Title
RKML: AN ALGOL 60 PROGRAM THAT OBTAINS THE PARAMETER DEFINING EQUATIONS FOR GENERALIZED RUNGE-KUTTA-FREY INTEGRATION SCHEMES

Author
Close, Elon R.

Publication Date
1970-06-17

For Conference on Second Symp. on
Symbolic and Algebraic Manipulation,
Los Angeles, Ca., March 23-25, 1971

UCRL-19861
Preprint

c. 2

RKM1: AN ALGOL 60 PROGRAM THAT OBTAINS THE PARAMETER DEFINING EQUATIONS FOR GENERALIZED RUNGE-KUTTA-FREY INTEGRATION SCHEMES

Elon R. Close

June 17, 1970

AEC Contract No. W-7405-eng-48

LAWRENCE RADIATION LABORATORY
UNIVERSITY of CALIFORNIA BERKELEY

DISCLAIMER

UCRL-19861

RKM1: AN ALGOL 60 PROGRAM THAT OBTAINS THE PARAMETER DEFINING EQUATIONS FOR GENERALIZED RUNGE-KUTTA-FREY INTEGRATION SCHEMES

Elon R. Close

Lawrence Radiation Laboratory
University of California
Berkeley, California

ABSTRACT

An ALGOL 60 program, RKM1, that derives the parameter defining equations associated with numerical integration schemes for ordinary differential equations to the form $D^p x = X(D^{p-1}x, \ldots, x)$ is discussed. A simple classical third order Runge-Kutta scheme is presented in detail, and it is shown how more complicated schemes can also be treated.

I. INTRODUCTION

In the following sections, we shall discuss an ALGOL 60 program RKM1 that derives the nonlinear parameter defining equations associated with numerical integration schemes used to solve ordinary differential equation initial value problems. We shall first define the problem and then illustrate how these equations are obtained using RKM1 by treating a classical third order Runge-Kutta scheme in detail. To show the ease with which other schemes can be constructed, we show how to set up some other examples: one from Butcher [1], R. DeVogelaere's scheme for second order equations [2], and a classical finite difference scheme of the Adams type [3].

The program RKM1 is described in detail in [4] and so is the theory upon which it is developed. In general, this work is an extension of ideas and results of R. De Vogelaere [5], Ceschino-Kuntzmann [3], and J. C. Butcher [6].

II. DEFINITION OF THE PROBLEM

We wish to derive the parameter defining equations that are associated with numerical integration schemes used to solve the initial value problem

$$D^p x = X(\xi), \quad \xi(a) = b \tag{2.1}$$

where

$$x \in R \to R^n, \; b \in R^n, \; a \in R$$

$$\xi = (D^{p-1}x, \ldots, x) \in R \to R^{n\times p}$$

$$X \in R^{n\times p}$$

and R is the real line, $R^{n\times p}$ and R^n are real $n\times p$ and n dimensional vector spaces.

For the sake of simplicity, we shall take $p = 1$. However, RKM1 can handle arbitrary p. The limitation on the value of p is determined by the storage requirements. The results that follow are given without proofs and in a rather informal style; however, precise definitions and rigorous proofs can be found in [4].

The class of schemes that we admit for the solution of (2.1) can be described in the following manner.

Given an interval of integration, we construct a set of approximations ξ_i to the true solution $\xi(t_i)$ by forming linear combinations of the approximations ξ_j, function values $X_j = X(\xi_j)$ and we also admit, after the fashion of Frey [7], the use of the derivative values $DX(\xi_j)$. Thus, our schemes can be written as

$$\xi_i = \Sigma\, a_j\, \xi_j + \Sigma\, b_j\, \eta_j \tag{2.2}$$

where $\eta_j = X(\xi_j)$ or else $DX(\xi_j)$ times a suitably constructed quantity. Since nothing is said about the order of accuracy of the approximations, (2.2) contains classical Runge-Kutta schemes, finite difference schemes, predictor-corrector schemes, and the more recent schemes utilizing off step points of lower accuracy along with points from the past. We call our schemes generalized Runge-Kutta (GRK) schemes; or, if the first derivative is utilized, then generalized Runge-Kutta-Frey (GRKF) schemes. The program RKM1 is built to handle GRKF schemes.

We shall, for the purpose of illustrating the program, derive the classical third order Runge-Kutta equations (RK3) for a first order system of equations. Thus, the problem is

$$\begin{aligned} Dx &= X(x) \\ x(a) &= b \end{aligned} \tag{2.3}$$

and the scheme can be defined in the following manner. Let

$$\eta_o = X(\xi_3);\ \eta_i = X(\xi_2);\ \eta_2 = X(\xi_1) \tag{2.4}$$

then the scheme is

$$\begin{aligned} \xi_2 &= \xi_3 + \eta_o \\ \xi_1 &= \xi_3 + \eta_o + \eta_1 \\ \xi_o &= \xi_3 + \eta_o + \eta_1 + \eta_2 \\ \xi_o &= x(t_o) + \ (h^4) \end{aligned} \tag{2.5}$$

where we have purposefully left out the parameters of the scheme (2.5) and we have also numbered the approximations in a reverse fashion. Thus,

$$t_3 = t \leq t_2 \leq t_1 \leq t_o = t + h.$$

where ξ_i being an approximation to $\xi(t_i)$.

III. DERIVATION OF THE EQUATIONS USING RKM1

In order to set up the data and interpret the results, it is necessary to sketch the manner in which the equations are obtained. All results are based on a Taylor's expansion of the solution and of the approximations.

We assume that there exists a set of basic elements A_i so that

$$\begin{aligned} \xi_j &= \Sigma\, \alpha_{ji}\, A_i \\ \xi(t_j) &= \Sigma\, \beta_{ji}\, A_i\ . \end{aligned} \tag{3.1}$$

Then the parameter defining equations become

$$\alpha_{oi} = \beta_{oi} \quad , \quad i = 1, 2, 3, \ldots \tag{3.2}$$

where α_{oi} are functions of the parameters α_{ji}. We also assume that given an approximation $\xi = \Sigma\, \alpha\, A$ there exists harmonics γ so that we can perform the substitution

$$\Sigma\, \gamma\, A = X(\Sigma\, \alpha\, A)$$

or the multiplication

$$\Sigma\, \gamma\, A = DX(\Sigma\, \beta\, A) \times \Sigma\, \beta\, A.$$

RKM1 works in the coefficent space and faithfully constructs the scheme as given in (2.4) and (2.5) by carrying out the substitutions and linear combinations. The results it outputs are the harmonics of the constructed quantities.

For the particular example chosen, RK3, the data input is given in Table I and the results obtained are given in Table II. We shall first discuss the data input of Table I. In connection with this, the schematic interval representation given in Table I will prove helpful.

The input data to RKM1 can be separated into three general types. First, there are quantities that set program parameters. These deal principally with storage requirements and the output format desired. They are relatively fixed and few in number. These quantities are inclosed in square boxes in Table I.

Secondly, there are data tables that allow the program to carry out the operations of summation and substitution indicated in (2.4) and (2.5). These tables are fixed for classes of schemes. For example, the tables required for RK3 are a subset of those required for RK4. These tables could themselves be specified (generated) by means of a suitably constructed program and, thus, we shall consider them to be given. Their construction is fully described in [4].

Finally, there is the data necessary to specify the initial value problem and the scheme. This data is labled A, B, and C in Table I.

The quantities input in A are output in Table II, Section A under the heading 'data input that particularizes the problem'. For our example, we have chosen a first order system. The first derivative that appears in our expansions is D^1X since we know $Dx = X$. The number of points in one h interval is 3, the number of h intervals is 1 since this is a Runge-Kutta scheme, and the scheme repeats itself after one h interval. To represent the derivatives D^2x, D^3x, D^4x, we need seven functions A_i, $i = 0, \ldots, 6$, thus there are seven harmonics for three derivatives.

In Section B, we input an identifying comment that appears in the output.

Section C defines the scheme, here RK3, given in (2.4) and (2.5).

The quantity (1, true, j) means substitute $X(\xi_j)$. The quantity (3, i, n, ...) means construct a new $\xi_i = \sum_{j=1}^{n}$ by forming a linear combination of the n items that follow. An item is specified by (type, j) where (0, j) are ξ_j and (1, j) are η_j. Thus, (1, true, 3) = $X(\xi_3)$ and (3,2,2, 0,3,1,0) = $\xi_2 = \xi_3 + \eta_0$. As the scheme is constructed, its definition is output as Section C of Table II.

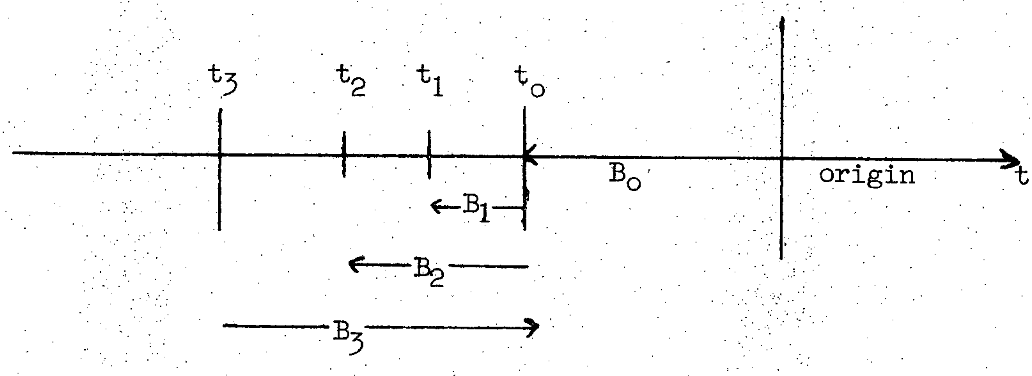
We, thus, have a very easy direct data representation of the scheme. The definition of the scheme essentially in itself defines the input to RKM1.

The actual equation output corresponding to this scheme begins in Section D of Table II and we shall now describe that. We have appended to that output some notes that should help clarify the printout. In particular, the power of h corresponding to the harmonic. We should keep in mind that $\xi_0 - \xi(t_0) = \mathcal{O}(h^4)$ and that $\xi_i - \xi(t_i) = \mathcal{O}(h^2)$ is also true for RK3; the latter is, in fact, true for all RK schemes.

The interval parameters $B_6, \ldots, B_9$ that appear in the parameter equations are defined in terms of $B_0, B_1, \ldots, B_3$ which appear in Figure 1. The location of the origin establishes the actual numerical values of these parameters and normally RK schemes are presented with the origin located at t_3; that is, $B_0 = h$. However, placing the origin at $B_0 = t_0$ is advantageous when dealing with general schemes.

In Section E, we present the equations that arise from requiring ξ_3, ξ_2, ξ_1, ξ_0 to agree respectively with $\xi(t_3), \ldots, \xi(t_0)$ through the first derivative Dx. We do this since Dx is known and it reduces the number of harmonics used thus reducing the storage requirements. These equations should evaluate to zero and can easily be checked by using undetermined

<u>Figure 1</u> Interval Parameters for RK3



parameters and applying the scheme to the polynomials $x = \frac{t^j}{j!}$ for $j = 0, 1, \ldots$. Thus, for a finite difference scheme these are all the equations that define the parameters. They will, however, extend to higher orders since we would require the approximations to agree to a higher order with the true solution.

The equations in F are the usual RK equations. They should evaluate to zero through h^3 and the h^4 values of C give the local truncation error. There is a very definite pattern to their generation and can be checked visually rather easily. Details are given in [4].

The equations in G serve to define the parameters $B_{17}, \ldots, B_{23}$ and $B_{24}, \ldots, B_{30}$ that are the harmonics of ξ_1 and ξ_2 respectively that are actually used in Section F. Thus, for example, B_{17} that appears in C_{413} is, in reality, the harmonic defined by setting $C_{5,1,1} = 0$ in Section G. The program automatically uses these undetermined harmonics whenever the approximation is used before it is constructed. In this example, we performed all substitutions $X(\xi_3)$, $X(\xi_2)$, $X(\xi_1)$ first and thus forced their use. After ξ_2 and ξ_1 were constructed, the program equated the undetermined parameters to the constructed harmonics, thus, arriving at the equations in Section G. If we, instead, constructed ξ_1, ξ_2 and used the constructed versions, the final equations arrived at would be those that arise by substituting in Section F the value of the parameters $B_{17}, \ldots, B_{30}$ and distributing all the terms in the products. This leads to the use of more storage and also one loses the pattern of development of the RK equations.

Section H shows that ξ_1 and ξ_2 have been used with undetermined parameters as harmonics with

$$\xi_1 = x(0) + B_1 \, Dx(0) + \sum_{i=0}^{6} B_{17+i} \, A_i$$

$$\xi_2 = x(0) + B_2 \, Dx(0) + \sum_{i=0}^{6} B_{24+i} \, A_i$$

where B_1 and B_2 are the distance from the point t_o.

Section I is a short dump that allows an estimate of how much storage was required. We have used only 622 locations in our list so it could have been shorter than the 5000 that was allocated to it.

The equations are output in FORTRAN; a simple input parameter selects FORTRAN or ALGOL, and can easily be punched and transferred to another program. For a known scheme, they are easy to verify; for new schemes, a solution may be hard to find.

We note in passing that the execution time for our example of generating the RD3 parameter equations was 14.9 central processor seconds using the CDC 6600 located at the Lawrence Radiation Laboratory at Berkeley, California.

To illustrate the ease with which schemes can be generated, we give in Tables III, IV, and V the data sections A, B, C needed to generate the schemes of R. DeVogelaere [2] for $D^2x = X(x)$, an example from Butcher's work [1], and a finite difference scheme of the Adams type [3]. The data tables are presumed known. The other program data need not be changed. Further details and examples can be found in [4].

IV. OBSERVATIONS

We have not presented here any details on how the program works since this is fully documented in [4]. A short summary would be that the scheme generation problem has been defined in a discrete coefficient space by using appropriate expansions. Suitable theorems have been developed to allow us to work in this space. Then, a program that reflects this structure has been built to carry out the work using lists to represent the constructed quantities.

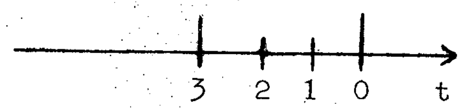
We also note that RKM1 is a rather large 'nut cracker' to use on many well known schemes. The equations it generates are only one representation of equations that can be obtained by other means for Runge-Kutta and Finite difference schemes. However, there is a proliferation of schemes that have been published and it is impossible to check whether coefficients are correct or to, sometimes, extend their accuracy. RKM1 easily does a lot of the work by furnishing the parameter defining equations for a minimum amount of work.

Finally, the interest in offstep schemes can lead to a formidable problem of obtaining these equations, especially when points from the past are used. These equations can easily be obtained using RKM1, the problem then becomes to find a solution for them.

Table I. RK3 Worksheet for RKM1 Input

Problem Dx = X(x) Order = 1

Scheme RK3



$N_o = X(E_3)$; $N_1 = X(E_2)$; $N_2 = X(E_1)$;

$E_2 = E_3 + N_o$

$E_1 = E_3 + N_o + N_1$

$E_o = E_3 + N_o + N_1 + N_2$

rank = 3
extent = 1
period = 1
3 derivatives require 7 harmonics

RKM1 Input Data

1, 1,1,1,1,1,3,3,3,1,2,1,1,0, 2,
50,72,6000,7000,10,10,10

1,1,3,1,1,7,3, A Scheme Parameters

false,

Data table(3,7,0,+,1)

comment RK3, third order Runge-Kutta; B A Comment

0,-1

1,true,3, 1,true,2, 1,true,1,
3,2,2, 0,3, 1,0,
3, ,3, 0,3, 1,0,1,1, C The Scheme
3,0,4, 0,3, 1,0,1,1,1,2

4,-1
-4,false

Table II RK3

```
     CONTROL .=      1

 DATA INPUT THAT SETS COMPUTER VARIABLES

I.= O., 'FOR' PRINT LENGTH(/I/).=  1,  1,  1,  1,  1,  3,  3,  3,  1,  2,  1,
  1,  0 'DO' I.=I+1.,
     TYPESET .=     2
     N FOR PRINT .=      50
     LINE LENGTH .=      72
     ORIGIN OF TEMPORARY STORE .=  60C0
     LIST LENGTH .=  7000
     MAXIMUM NUMBER OF VECTORS  N OR SUMS  S .=      10
     MAXIMUM LENGTH OF A SUM .=      10
     MAXIMUM LENGTH OF A LIST PRODUCT .=      10

DATA INPUT THAT PARTICULARIZES THE PROBLEM


     ORDER OF THE DIFFERENTIAL EQUATION .=      1
      D'POWER'(ORDER+UPPER)X IS THE FIRST NEGLECTED  TAYLOR    TERM, UPPER .=
     1
     NUMBER OF POINTS IN ONE H INTERVAL .=      3
     NUMBER OF H INTERVALS .=      1
     PERIOD OF SCHEME .=      1
     NUMBER OF BASIC FUNCTIONS .=      7
     NUMBER OF DERIVATIVES WE ATTEMPT TO MATCH .=      3
                                       INOUT .= 'FALSE'

DATA TABLE INPUT
```

A.

THE OUTPUT IS SUPPRESSED

Table II RK3

```
DUMMY PRINT SCHEME.  PRINT SCHEME.= 'FALSE'

'COMMENT'' THIRD ORDER RUNGE KUTTA RK3.,
```

B.

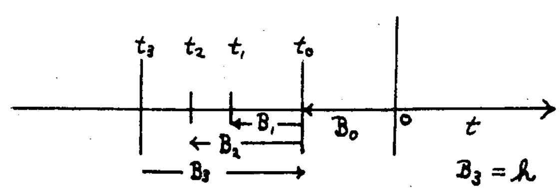
C.

```
 N(/  0/).=  X( E(/  3/))

 N(/  1/).=  X( E(/  2/))

 N(/  2/).=  X( E(/  1/))

 E(/  2,  0/).=
SUM(I,0,  0,SUM(J,0,  0, B(/  31 + I *  1 + J/) *  E(/V(/  0,I/), J/)))+
SUM(I,0,  0, B(/  32 + I/) *  N(/V(/  1,I/)/))

WHERE  V(/   0/).=(   3)  AND  V(/   0,I/).= V(/   0/)(/I/)
WHERE  V(/   1/).=(   0)  AND  V(/   1,I/).= V(/   1/)(/I/)

 E(/  1,  0/).=
SUM(I,C,  0,SUM(J,0,  0, B(/  33 + I *  1 + J/) *  E(/V(/  0,I/), J/)))+
SUM(I,0,  1, B(/  34 + I/) *  N(/V(/  1,I/)/))

WHERE  V(/   0/).=(   3)  AND  V(/   0,I/).= V(/   0/)(/I/)
WHERE  V(/   1/).=(   0,   1)  AND  V(/   1,I/).= V(/   1/)(/I/)

 E(/  0,  C/).=
SUM(I,0,  0,SUM(J,0,  0, B(/  36 + I *  1 + J/) *  E(/V(/  0,I/), J/)))+
SUM(I,0,  2, B(/  37 + I/) *  N(/V(/  1,I/)/))

WHERE  V(/   0/).=(   3)  AND  V(/   0,I/).= V(/   0/)(/I/)
WHERE  V(/   1/).=(   0,   1,   2)  AND  V(/   1,I/).= V(/   1/)(/I/)

 E(/0/).=  E(0) + ERROR TERMS.   ALL DEFINING EQUATIONS ARE PRINTEC BELOW

C(/I,K,N/), WHERE K IS IN(0,...,ORDER-1) AND N IS IN (0,...,IM(/0/)-1)
'COMMENT' EQUATIONS ARISING FROM REQUIRING
 E(/.../).= U(/.../)(...) + SUM(I,0,IM(/0/)-1, B(/.../)* A(...)(/I/)).,
```

SCHEME

ξ				
ξ	ξ_3	η_0	η_1	η_2
2	B_{31}	B_{32}		
1	B_{33}	B_{34}	B_{35}	
0	B_{36}	B_{37}	B_{38}	B_{39}

```
B( 6)=B 0
B( 7)=B 0 + B( 1)
B( 8)=B 0 + R( 2)
B( 9)=B 0 -1 * B( 3)
```



D.

Table II RK3

```
'COMMENT' CONDITICNS ON  E(/  0/).,      E_0 - E(t_0)                E.

h^0     C(  1,  1,  1)=
       .( 1./ 1.) * B( 36) - ( 1./ 1.)
h^1     C(  1,  1,  2)=
       .( 1./ 1.) * B( 9) * B( 36) + ( 1./ 1.) * B( 37) + ( 1./ 1.) * B
       .( 38) + ( 1./ 1.) * B( 39) - ( 1./ 1.) * B( 6)
'COMMENT' CONDITICNS ON  E(/  1/).,      E_1 - E(t_1)

h^0     C(  2,  1,  1)=
       .( 1./ 1.) * B( 33) - ( 1./ 1.)
h^1     C(  2,  1,  2)=
       .( 1./ 1.) * B( 9) * B( 33) + ( 1./ 1.) * B( 34) + ( 1./ 1.) * B
       .( 35) - ( 1./ 1.) * B( 7)
'COMMENT' CONDITICNS ON  E(/  2/).,      E_2 - E(t_2)

h^0     C(  3,  1,  1)=
       .( 1./ 1.) * B( 31) - ( 1./ 1.)
h^1     C(  3,  1,  2)=
       .( 1./ 1.) * B( 9) * B( 31) + ( 1./ 1.) * B( 32) - ( 1./ 1.) * B
       .( 8)
'COMMENT' EQUATIONS ARISING FROM  E(/C,K/) -  E(0)(/K/).=  O(H'POWER'(K+  5)).,     F.     E_0 - E(t_0)
h^2     C(  4,  1,  1)=
       . + ( 1./ 2.) * B( 9)** 2 * B( 36)
       . + ( 1./ 1.) * B( 9) * B( 37) + ( 1./ 1.) * B( 8) * B( 38) + (
       . 1./ 1.) * B( 7) * B( 39)
h^3     C(  4,  1,  2)=
       . + ( 1./ 6.) * B( 9)** 3 * B( 36)
       . + ( 1./ 2.) * B( 9)** 2 * B( 37) + ( 1./ 2.) * B( 8)** 2 * B(
       . 38) + ( 1./ 2.) * B( 7)** 2 * B( 39)
h^3     C(  4,  1,  3)=
       . + ( 1./ 6.) * B( 9)** 3 * B( 36)
       . + ( 1./ 2.) * B( 9)** 2 * B( 37) + ( 1./ 1.) * B( 24) * B(
       . 38) + ( 1./ 1.) * B( 17) * B( 39)
h^4     C(  4,  1,  4)=
       . + ( 1./ 24.) * B( 9)** 4 * B( 36)
       . + ( 1./ 6.) * B( 9)** 3 * B( 37) + ( 1./ 6.) * B( 8)** 3 * B(
       . 38) + ( 1./ 6.) * B( 7)** 3 * B( 39)

        C(  4,  1,  5)=
```

Table II RK3

```
. + ( 3./ 24.) * B( 9)** 4 * B( 36)
. + ( 1./ 2.) * B( 9)** 3 * B( 37) + ( 1./ 1.) * B( 8) * B( 24)
. * B( 38) + ( 1./ 1.) * B( 7) * B( 17) * B( 39)
```

ℓ^4

```
C( 4, 1, 6)=
. + ( 1./ 24.) * B( 9)** 4 * B( 36)
. + ( 1./ 6.) * B( 9)** 3 * B( 37) + ( 1./ 1.) * B( 25) * B(
. 38) + ( 1./ 1.) * B( 18) * B( 39)
```

ℓ^4

```
C( 4, 1, 7)=
. + ( 1./ 24.) * B( 9)** 4 * B( 36)
. + ( 1./ 6.) * B( 9)** 3 * B( 37) + ( 1./ 1.) * B( 26) * B(
. 38) + ( 1./ 1.) * B( 19) * B( 39)
```

G.

```
'COMMENT' EQUATIONS WHICH DEFINE THE UNDETERMINED PARAMETERS
USED IN THE EXPANSION OF  E(/   1/).,
```

ℓ^2

```
C( 5, 1, 1)=
. - ( 1./ 1.) * B( 17)
. + ( 1./ 2.) * B( 9)** 2 * B( 33) + ( 1./ 1.) * B( 9) * B( 34)
. + ( 1./ 1.) * B( 8) * B( 35)
```

ℓ^3

```
C( 5, 1, 2)=
. - ( 1./ 1.) * B( 18)
. + ( 1./ 6.) * B( 9)** 3 * B( 33) + ( 1./ 2.) * B( 9)** 2 * B(
. 34) + ( 1./ 2.) * B( 8)** 2 * B( 35)
```

ℓ^3

```
C( 5, 1, 3)=
. - ( 1./ 1.) * B( 19)
. + ( 1./ 6.) * B( 9)** 3 * B( 33) + ( 1./ 2.) * B( 9)** 2 * B(
. 34) + ( 1./ 1.) * B( 24) * B( 35)
```

ℓ^4

```
C( 5, 1, 4)=
. - ( 1./ 1.) * B( 20)
. + ( 1./ 24.) * B( 9)** 4 * B( 33) + ( 1./ 6.) * B( 9)** 3 * B(
. 34) + ( 1./ 6.) * B( 8)** 3 * B( 35)
```

ℓ^4

```
C( 5, 1, 5)=
. - ( 1./ 1.) * B( 21)
. + ( 3./ 24.) * B( 9)** 4 * B( 33) + ( 1./ 2.) * B( 9)** 3 * B(
. 34) + ( 1./ 1.) * B( 8) * B( 24) * B( 35)
```

ℓ^4

```
C( 5, 1, 6)=
. - ( 1./ 1.) * B( 22)
. + ( 1./ 24.) * B( 9)** 4 * B( 33) + ( 1./ 6.) * B( 9)** 3 * B(
. 34) + ( 1./ 1.) * B( 25) * B( 35)
```

ℓ^4

```
C( 5, 1, 7)=
. - ( 1./ 1.) * B( 23)
. + ( 1./ 24.) * B( 9)** 4 * B( 33) + ( 1./ 6.) * B( 9)** 3 * B(
. 34) + ( 1./ 1.) * B( 26) * B( 35)
```

Table II RK3

```
'COMMENT' EQUATIONS WHICH DEFINE THE UNDETERMINED PARAMETERS
USED IN THE EXPANSION OF  E(/   2/).,

ℓ2   C(  6,  1,  1)=
   . - ( 1./ 1.) * B( 24)
   . + ( 1./ 2.) * B( 9)** 2 * B( 31) + ( 1./ 1.) * B( 9) * B( 32)
ℓ3   C(  6,  1,  2)=
   . - ( 1./ 1.) * B( 25)
   . + ( 1./ 6.) * B( 9)** 3 * B( 31) + ( 1./ 2.) * B( 9)** 2 * B(
   . 32)
ℓ3   C(  6,  1,  3)=
   . - ( 1./ 1.) * B( 26)
   . + ( 1./ 6.) * B( 9)** 3 * B( 31) + ( 1./ 2.) * B( 9)** 2 * B(
   . 32)
ℓ4   C(  6,  1,  4)=
   . - ( 1./ 1.) * B( 27)
   . + ( 1./ 24.) * B( 9)** 4 * B( 31) + ( 1./ 6.) * B( 9)** 3 * B(
   . 32)
ℓ4   C(  6,  1,  5)=
   . - ( 1./ 1.) * B( 28)
   . + ( 3./ 24.) * B( 9)** 4 * B( 31) + ( 1./ 2.) * B( 9)** 3 * B(
   . 32)
ℓ4   C(  6,  1,  6)=
   . - ( 1./ 1.) * B( 29)
   . + ( 1./ 24.) * B( 9)** 4 * B( 31) + ( 1./ 6.) * B( 9)** 3 * B(
   . 32)
ℓ4   C(  6,  1,  7)=
   . - ( 1./ 1.) * B( 30)
   . + ( 1./ 24.) * B( 9)** 4 * B( 31) + ( 1./ 6.) * B( 9)** 3 * B(
   . 32)


 DEFINE THE UNDETERMINED PARAMETERS B(/.../) BY PRINTING OUT          H.
THE EXPANSIONS OF   E(/I,K/), I IN  M .= (1,...,E*Q), K IN  P .=
 (0,...,ORDER - 1), WITH RESPECT TO THESE PARAMETERS, 'IF' MODE
= 0 'THEN' THEIR LOCAL ORIGINS ARE THE POINT C 'ELSE'
'IF' MODE = -1 'THEN' THEY ARE - I1 * H WHERE I1.= I 'DIV' Q

 E(/   1,   0/).= U(/   0/) ( B(/   1/)) (/   0/) + SUM(I,0,   6,B(/  17 + I/)*
 A(   0 * H)(/I/)

 E(/   2,   0/).= U(/   0/) ( B(/   2/)) (/   0/) + SUM(I,0,   6,B(/  24 + I/)*
 A(   0 * H)(/I/)
```

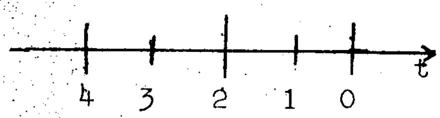
Table II RK3

```
'COMMENT'LAST.=   622  LAST1.=     0  TEMPO-LAST.=  5378  LIST LENGTH-TEMPO.=       I.
  LOCO
NEXT FREE PARAMETER B(/    40/).,

 END OF COMPUTATICNS.
```

Table III. RDV [2] Worksheet for RKM1 Input

Problem $D^2x = X(x)$

Order = 2



Scheme RDV

$N_0 = X(E_3);\ N_1 = X(E_2);\ N_2 = X(E_1);\ N_3 = X(E_0)$

$E_1 = E_2 + N_0 + N_1$

$E_0 = E_2 + N_1 + N_2 + N_3$

rank = 2
period = 1
extent = 2
3 derivatives require 4 harmonics.

RKM1 Input Data

2,2,2,2,1,4,3, — A. Scheme Parameters

data table (3,4,[1,∞],+,2)

comment RDV 2 , second order equation; — B. A Comment

1,true,3, 1,true,2, — C. The Scheme

1,true,1, 1,true,0,

3,1,3, 0,2, 1,0,1,1,

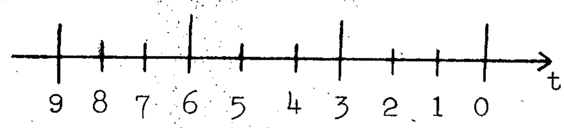
3,0,4, 0,2, 1,1,1,2,1,3

All other data is the same as in Table I.

Table IV. Butcher [1] Worksheet for RKM1

Problem Dx = X(x) Order = 1

Scheme Butcher, k = 3



$N_o = X(E_9)$; $N_1 = X(E_6)$; $N_2 = X(E_3)$; $N_3 = X(E_2)$; $N_4 = X(E_1)$;

$$E_2 = E_3 + E_6 + E_9 + N_o + N_1 + N_2 + N_3$$
$$E_1 = E_3 + E_6 + E_9 + N_o + N_1 + N_2 + N_3$$
$$E_o = E_3 + E_6 + E_9 + N_o + N_1 + N_2 + N_3 + N_4$$

rank = 3
period = 1
extent = 3
3 derivatives require 7 harmonics.

RKM Input data

1,5,3,3,1,7,3, A. Scheme Parameters

data table(3,7,0,+,5)

comment Butcher [1] with k = 3; B. A Comment

1,true,9,	1,true,6,	1,true,3,	C. Scheme
1,true,2,	1, true, 1,		
3,2,6,	0,3,0,6,0,9,	1,0,1,1,1,2,	
3,1,7,	0,3,0,6,0,9,	1,0,1,1,1,2,1,3,	
3,0,8,	0,3,0,6,0,9,	1,0,1,1,1,2,1,3,1,4,	

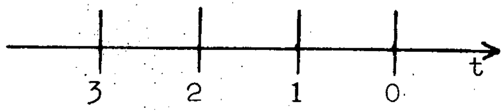
All other data is the same as in Table I.

Table V. Adams [3] Worksheet for RKM1

<u>Problem</u> Dx = X(x)

Order = 1

<u>Scheme</u> Adams



$N_o = X(E_3)$; $N_1 = X(E_2)$; $N_2 = X(E_1)$;

$E_o = E_1 + N_1 + N_2 + N_3$

rank = 1
period = 1
extent = 3
1 derivative requires 1 harmonic

<u>RKM1</u> <u>Input Data</u>

1,3,1,3,1,1,1,			A. Scheme Parameters
data table (1,1,0,+,3)			
<u>comment</u> Illustrative Adams scheme;			B. A Comment
1,<u>true</u>,3,	1,<u>true</u>,2,	1,<u>true</u>,1,	C. Scheme
3,0,4,	0,1,	1,0,1, 1,1,2,	

All other data is the same as in Table I.

REFERENCES

1. J. C. Butcher, A Modified Multistep Method for the Numerical Integration of Ordinary Differential Equations, J. ACM, 12, 1, Jan. 1965, p. 131.

2. R. DeVogelaere, A Method for the Numerical Integration of Differential Equations of Second Order Without Explicit First Derivatives, J. Res. NBS 54, 3, 1955, pp. 119-125.

3. F. Ceschino and J. Kuntzmann, Numerical Solution of Initial Value Problems, Prentice Hall, 1966.

4. E. Close, Generation of Generalized Runge-Kutta Integration Methods for n-th Order Systems of p-th Order Ordinary Differential Equations, UCRL-18650, Lawrence Radiation Laboratory, Univ. of Calif., Berkeley, Calif., Dec. 1968.

5. R. DeVogelaere, Mathematics Department, University of Calif., Berkeley, Calif; private communication.

6. J. C. Butcher, Coefficients for the Study of Runge-Kutta Integration Processes, J. of the Australian Math. Soc., 3, Part 2, May 1963, pp. 185-201.

7. T. Frey, On Improvement of the Runge-Kutta-Nyström Method I, Perodica Polytechnica, Electrical Engineering - Elektrotechnik, Budapest, 2, 2, 1958, pp. 141-165.

TECHNICAL INFORMATION DIVISION
LAWRENCE RADIATION LABORATORY
UNIVERSITY OF CALIFORNIA
BERKELEY, CALIFORNIA 94720