GIER SYSTEM LIBRARY

Order No.: 303

Class: 3.6

Type: ALGOL Procedure

Author: P. Villemoes

Ed.: December 1964

Simultaneous First Order Differential Equations I

RK fifth order x

A/S REGNECENTRALEN

GIER System Library
3.6 Calculus and Differential Equations.
P. Villemoes
A/S Regnecentralen
December 1964

Simultaneous First Order Differential Equations I

ALGOL Procedure

RK fifth order x (f,x,y,b,eps,n,fi)

ABSTRACT

RK fifth order x solves a system of first order ordinary differential equations with given initial values by a fifth order Runge Kutta method using the same integration variable throughout the integration. It uses variable step size in order to achieve a given accuracy by a minimum number of steps. It uses 7 points per step.

1. Function and Parameters.

RK fifth order x solves the system of simultaneous first order ordinary differential equations

$$dy[i]/dx = gi(x,y[1],y[2],...y[n]) \quad , \quad i = 1,2,...,n$$

with given initial values, by a fifth order Runge Kutta method, which uses x as integration variable. It has an automatic control of step size, which tries to minimize the number of steps necessary to obtain a prescribed accuracy. It performs 7 evaluations of the differential equations per step.

Input parameters:

- f — a procedure that defines the differential equations. When called as f(x,y,z), it must assign the derivatives dy[i]/dx to z[i] for i = 1,2,...,n. Thus x must be a real variable and y and z must be arrays with components 1 to n. x and y must not be changed by f.
- b — the end of the integration interval of x.
- n — the number of equations.
- fi — a Boolean which should be set true for a first or isolated call, false at subsequent calls. It tells whether information of the step size is available from a previous call.

Input/output parameters:

- x — the real integration variable. On entry it must contain the beginning of the integration interval. Upon exit it contains the end of the integration interval.
- y — array y[1:n] contains the dependent variables. On entry y must contain the initial values of the dependent variables at the beginning of the integration interval. Upon exit y contains the values of the dependent variables at the end of the integration interval.

eps a real variable having a positive value, which on entry specifies the desired accuracy of the solution. In order to utilize the full capacity of GIER, eps should have the value 10^{-5} on entry. Upon exit eps is in general unchanged, except for those cases, when the specified accuracy could not be obtained, then eps has a value larger than the input value. See section 2.4 and 3.2 for details.

2. Method.

2.1 The fifth order Runge Kutta formulae used are derived by Zonneveld (ref. 1). They use six intermediate points in each interval and one additional point for the step control. Some of the linear combinations, that determine the points to be used, contain negative weight coefficients, but the expression for the final increment has positive coefficients only.

2.2 In order to repair the lack of correct round off of floating point operations in GIER the quasidouble precision method of ref. 3 is utilized for the addition of the final x and y increments to x and the y-s at each step.

2.3 The strategy of the automatic step size control is the following: When a step h is to be taken, the contribution c5 of that term of the corresponding Taylor expansion, which contains h↑5, is calculated. This calculation requires one extra point. Then the largest value for i = 1 to n of the ratio

fh := c5/(h + h×dy[i]/dx)

is calculated. The denominator is approximately a two dimensional distance between the present point and the previous one. fh is proportional to h↑4, and the size of a new step is then determined from an approximation to the formula

h new := h×(eps/fh)↑(1/4)

so that the new step tries to keep fh = eps. If the ratio fh becomes greater than 2, the step h is rejected and another step of size h new is tried.

2.4 If the resulting step size becomes smaller than eps × length of integration interval, corresponding to 1/eps steps through the entire interval, the requirements for accuracy are violated by doubling the value of eps. In this way the procedure itself compensates for too strict demands for accuracy by giving some results, although of a lower accuracy than desired, and then telling, by means of the output value of eps, that it has been in trouble. This mechanism also allows the integration over singularities in some cases.

3. Accuracy, Time and Storage Requirements.

3.1 In the following discussion the absolute error of a result is defined as the Euclidean distance between two points with the same abscissae on the calculated and the exact solution curve. The relative error is then defined as the ratio of the absolute error and the length of the known n-dimensional solution vector composed by the y-s.

3.2 The accuracy is determined by eps. For systems without singularities it is found that the relative error is not greater than eps and normally it is smaller than $_{10}$-2×eps with $_{10}$-2 ≤ eps ≤ $_{10}$-5. For input values of eps of the order of $_{10}$-6 or less it is in general found that the corresponding output values of eps are not less than $_{10}$-6. When the full accuracy of GIER is to be utilized, an input value of $_{10}$-5 is recommended, while values as high as $_{10}$-2 may be used, when lesser accuracy is needed. In these cases, however, a simpler integration procedure than the present might be preferable.

3.3 The time requirements of the procedure depends on the problem to be solved and the choice of eps. As an indication may serve that the solution of two equations runs with a speed of about of 3 integration steps per second.

3.4 The code of the procedure occupies about 450 machine words, and it uses about 30 + 8×n locations in the stack for variables. Typographical length is 115 lines.

4. Test and General Discussion.

4.1 Besides the test results mentioned here is referred to ref. 1. The procedure has been tested on the GIER on a variety of differential equations. The tests showed that

a. The use of the quasi double precision scheme of ref. 3 improved the highest obtainable accuracy by at least one decimal digit.
b. The adopted strategy for calculating a new step assured that the total number of steps was minimum for a number of different equations.
c. The use of a minimum step size permitted the integration into a singularity and assured that a call of the procedure was executed in a reasonable time, also when the required accuracy could not be obtained in the GIER arithmetic.

4.2 For a specific set of two first order equations derived from the differential equation

d2y/dx↑2 + 2/(x-1)×dy/dx + y/(x-1)↑4 = 0

with initial values (x,y,dy/dx) = (0,sin(1), cos(1)), with the exact solution y = sin(1/(1-x)), the procedure has been compared with procedure Mersn, which uses about half the amount of core store, (ref. 2). The comparison showed that RK fifth order x was able to integrate the equations from x = 0 to x = .85 in about 10 percent less time than Mersn while obtaining the same accuracy, and that the smallest obtainable relative errors were 3_{10}-7 and 5_{10}-8 for Mersn and RK fifth order x respectively with both procedures using about 30 seconds in the latter case.

4.3 A drawback of the step size control mechanism is that it may be cheated, as it solely depends on the contribution to the increment of the solution from the fifth order term of the Taylor expansion. When the procedure is called with fi = false, it tries to integrate over the total interval in one step, and this step may erroneously be accepted if the Taylor expansion around the starting point has no fifth order term. The problem is easily remedied, if one is aware of it, by performing the integration by several calls of the procedure in succession.

5. References.

[1] J.A. Zonneveld: Automatic Numerical Integration, Mathematical Centre Tracts 8, Amsterdam (1964).
[2] Peter Naur: procedure Mersn, GIER System Library Order no. 141, A/S Regnecentralen (1962).
[3] Ole Møller: On a Quasi Double-Precision in Floating-Point Addition, BIT Vol 5, No. 1 (1965).

6. Algorithm.

```
procedure RK fifth order x(f,x,y,b,eps,n,fi);
value b,n,fi;
procedure f; array y;  real x,b,eps; integer n; Boolean fi;
begin comment
  RK fifth order  x  solves the system of simultaneous first order
  ordinary differential equations
          dy.i]/dx = gi(x,y[1],y[2],....y[n] , i = 1,2.....n
  with given initial values, by a fifth order Runge Kutta method, which
  uses x  as integration variable. It has an automatic control of step
  size, which tries to minimize the number of steps necessary to obtain
  a prescribed accuracy. It performs 7 evaluations of the differential
  equations per step.

  Input parameters:
     f        a procedure that defines the differential equations. When
              it is called as f(x,y,z) , it must assign the derivatives
              dy[i]/dx  to  z[i]  for i = 1,2....,n. Thus  x  must be
              a real variable and y and z must be arrays with components
              1 to n. x and y must not be changed by f.
     b        the end of the integration interval of x.
     n        the number of equations.
     fi       a Boolean which should be set true for a first or isolated
              call, false at subsequent calls. It tells whether
              information of the step size is available from a previous
              call.

  Input/output parameters:
     x        the real integration variable. On entry it must contain
              the beginnirg of the integration interval  Upon exit it
              contains the end of the integration interval.
     y        array y[1:n] contains the dependent variables. On entry
              y must contain the initial values of the dependent variables
              at the beginning of the integration interval. Upon exit y
              contains the values of the dependent variables at the end
              of the integration interval.
     eps      a real variable having a positive value, which on entry
              specifies the desired accuracy of the solution. In order
              to utilize the full capacity of GIER, eps should have the
              value 10-5 on entry. Upon exit eps is in general unchanged,
              except for those cases, when the specified accuracy could
              not be obtained, then eps has a value larger than the
              input value;
```

```
integer i; real h,xl,int,fh,fhm,mu,d0; own real old h;
array d, yl,k0,k1,k2,k3,k4,k5[1:n];
procedure step(xincr,yincr,k);
real xincr,yincr; array k;
begin
  x:=xincr×h+xl;
  for i:=1 step 1 until n do y[i]:=yl[i]+y incr×h;
  f(x,y,k)
end step;
xl:=x; int:=b-xl;
if fi then old h:=int;
for i:=1 step 1 until n do
begin
  yl[i]:=y[i]; d[i]:=0
end;
d0:=0; h:=abs(old h);
if int<0 then
begin
  int:=-int;
  h:=-h
end;
int:=eps×int;
fi:=true;
for mu:=abs(h) while fi do
begin comment integration loop;
  if mu<int then
  begin
    eps:=2×eps;
    int:=2×int;
    h:=(if h<0 then -int else int)×5
  end adjustment af eps;
  if (h-(b-xl))×h>0 then
  begin
    old h:=h;
    h:=b-xl;
    fi:=false
  end set last step;
  f(xl,yl,k0);
  step(0.2222222222,k0[i]×2/9,k1);
  step(0.3333333333,(k0[i]+k1[i]×3)/12,k2);
  step(0.5,(k0[i]+k2[i]×3)/8,k3);
  step(0.8,(53×k0[i]-135×k1[i]+126×k2[i]+56×k3[i])/125,k4);
  step(1,(133×k0[i]-378×k1[i]+276×k2[i]+112×k3[i]+25×k4[i])/168,k5);
  fhm:=0;
  for i:=1 step 1 until n do
  begin
    fh:=abs(21×k0[i]-162×k2[i]+224×k3[i]-125×k4[i]
      +42×k5[i])/14/(abs(k0[i])+1)/eps;
    if fh>fhm then fhm:=fh
  end;
  mu:=1/(1+fhm)+0.5;
  if fhm<2 then
  begin comment step accepted;
    step(1,(-63×k0[i]+189×k1[i]-36×k2[i]-112×k3[i]
      +50×k4[i])/28,k5);
```

```
        for i:=1 step 1 until n do
        begin comment final increment;
          fh:=(35×k0[i]+162×k2[i]+125×k4[i]+14×k5[i])/336×h+d[i];
          fhm:=y[i]:=y1[i]+fh;
          d[i]:=fh-(fhm-y1[i])
        end;
        fh:=h+d0; x:=x1+fh; d0:=fh-(x-x1); x1:=x;
        for i:=1 step 1 until n do y1[i]:=y[i]
      end step accepted
      else fi:=true;
      h:=mu×h
    end integration loop
end RK fifth order x;
```