ZASTOSOWANIA MATEMATYKI
APPLICATIONES MATHEMATICAE
XIII, 2 (1972)

TERESA POKORA and Ś. ZĄBEK (Lublin)

ON AN EULER-LIKE METHOD WITH EXPONENTIAL CORRECTION FOR INITIAL-VALUE PROBLEMS

There exist a few one-step methods for solving the initial-value problem

$$\frac{dx}{dt} = f(t, x), \quad x(t_0) = x_0, \tag{1}$$

using partial derivatives of $f(t, x)$. The most popular one is the power series method. In 1952 Zurmühl [3] has proposed to formulate a class of methods similar to the well-known Runge-Kutta processes. Several formulas of this type have been obtained by Hobot [1].

The simplest method is expressed by the second order formula

$$x_{n+1} = x_n + hf(t_n, x_n) + \frac{h^2}{2} g(t_n, x_n), \quad n = 0, 1, 2, \ldots, \tag{2}$$

where

$$g(t, x) = \frac{df(t, x)}{dt} = \frac{\partial}{\partial t} f(t, x) + f(t, x) \frac{\partial}{\partial x} f(t, x).$$

If $x_n = x(t_n)$, then $x_{n+1} = x(t_{n+1}) + O(h^3)$, $t_{n+1} = t_n + h$. This formula, obtained as the first three terms of the Taylor series for $x(t)$, is analogous to the well-known Euler algorithm for the problem (1). The present paper deals with a modification of this formula, obtained by taking into account some terms of the infinite remainder of the Taylor series; namely, the terms depending upon f, $\partial f/\partial t$, $\partial f/\partial x$, only.

Let us denote

$$x^{(j)}(t) = \frac{d^j x(t)}{dt^j}.$$

For a solution $x(t)$ of equation (1), we have evidently

$$x'(t) = f(t,x),$$

$$x''(t) = \frac{\partial}{\partial t}f(t,x)+f(t,x)\frac{\partial}{\partial x}f(t,x),$$

$$x'''(t) = \frac{\partial^2}{\partial t^2}f(t,x)+\left[\frac{\partial}{\partial t}f(t,x)+f(t,x)\frac{\partial}{\partial x}f(t,x)\right]\frac{\partial}{\partial x}f(t,x)+$$

$$+f(t,x)\left[2\frac{\partial^2}{\partial t\partial x}f(t,x)+f(t,x)\frac{\partial^2}{\partial x^2}f(t,x)\right]$$

$$= x''(t)\frac{\partial}{\partial x}f(t,x)+r_3(t,x),$$

where

$$r_3(t,x) = \frac{\partial^2}{\partial t^2}f(t,x)+f(t,x)\left[2\frac{\partial^2}{\partial t\partial x}f(t,x)+f(t,x)\frac{\partial^2}{\partial x^2}f(t,x)\right]$$

contains terms with at least one factor being the derivative of the function f of order greater than 1.

We may infer from this by induction for $j = 4, 5, 6, \ldots$ that

$$x^{(j)}(t) = x''(t)\left[\frac{\partial}{\partial x}f(t,x)\right]^{j-2}+r_j(t,x),$$

where

$$r_j(t,x) = \frac{d}{dt}r_{j-1}(t,x)+x''(t)\frac{d}{dt}\left[\frac{\partial}{\partial x}f(t,x)\right]^{j-3}+\left[\frac{\partial}{\partial x}f(t,x)\right]^{j-3}r_3(t,x).$$

Thus

$$x(t+h) = x(t)+\sum_{j=1}^{\infty}x^{(j)}(t)h^j/j!$$

$$= x(t)+hf(t,x)+x''(t)\sum_{j=2}^{\infty}\left[\frac{\partial}{\partial x}f(t,x)\right]^{j-2}h^j/j!+$$

$$+\sum_{j=3}^{\infty}r_j(t,x)h^j/j!$$

$$= x(t)+hf(t,x)+x''(t)\left[\frac{\partial}{\partial x}f(t,x)\right]^{-2}\times$$

$$\times\left\{\exp\left[h\frac{\partial}{\partial x}f(t,x)\right]-1-h\frac{\partial}{\partial x}f(t,x)\right\}+O(h^3).$$

That is, we obtain a new second order formula

$$(3)\qquad x_{n+1} = x_n + hf(t_n, x_n) + g(t_n, x_n)[k(t_n, x_n)]^{-2}\{\exp[hk(t_n, x_n)] - 1\} - \\ - hg(t_n, x_n)[k(t_n, x_n)]^{-1},$$

where

$$k(t, x) = \frac{\partial}{\partial x} f(t, x), \qquad g(t, x) = x''(t) = \frac{\partial}{\partial t} f(t, x) + f(t, x) k(t, x).$$

If $x_n = x(t_n)$, then $x_{n+1} = x(t_{n+1}) + O(h^3)$, $t_{n+1} = t_n + h$.

It is remarkable that this formula is exact for all linear differential equations with constant coefficients, because all the $r_j(t, x) = 0$ identically for $j = 3, 4, \ldots$ in this case. For many other equations (namely, those for which solutions are not comparable with a second degree polynomial) it is to be hoped that the remainder of formula (3) will be less than that for (2).

If we solve the initial problem with the aid of an automatic computer, the evaluation of the exponential function introduces no significant complication, otherwise we exploit only the same elements as in (2), that is, $f, \partial f/\partial t$ and $\partial f/\partial x$ in the point (t_n, x_n).

One of the present authors [2] has used the above-mentioned idea of the "exponential correction" to obtain formulae analogous to the Runge-Kutta fourth order method.

Because for a very small h, $\exp(hk_n)$ may be too close to 1, to avoid the increase of the propagated relative error it is recommended (¹) to use fomula (3) as the following algorithm:

$$\begin{aligned} a &= hf(t_n, x_n), \\ b &= [k(t_n, x_n)]^{-1}, \\ c &= bg(t_n, x_n), \\ d &= \exp[hk(t_n, x_n)], \\ x_{n+1} &= x_n + a + bcd - (b + h)c. \end{aligned}$$

This algorithm was tested on the Odra 1013 computer with 31 bit floating-point mantissa, in the FALA-69 autocode. The obtained results

(¹) If it is possible to calculate the function $p(u) = (\exp(u) - 1)/u - 1$ instead of the Algol-60 standard function $\exp(u)$, it is of advantage to use formula (3) in the form

$$x_{n+1} = x_n + h[f(t_n, x_n) + g(t_n, x_n) p(hk(t_n, x_n))/k(t_n, x_n)].$$

are compared below with analogous results for the Runge-Kutta method of second order,

$$x_{n+1} = x_n + hf[t_n + \tfrac{2}{3}h,\ x_n + \tfrac{2}{3}hf(t_n,\ x_n)]. \tag{4}$$

All these calculations were executed without subdivision of the step h.

Examples.

I. $dx/dt = x+t+1$, $x(0) = 1$. Exact solution: $x = 3e^t - t - 2$. $k(t, x) = 1$, $g(t, x) = x+t+2$, $h = 0.1$.

t	$x(t)$ from (3)	error	$x(t)$ from (4)	error
0.1	1.215512751	$-2_{10}-9$	1.214999998	$-5.1_{10}-4$
0.2	1.464208270	$-2_{10}-9$	1.463074997	$-1.1_{10}-3$
0.5	2.446163782	$-3.2_{10}-8$	2.442340290	$-3.8_{10}-3$
0.8	3.876622712	$-7.3_{10}-8$	3.868366757	$-8.2_{10}-3$
1.0	5.154845375	$-1.2_{10}-7$	5.142242509	$-1.3_{10}-2$

II. $dx/dt = t^3 - 2tx$, $x(1) = 1$. Exact solution: $x = e^{-t^2+1} + (t^2-1)/2$. $k(t, x) = -2t$, $g(t, x) = 3t^2 - 2x - 2t(t^3 - 2tx)$, $h = 0.1$.

t	$x(t)$ from (3)	error	$x(t)$ from (4)	error
1.1	0.914048065	$-1.5_{10}-3$	0.916688887	$+1.1_{10}-3$
1.2	0.861400501	$-2.6_{10}-3$	0.866222679	$+2.2_{10}-3$
1.5	0.907682460	$-3.8_{10}-3$	0.916036444	$+4.5_{10}-3$
1.8	1.223153646	$-3.3_{10}-3$	1.231418826	$+4.9_{10}-3$
2.0	1.547011221	$-2.7_{10}-3$	1.554272520	$+4.4_{10}-3$

III. $dx/dt = (x - t^2)/t$, $x(1) = 1$. Exact solution: $x = 2t - t^2$. $k(t, x) = t^{-1}$, $g(t, x) = 2$, $h = 0.05$.

t	$x(t)$ from (3)	error	$x(t)$ from (4)	error
1.05	0.997457806	$-4.2_{10}-5$	0.997540323	$+4.0_{10}-5$
1.10	0.989915635	$-8.4_{10}-5$	0.990080705	$+8.0_{10}-5$
1.15	0.977373488	$-1.3_{10}-4$	0.977621138	$+1.2_{10}-4$
1.20	0.959831361	$-1.7_{10}-4$	0.960161616	$+1.6_{10}-4$
1.25	0.937289249	$-2.1_{10}-4$	0.937702134	$+2.0_{10}-4$

IV. $dx/dt = t + (x + x^2)/t$, $x(1) = 1$. Exact solution: $x = t\tan(t - 1 + \pi/4)$. $k(t, x) = (1+2x)/t$, $g(t, x) = 2 + 2x + (2x^2 + 2x^3)/t^2$, $h = 0.1$.

t	$x(t)$ from (3)	error	$x(t)$ from (4)	error
1.1	1.344318942	$-1.0_{10}-3$	1.340624996	$-4.7_{10}-3$
1.2	1.806397567	$-3.7_{10}-3$	1.795486788	$-1.9_{10}-2$
1.3	2.453476613	$-1.1_{10}-2$	2.427419336	$-3.7_{10}-2$
1.4	3.419628856	$-3.1_{10}-2$	3.358380557	$-9.2_{10}-2$
1.5	5.013549204	$-9.8_{10}-2$	4.857059981	$-2.5_{10}-1$

For (3) the computation time was always shorter than (or, in the case of example I, equal to) that for (4), because in (4) we must calculate $f(t, x)$ twice, whereas in (3) we may use some common elements during the calculation of the functions f, g and k.

In examples II (especially for t close to 1) and III, the exact solution is comparable with a second degree polynomial, and formula (3) is not distinctly better than (4); both they are even worse than (2).

References

[1] G. Hobot, *Pewne metody rozwiązywania równań różniczkowych zwyczajnych* (*Certain methods of solving ordinary differential equations*), Doctoral dissertation, University of Lublin 1971 (to be published).

[2] T. Pokora, *Runge-Kutta-like method with exponential correction* (in preparation).

[3] R. Zurmühl, *Runge-Kutta Verfahren unter Verwendung höherer Ableitungen*, Z. angew. Math. Mech. 32 (1952), p. 153-154.

DEPARTMENT OF NUMERICAL METHODS
M. CURIE-SKŁODOWSKA UNIVERSITY, LUBLIN

Received on 9. 7. 1971

TERESA POKORA i Ś. ZĄBEK (Lublin)

O ANALOGONIE METODY EULERA Z POPRAWKĄ WYKŁADNICZĄ DLA ZADAŃ POCZĄTKOWYCH

STRESZCZENIE

W pracy podana jest metoda krokowa (3) rozwiązywania zadań początkowych typu (1) przy założeniu możliwości obliczania pierwszych pochodnych cząstkowych prawej strony równania różniczkowego. Metoda ta jest dokładna dla równań liniowych o stałych współczynnikach. Załączono przykładowe wyniki obliczeń, porównane z wynikami otrzymanymi metodą Rungego-Kutty rzędu II, określoną wzorem (4).
