Centrum voor Wiskunde en Informatica
Centre for Mathematics and Computer Science

W. Couzy

Performance evaluation of Rosser's method

Department of Numerical Mathematics Report NM-N9001 January

The Centre for Mathematics and Computer Science is a research institute of the Stichting Mathematisch Centrum, which was founded on February 11, 1946, as a nonprofit institution aiming at the promotion of mathematics, computer science, and their applications. It is sponsored by the Dutch Government through the Netherlands Organization for the Advancement of Research (N.W.O.).

Performance Evaluation of Rosser's Method

W. Couzy

Centre for Mathematics and Computer Science
Post Box 4079, 1009 AB Amsterdam, The Netherlands

In this paper we investigate a block method presented by Rosser [1], which is in fact a six-stage, fourth-order Runge-Kutta method. We compare this method with the classical fourth-order Runge-Kutta method. Numerical experiments are included.

1980 Mathematics Subject Classification: 65M10, 65M20
1982 CR Categories: 5.17
Key Words and Phrases: Block methods, Runge-Kutta methods.

1. Introduction.

Consider the initial-value problem

(1.1) $$y'(t) = f(y(t)), \qquad y(t_0) = y_0 .$$

Lambert [2] analyzed a block method which was proposed by Rosser [1]. The method generates approximations y_{n+1} and y_{n+2} to $y(t_{n+1})$ and $y(t_{n+2})$ respectively, according to

(1.2) $$\begin{aligned} u_{n+1} &= y_n + \tfrac{1}{2}h(\ f(u_{n+1}) + f(y_n)\ ), \\ u_{n+2} &= y_n + 2hf(u_{n+1})\ , \\ y_{n+1} &= y_n + \tfrac{1}{12}h(\ -f(u_{n+2}) + 8f(u_{n+1}) + 5f(y_n\ )\ ), \\ y_{n+2} &= y_n + \tfrac{1}{3}h(\ f(u_{n+2}) + 4f(u_{n+1}) + f(y_n)\ ). \end{aligned}$$

If u_{n+1} is approximated by one correction step started with forward Euler as predictor and if the stepsize is halved, then (1.2) can be written as the fourth-order, six-stage Runge-Kutta method

(1.3)
$$k_1 = f(y_n),$$
$$k_2 = f(y_n + \tfrac{1}{2}hk_1),$$
$$k_3 = f(y_n + \tfrac{1}{4}h(k_1 + k_2)),$$
$$k_4 = f(y_n + hk_3),$$
$$k_5 = f(y_n + \tfrac{1}{24}h(5k_1 + 8k_3 - k_4)),$$
$$k_6 = f(y_n + \tfrac{1}{6}h(k_1 + k_4 + 4k_5)),$$

$$y_{n+1} = y_n + \tfrac{1}{6}h(k_1 + 4k_5 + k_6).$$

We will call this method RRK6.

2. Modified Rosser method.

Lambert observed, that k_6 in formula (1.3) is a third-order approximation to $f(y_{n+1})$. Therefore, for n>0 the function evaluation made in order to compute k_1 can be replaced by k_6 from the preceding step and (1.3) is transformed to a two-step method, still of order four, which we will call RRK5. Lambert compared RRK5 with the classical fourth-order, four-stage Runge-Kutta method (CRK), which reads

(2.1)
$$k_1 = f(y_n),$$
$$k_2 = f(y_n + \tfrac{1}{2}hk_1),$$
$$k_3 = f(y_n + \tfrac{1}{2}hk_2),$$
$$k_4 = f(y_n + hk_3),$$

$$y_{n+1} = y_n + \tfrac{1}{6}h(k_1 + 2k_2 + 2k_3 + k_4).$$

and stated

(2.2) "The two-step block method is considerably more economical than a conventional fourth-order Runge-Kutta method."

In the next paragraph we will investigate the correctness of (2.2).

3. Integration of a test equation.

In order to compare RRK5, RRK6 and CRK theoretically, these methods will be applied to the test equation

$$(3.1) \qquad y'(t) = \lambda y(t), \qquad y(0) = 1, \qquad 0 \le t \le 1.$$

and the analytical numerical solutions will be compared with the true solution $y(t) = e^{\lambda t}$. It is well known that using CRK to integrate (3.1) yields

$$(3.2a) \qquad y_{n+1} = R_{CRK}(z)\, y_n, \qquad z := \lambda h,$$

$$R_{CRK}(z) = 1 + z + \frac{1}{2}z^2 + \frac{1}{6}z^3 + \frac{1}{24}z^4 .$$

Using RRK6 to integrate (3.1) yields

$$(3.2b) \qquad y_{n+1} = R_{RRK6}(z)\, y_n, \qquad z := \lambda h,$$

$$R_{RRK6}(z) = 1 + z + \frac{1}{2}z^2 + \frac{1}{6}z^3 + \frac{1}{24}z^4 + \frac{1}{432}z^5 - \frac{1}{1728}z^6.$$

Using RRK5 to integrate (3.1) is somewhat more complex. We will denote k_6 for the approximation to $f(y_n)$, made by calculating y_n ($n>1$). We find

$$(3.2c) \qquad y_{n+1} = y_n + \left(\frac{1}{6}k_6 + \frac{5}{6}\lambda y_n\right)h + \left(\frac{1}{6}k_6 + \frac{1}{3}\lambda y_n\right)h^2\lambda + \left(\frac{17}{216}k_6 + \frac{19}{216}\lambda y_n\right)h^3\lambda^2$$
$$+ \left(\frac{1}{27}k_6 + \frac{1}{216}\lambda y_n\right)h^4\lambda^3 + \left(\frac{1}{288}k_6 - \frac{1}{864}\lambda y_n\right)h^5\lambda^4 + \frac{1}{1728}h^6\lambda^5 k_6,$$

$$k_6 = \left(1 + z + \frac{1}{2}z^2 + \frac{1}{6}z^3 + \frac{5}{144}z^4 - \frac{1}{288}z^5\right)\lambda y_{n-1}, \qquad z := \lambda h.$$

By assuming that $y_{n-1}=y(t_{n-1})$ and $y_n=y(t_n)$ and by using that $y_{n-1} = y_n + O(z)$, we find

$$(3.3) \qquad f(y_n) = \lambda e^{nz} = \lambda e^{(n-1)z+z} = \lambda e^{z} \cdot e^{(n-1)z} = e^{z}\lambda y_{n-1}.$$

and

$$(3.4) \qquad k_6 = f(y_n) + \left(\left(\frac{5}{144} - \frac{1}{24}\right)z^4 + O(z^5)\right)\lambda y_{n-1} = \left(1 - \frac{1}{144}z^4 + O(z^5)\right)\lambda y_n .$$

Now (3.2c) can be rewritten as

(3.2c') $y_{n+1} = R_{RRK5}(z)y_n, \qquad z := \lambda h,$

$$R_{RRK5}(z) = 1 + z + \frac{1}{2}z^2 + \frac{1}{6}z^3 + \frac{1}{24}z^4 + \frac{1}{864}z^5 + O(z^6).$$

By assuming, that y_n is an exact approximation to $y(t_n)$, we find the following expressions for the local errors of the methods CRK, RRK6 and RRK5

(3.5)
$$\text{Local error CRK} = \frac{1}{120}h^5\lambda^5 + O(h^6\lambda^6),$$
$$\text{Local error RRK6} = (\frac{1}{120} - \frac{1}{432})h^5\lambda^5 + O(h^6\lambda^6),$$
$$\text{Local error RRK5} = (\frac{1}{120} - \frac{1}{864})h^5\lambda^5 + O(h^6\lambda^6).$$

In order to compare CRK, RRK6 and RRK5, we allow these methods to make an equal number of function evaluations. This means, that the stepsize used by RRK6, which we will call h_{RRK6}, is $\frac{6}{4}$ times as large as the stepsize used by CRK and $\frac{6}{5}$ times as large as the stepsize used by RRK5. Finally, we derive the global errors at t=1, using

(3.6) $$R^M(z) - e^{Mz} \approx (R(z) - e^z)\cdot M\cdot e^{(M-1)z} = LE\cdot M\cdot e^{\lambda(1-h)}, \qquad z := \lambda h.$$

Here, R(z) is one of the polynomials $R_{CRK}(z)$, $R_{RRK5}(z)$ or $R_{RRK6}(z)$, LE denotes the local error and $M := \frac{1}{h}$. This yields

(3.7)
$$\text{Global error CRK} \approx 0.0017\cdot(h_{RRK6})^4\lambda^5\exp(\lambda(1 - \frac{2}{3}h_{RRK6})) + O((h_{RRK6})^5)$$
$$\text{Global error RRK6} \approx 0.0072\cdot(h_{RRK6})^4\lambda^5\exp(\lambda(1 - h_{RRK6})) + O((h_{RRK6})^5)$$
$$\text{Global error RRK5} \approx 0.0029\cdot(h_{RRK6})^4\lambda^5\exp(\lambda(1 - \frac{5}{6}h_{RRK6})) + O((h_{RRK6})^5).$$

Thus, integrating the test equation (3.1) shows that CRK is more economical than RRK5 and RRK6, because it yields a more accurate solution. It is to be expected, that CRK is also cheaper in solving other differential equations. To illustrate this, we will perform some numerical experiments in the next paragraph.

4. Numerical experiments.

We used CRK, RRK5 and RRK6 to solve problems (3.1), (4.1) and (4.2), allowing each method to require the same number of function evaluations, say N.

(4.1) $$y'(t) = \sin(y^5) - \sin(\sin^5(t)) + \cos(t), \quad y(0) = 0, \quad 0 \le t \le \frac{\pi}{2}$$

exact solution $y(t) = \sin(t)$.

(4.2) $y'(t) = -y^3 + t^9(10 + t^{21})$, $y(0) = 0$, $0 \le t \le 1$

exact solution $y(t) = t^{10}$.

N must be a common multiple of 5q+1, 4 and 6, q an integer . For a few choices of q we found the following table of results, writing the absolute error obtained at the end of the integration interval in the form 10^{-d} (d may be interpreted as the number of correct decimal digits).

Table 4.1 correct decimal digits

problem	method	number of function evaluations 36	96	216	396	616	1596
(3.1)	CRK	5.50	7.18	8.58	9.63	10.4	12.1
	RRK5	5.14	6.84	8.25	9.30	10.1	11.7
	RRK6	4.95	6.62	8.02	9.07	9.82	11.5
(4.1)	CRK	3.69	5.36	6.76	7.81	8.58	10.2
	RRK5	3.34	5.03	6.43	7.48	8.25	9.90
	RRK6	3.14	4.76	6.15	7.19	7.94	9.60
(4.2)	CRK	2.96	4.77	6.29	7.40	8.20	9.89
	RRK5	3.18	4.70	6.08	7.13	7.90	9.55
	RRK6	2.97	4.42	5.77	6.81	7.56	9.22

These results sustain the conclusion of the previous paragraph; the classical Runge-Kutta method is more efficient than RRK5.

Acknowledgement. The auhor likes to thank Prof.dr. P.J. van der Houwen and drs. B.P. Sommeijer for their suggestions and improvements to this paper.

REFERENCES

[1] **Rosser, J.B.**, A Runge-Kutta For All Seasons, SIAM rev., vol 9, 417-452 (1967).

[2] **Lambert, J.D.**, Computional Methods in Ordinary Differential Equations, John Wiley & Sons, 1973.

Note NM-N9001
Centre for Mathematics and Computer Science
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands