B[9,6]:=-5040/820; B[9,7]:=-60/
820; B[9,8]:=720/820;
```

SHANKS(8-12)[19]

```
C[0]:=41/840; C[1]:=0; C[2]:=0; C[3]:=0; C[4]:=0; C[5]:=216/840;
C[6]:=272/840; C[7]:=27/840; C[8]:=27/840; C[9]:=36/840; C[10]:=
180/840; C[11]:=41/840; C[12]:=0; A[1]:=1/9; A[2]:=1/6; A[3]:=1/4;
A[4]:=1/10; A[5]:=1/6; A[6]:=1/2; A[7]:=2/3; A[8]:=1/3; A[9]:=5/6;
A[10]:=5/6; A[11]:=1; B[1,0]:=1/9; B[2,0]:=1/24; B[2,1]:=3/24;
B[3,0]:=1/16; B[3,1]:=0; B[3,2]:=3/16; B[4,0]:=29/500; B[4,1]:=0;
B[4,2]:=33/500; B[4,3]:=-12/500; B[5,0]:=33/972; B[5,1]:=0;
B[5,2]:=0; B[5,3]:=4/972; B[5,4]:=125/972; B[6,0]:=-21/36;
B[6,1]:=0; B[6,2]:=0; B[6,3]:=76/36; B[6,4]:=125/36; B[6,5]:=
-162/36; B[7,0]:=-30/243; B[7,1]:=0; B[7,2]:=0; B[7,3]:=-32/243;
B[7,4]:=125/243; B[7,5]:=0; B[7,6]:=99/243; B[8,0]:=1175/324;
B[8,1]:=0; B[8,2]:=0; B[8,3]:=-3456/324; B[8,4]:=-6250/324;
B[8,5]:=8424/324; B[8,6]:=242/324; B[8,7]:=-27/324; B[9,0]:=293/324;
B[9,1]:=0; B[9,2]:=0; B[9,3]:=-852/324; B[9,4]:=-1375/324;
```

```
B[9,5]:=1836/324; B[9,6]:=-118/324; B[9,7]:=162/324; B[9,8]:=1;
B[10,0]:=1303/1620; B[10,1]:=0; B[10,2]:=0; B[10,3]:=-4260/1620;
B[10,4]:=-6875/1620; B[10,5]:=9990/1620; B[10,6]:=1030/1620;
B[10,7]:=0; B[10,8]:=0; B[10,9]:=162/1620; B[11,0]:=-8595/4428;
B[11,1]:=0; B[11,2]:=0; B[11,3]:=30720/4428; B[11,4]:=48750/4428;
B[11,5]:=-66096/4428; B[11,6]:=378/4428; B[11,7]:=-729/4428;
B[11,8]:=-1944/4428; B[11,9]:=-1296/4428; B[11,10]:=3240/4428;
```

CURTIS[3]

```
C[0]:=1/20; C[1]:=0; C[2]:=0; C[3]:=0; C[4]:=0; C[5]:=13/180;
C[6]:=36/180; C[7]:=64/180; C[8]:=1/5; C[9]:=13/180; C[10]:=1/20;
C[11]:=0; C[12]:=0; A[1]:=183855/1000000; A[2]:=183855/1000000;
A[3]:=275775/1000000; A[4]:=689439/1000000; A[5]:=827326/1000000;
A[6]:=172673/1000000; A[7]:=1/2; A[8]:=827326/1000000;
```

```
A[9]:=172673/1000000; A[10]:=1; B[1,0]:=183850/1000000;
B[2,0]:=92128/1000000; B[2,1]:=91922/1000000; B[3,0]:=68943/1000000;
B[3,1]:=0; B[3,2]:=206831/1000000; B[4,0]:=689439/1000000; B[4,1]:=
0; B[4,2]:=-2585396/1000000; B[4,3]:=2585396/1000000; B[5,0]:=82732/
1000000; B[5,1]:=0; B[5,2]:=0; B[5,3]:=413663/1000000; B[5,4]:=
330930/1000000; B[6,0]:=97115/1000000; B[6,1]:=0; B[6,2]:=0;
B[6,3]:=97308/1000000; B[6,4]:=-44005/1000000; B[6,5]:=22254/
1000000; B[7,0]:=-623200/1000000; B[7,1]:=0; B[7,2]:=0;
B[7,3]:=-259948/1000000; B[7,4]:=249582/1000000; B[7,5]:=-113452/
1000000; B[7,6]:=686138/1000000; B[8,0]:=308186/1000000; B[8,1]:=0;
B[8,2]:=0; B[8,3]:=519705/1000000; B[8,4]:=-761906/1000000;
B[8,5]:=419861/1000000; B[8,6]:=-629640/1000000; B[8,7]:=971118/
1000000; B[9,0]:=139493/1000000; B[9,1]:=0; B[9,2]:=0; B[9,3]:=
117240/1000000; B[9,4]:=-249427/1000000; B[9,5]:=-26481/1000000;
B[9,6]:=-118354/1000000; B[9,7]:=182542/1000000; B[9,8]:=127657/
1000000; B[10,0]:=-499040/1000000; B[10,1]:=0; B[10,2]:=0; B[10,3]:=
-1386394/1000000; B[10,4]:=1331109/1000000; B[10,5]:=-674026/
1000000; B[10,6]:=1119619/1000000; B[10,7]:=-592592/1000000;
B[10,8]:=506298/1000000; B[10,9]:=1195027/1000000;
```

BIBLIOGRAPHY

1. Bettis, D. G., private communication, (1971).

2. Butcher, J. C., "Coefficients for the Study of Runge-Kutta Integration Processes," J. Austral. Math. Soc. 3, 185 (1963).

3. Curtis, A. R., "An Eighth Order Runge-Kutta Process with Eleven Function Evaluations Per Step," Numer. Math. 16, 268 (1970).

4. Engeli, M. E., User's Manual for the Formula Manipulation Language SYMBAL, The University of Texas at Austin, TRM-8.01 (1969).

5. Fehlberg, E., "Low Order Classical Runge-Kutta Formulas with Step-Size Control and Their Application to Some Heat Transfer Problems," NASA Technical Report, NASA TR R-315 (1969).

6. Fehlberg, E., "Some Experimental Results Concerning the Error Propagation in Runge-Kutta Type Integration Formulas," NASA Technical Report, NASA TR R-352 (1970).

7. Heun, K., "Neue Methode zur Approximativen Integration der Differentialgleichungen einer Unabhängigen Variable," Z. Angew. Math. Phys. 45, 28 (1900).

8. Ince. E. L., Ordinary Differential Equations, New York: Dover (1956).

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

10. Kutta, W., Z. Math. Phys. 46, 435 (1901).

11. Lapidus, L. and J. H. Seinfeld, Numerical Solutions of Ordinary Differential Equations, New York: Academic Press (1971).

12. Lawson, J. D., "An Order Five Runge-Kutta Process with Extended Region Stability," SIAM J. Numer. Anal. 3, 593 (1966).

13. Luther, H. A., "Further Explicit Fifth Order Runge-Kutta Formulas," SIAM Rev. 8, 374 (1966).

14. Nyström, E. J., "Uber die Numerische Integration von Differentialgleichungen," Acta Soc. Sci. Fenn. 50, No. 13, 1-55 (1925).

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

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

17. Runge, C., Math. Ann. 46, 167 (1895).

18. Sarafyan, D., "Error Estimation for Runge-Kutta Methods Through Pseudo-Iterative Formulas," Louisiana State University, Technical Report No. 14 (1966).

19. Shanks, E. B., "Solutions of Differential Equations by Evaluations of Functions," Math. Comp. 20, 21 (1966).

20. Zurmuhl, R., "Runge-Kutta Verfahren zur Numerischen Integration von Differentialgleichungen n-ter Ordnung," Z. Angew. Math. Mech. 28, 173 (1948).