ZASTOSOWANIA MATEMATYKI
APPLICATIONES MATHEMATICAE
XIX, 1 (1987), pp. 157–168

ALGORITHMS 91-92

M. SZYSZKOWICZ (Wrocław)

TWO ONE-STEP METHODS WITH A GIVEN PARAMETER

1. Procedure declaration. Two procedures (*sode123*, *sode567*) in ALGOL 60 for solving the system of ordinary differential equations

$$(1.1)\qquad y' = f(x, y), \quad y(x_0) \text{ given},$$

are presented.

The procedures have the same parameters.

Data:

x – value of x_0 in (1.1),
x1 – value of the argument for which the problem (1.1) is solved,
eps – relative error (the given tolerance),
eta – number which is used instead of zero obtained as solution,
hmin – minimum allowed step-size,
n – number of differential equations,
y[1:n] – vector with initial data $y(x_0)$ in (1.1),
sigma – for the procedure *sode123*: $sigma = \frac{1}{7}$ or $\frac{1}{3}$, for the procedure *sode567*: $sigma = \frac{1}{64}$ or $\frac{1}{42}$.

Results:

x – value of *x1*,
y[1:n] – vector with the solution at point *x1*.

Additional parameters:

steph – label outside of the body of the procedures (*sode123*, *sode567*) to which a jump is made if $|h| < hmin$ (h is the step-size of integration); increasing *eps* or decreasing *hmin* it is possible to continue the computations,
f – procedure with the heading: **procedure** $f(x, n, y, d)$; **value** x, n; **real** x; **integer** n; **array** y, d; which computes the values of the functions $f(x, y)$ in (1.1) and assigns them to $d[1:n]$.

```
procedure sode123(x,x1,eps,eta,hmin,n,y,steph,f,sigma);
 value x1,eps,eta,hmin,n,sigma;
 real x,x1,eps,eta,hmin,sigma;
 integer n;
 array y;
 label steph;
 procedure f;
 begin
  real h,hh,ww,w1,w2,w3;
  integer i;
  Boolean last;
  array d1,y1,y2,y3,y4[1:n];
  eps:=.5/eps;
  h:=x1-x;
  last:=true;
  f(x,n,y,d1);
conth:
  hh:=.5×h;
  w3:=h×sigma;
  ww:=hh×sigma;
  for i:=1 step 1 until n do
   begin
    w1:=y[i];
    w2:=d1[i];
    y1[i]:=w1+w3×w2;
    y2[i]:=w1+ww×w2
   end i;
```

```
f(x+w3,n,y1,y3);
for i:=1 step 1 until n do
 y1[i]:=y[i]+h×y3[i];
f(x+ww,n,y2,y3);
for i:=1 step 1 until n do
 y2[i]:=y[i]+hh×y3[i];
f(x+hh,n,y2,y3);
for i:=1 step 1 until n do
 y4[i]:=y2[i]+ww×y3[i];
f(x+hh+ww,n,y4,y3);
ww:=.0;
for i:=1 step 1 until n do
 begin
  w2:=y2[i]+hh×y3[i];
  w3:=w2-y1[i];
  w1:=y3[i]:=w2+w3;
  w3:=abs(w3);
  w1:=abs(w1);
  if w1<eta
   then w1:=eta;
  w1:=w3/w1;
  if w1>ww
   then ww:=w1
 end i;
ww:=if ww=0 then eta else sqrt(eps×ww)×1.25;
hh:=h/ww;
if ww>1.25
 then
 begin
```

```
      if abs(hh)<hmin
       then go to steph;
      last:=false
     end ww>1.25
     else
     begin
      x:=x+h;
      for i:=1 step 1 until n do
       y[i]:=y3[i];
      if last
       then go to endp;
      f(x,n,y,d1);
      w1:=x1-x;
      if (w1-hh)×sign(h)<0
       then
       begin
        hh:=w1;
        last:=true
       end (w1-hh)×h<0
    end ww<1.25;
   h:=hh;
   go to conth;
 endp:
 end sode123;
```

```
procedure sode567(x,x1,eps,eta,hmin,n,y,steph,f,sigma);
 value x1,eps,eta,hmin,n,sigma;
 real x,x1,eps,eta,hmin,sigma;
 integer n;
 array y;
 label steph;
 procedure f;
 begin
  real h,hh,ww,w1,w2,w3;
  integer i;
  Boolean last;
  array k,k1,k2,k3,k4,yh,y1,y2,y3[1:n];
  procedure steprk5(h,x,k1,y,df);
   value h,x;
   real h,x;
   array k1,y,df;
   begin
    w1:=.5×h;
    for i:=1 step 1 until n do
     yh[i]:=y[i]+w1×k1[i];
    f(x+.5×h,n,yh,k2);
    w1:=.0625×h;
    for i:=1 step 1 until n do
     yh[i]:=y[i]+w1×(3.0×k1[i]+k2[i]);
    f(x+.25×h,n,yh,k3);
    w1:=.25-16.0×sigma;
    w2:=32.0×sigma;
```

```
    for i:=1 step 1 until n do
     yh[i]:=y[i]+h×(w1×(k1[i]+k2[i])+w2×k3[i]);
    f(x+.5×h,n,yh,k4);
    w1:=12.0×sigma-.1875;
    w2:=12.0×sigma-.375;
    w3:=.75-24.0×sigma;
    for i:=1 step 1 until n do
     yh[i]:=y[i]+h×(w1×k1[i]+w2×k2[i]+w3×k3[i]+.5625×k4[i]);
    f(x+.75×h,n,yh,df);
    w1:=h/7.0;
    w2:=4.0-192.0×sigma;
    w3:=7.0-192.0×sigma;
    ww:=384.0×sigma;
    for i:=1 step 1 until n do
     yh[i]:=y[i]+w1×(w2×k1[i]+w3×k2[i]+ww×k3[i]-12.0×k4[i]+8.0×df[i]);
    f(x+h,n,yh,k2);
    w1:=h/90.0;
    for i:=1 step 1 until n do
     df[i]:=y[i]+w1×(7.0×(k1[i]+k2[i])+32.0×(k3[i]+df[i])+12.0×k4[i])
   end steprk5;
  h:=x1-x;
  eps:=1.0/(eps×62.0);
  last:=true;
  f(x,n,y,k1);
conth:
  hh:=.5×h;
  steprk5(h,x,k1,y,y1);
  steprk5(hh,x,k1,y,y2);
  f(x+hh,n,y2,k);
```

```
steprk5(hh,x+hh,k,y2,y3);
ww:=.0;
for i:=1 step 1 until n do
 begin
  w3:=y3[i];
  w1:=w3-y1[i];
  w3:=y3[i]:=w3+w1/31.0;
  w1:=abs(w1);
  w3:=abs(w3);
  if w3<eta
   then w3:=eta;
  w1:=w1/w3;
  if w1>ww
   then ww:=w1
 end i;
ww:=if ww=0 then eta else (eps×ww)↑.166666666666×1.25;
hh:=h/ww;
if ww>1.25
 then
 begin
  if abs(hh)<hmin
   then go to steph;
  last:=false
 end ww>1.25
 else
 begin
  x:=x+h;
  for i:=1 step 1 until n do
   y[i]:=y3[i];
```

```
    if last
     then go to endp;
    f(x,n,y,k1);
    w1:=x1-x;
    if (w1-hh)×sign(h)<0
     then
     begin
      hh:=w1;
      last:=true
     end (w1-hh)×h<0
   end ww<1.25;
  h:=hh;
  go to conth;
endp:
 end sode567;
```

2. Method used. To solve the initial value problem

$$y' = f(x, y), \qquad y(x_0) \text{ given},$$

we consider the one-step method Φ which satisfies the following property: When applied to the test equation

$$y' = \lambda y, \qquad y(x_0) = y_0, \qquad \lambda \in C$$

with a constant step h the m-stage method Φ yields a numerical solution $\{y_n\}$ which satisfies a recurrence relation of the form

$$y_{n+1} = w(z)\, y_n,$$

where $z = h\lambda$ and

$$w(z) = 1 + z + \ldots + \frac{z^p}{p!} + \sum_{i=p+1}^{m} a_i \frac{z^i}{i!}. \tag{2.1}$$

The coefficients a_i $(i = p+1, p+2, \ldots, m)$ are depending on the parameters of the method Φ. Moreover, if the method Φ has an error of order p, then one has

$$w(z) - e^z = O(z^{p+1}).$$

In this paper we propose the choice of the coefficients a_i ($i = p+1, p+2, \ldots, m$) in (2.1) to be

$$w^*(z) - e^z = O(z^{m+2}), \tag{2.2}$$

where

$$w^*(z) = w^2(z/2) + \frac{w^2(z/2) - w(z)}{2^p - 1}$$

and

$$y^*_{n+1} = w^*(z)\, y_n.$$

The new numerical solution $\{y^*_n\}$ ($y_k := y^*_k$, $1 \leqslant k \leqslant n$) is obtained by using Richardson's extrapolation applied to the solutions obtained with step h and step $h/2$.

Here we present two one-step methods Φ_1 and Φ_2 with $m > p$. The method Φ_1 has $p = 1$, $m = 2$, the method Φ_2 has $p = 5$, $m = 6$. The method Φ_1 is given by the following formulae

$$\Phi_1: \quad \begin{aligned} y_{n+a} &= y_n + ahf_n, \\ y_{n+1} &= y_n + hf_{n+a}, \end{aligned}$$

where a is the parameter ([2]).

For the method Φ_1 the polynomial (2.1) has the form

$$w(z) = 1 + z + az^2.$$

In this paper the method Φ_1 is realized with two values of the parameter a, $a = \frac{1}{7}$ and $a = \frac{1}{3}$. For $a = \frac{1}{7}$ the method Φ_1 has the interval of absolute stability $(-7.0, 0)$ and there holds

$$w^*(z) - e^z = O(z^3).$$

For $a = \frac{1}{3}$ the method Φ_1 has the interval of absolute stability $(-3.0, 0)$ but there holds (2.2), i.e.

$$w^*(z) - e^z = O(z^4).$$

The method Φ_1 in this paper is realized in the procedure *sode123* with the parameter *sigma* (a = *sigma*).

The method Φ_2 was given by Lawson [1] and has the following form

$$\begin{array}{c|cccccc}
\frac{1}{2} & \frac{1}{2} \\
\frac{1}{4} & \frac{3}{16} & \frac{1}{16} \\
\frac{1}{2} & \frac{1}{4}-16\sigma & \frac{1}{4}-16\sigma & 32\sigma \\
\frac{3}{4} & \frac{-3}{16}+12\sigma & \frac{-6}{16}+12\sigma & \frac{3}{4}-24\sigma & \frac{9}{16} \\
1 & \frac{4}{7}-\frac{192\sigma}{7} & 1-\frac{192\sigma}{7} & \frac{384\sigma}{7} & -\frac{12}{7} & \frac{8}{7} \\
\hline
 & \frac{7}{90} & 0 & \frac{32}{90} & \frac{12}{90} & \frac{32}{90} & \frac{7}{90}
\end{array}$$

For the method Φ_2 the polynomial (2.2) has the form

$$w(z) = \sum_{i=0}^{5} \frac{z^i}{i!} + 36\sigma \frac{z^6}{6!}.$$

Lawson [1] has used this method with $\sigma = \frac{1}{64}$, with this value of the parameter σ the method Φ_2 has the interval of absolute stability $(-5.62, 0)$. For $\sigma = \frac{1}{42}$ ([2]) there holds (2.2), i.e.

$$w^*(z) - e^z = O(z^8)$$

and the interval of absolute stability is $(-4.25, 0)$. The method Φ_2 is realized in the procedure *sode567* with the parameter *sigma* (σ = *sigma*).

3. Certification. The procedures *sode123* and *sode567* were tested on the following problems.

Problem A

$$y_1' = \quad 10\,\mathrm{sgn}\sin(20x)\,y_2, \qquad y_1(0) = 0,$$
$$y_2' = -10\,\mathrm{sgn}\sin(20x)\,y_1, \qquad y_2(0) = 1$$

with the exact solution

$$y_1(x) = |\sin 10x|, \qquad y_2(x) = |\cos 10x|.$$

Problem B

$$y_1' = 1/y_2; \qquad y_1(0) = 1,$$
$$y_2' = -1/y_1, \quad y_2(0) = 1$$

with the exact solution

$$y_1(x) = e^x, \qquad y_2(x) = e^{-x}.$$

Problem C

$$y' = \lambda y, \qquad y(0) = 1$$

with the exact solution

$$y(x) = e^{\lambda x}.$$

Below the relative errors $(y_n - y(x_n))/y(x_n)$ and the numbers of function evaluations f ([f]) are given. The results presented here were obtained by the procedures *sode123*, *sode567* with automatic step size control and with *sigma* as the parameter in the procedures.

For problem A

sode123

x	*sigma* $= \frac{1}{7}$ *eps* $= {}_{10}-4$	[f]	*sigma* $= \frac{1}{3}$ *eps* $= {}_{10}-4$	[f]
1.0	$-6.66_{10}-4$ $-1.46_{10}-4$	3346	$-7.64_{10}-4$ $-4.13_{10}-4$	3978

sode567

x	$sigma = \frac{1}{64}$ $eps = {}_{10}{-4}$	$[f]$	$sigma = \frac{1}{42}$ $eps = {}_{10}{-4}$	$[f]$
1.0	$5.70_{10}-5$	8756	$-2.86_{10}-5$	9020
	$2.64_{10}-5$		$-2.21_{10}-5$	

For problem B

sode123

x	$sigma = \frac{1}{7}$ $eps = {}_{10}{-6}$	$[f]$	$sigma = \frac{1}{3}$ $eps = {}_{10}{-6}$	$[f]$
.5	$2.61_{10}-7$	939	$-2.12_{10}-7$	644
	$-2.61_{10}-7$		$-2.11_{10}-7$	
10.0	$4.95_{10}-6$	17763	$3.97_{10}-6$	12143
	$4.95_{10}-6$		$-3.96_{10}-6$	

sode567

x	$sigma = \frac{1}{36}$ $eps = {}_{10}{-3}$	$[f]$	$sigma = \frac{1}{42}$ $eps = {}_{10}{-3}$	$[f]$
10.0	$-3.33_{10}-3$	216	$-1.39_{10}-2$	198
	$4.06_{10}-3$		$1.83_{10}-2$	

x	$sigma = \frac{1}{64}$ $eps = {}_{10}{-3}$	$[f]$	$sigma = 0$ $eps = {}_{10}{-3}$	$[f]$
10.0	$9.32_{10}-3$	234	$1.85_{10}-2$	252
	$-1.22_{10}-2$		$-2.28_{10}-2$	

For problem C

sode567

x	$sigma = \frac{1}{64}$ $eps = {}_{10}{-9}$	$[f]$	$sigma = \frac{1}{42}$ $eps = {}_{10}{-9}$	$[f]$
-6	$1.10_{10}-9$	628	$-2.98_{10}-10$	509
-1	$1.38_{10}-10$	118	$-1.97_{10}-11$	101
1	$-1.92_{10}-10$	118	$-4.28_{10}-11$	101
6	$-1.20_{10}-9$	610	$-2.40_{10}-10$	525

References

[1] J. D. Lawson, *An order five Runge–Kutta process with extended region of stability*, SIAM J. Numer. Anal. 3 (1966), p. 593-597.

[2] M. Szyszkowicz, *Metody jednokrokowe o podwyższonym rzędzie dokładności*, Report N-118, Instytut Informatyki Uniwersytetu Wrocławskiego, Wrocław 1983.

INSTITUTE OF COMPUTER SCIENCE
UNIVERSITY OF WROCŁAW
51-151 WROCŁAW

Received on 1983.05.20