Appl. Math. Lett. Vol. 2, No. 4, pp. 321–325, 1989
Printed in Great Britain.

0893-9659/89 $3.00 + 0.00

A 3(2) Pair of Runge - Kutta Formulas

P. BOGACKI, L.F. SHAMPINE[1]

Mathematics Department, Southern Methodist University

INTRODUCTION

Low order explicit Runge - Kutta formulas are quite popular for the solution of partial differential equations (PDEs) by semi-discretization, but in general-purpose codes for the solution of the initial value problem for a system of ordinary differential equations (ODEs), current practice favors moderate to high order. Nevertheless, it is observed that a low order formula is more efficient at crude accuracies. Also, the stability of the formula is especially important at these accuracies, and the stability properties of explicit Runge - Kutta formulas worsen considerably as one goes to (efficient) higher order formulas. A matter of considerable importance is the availability of "free" interpolants for low order formulas; it is even possible to obtain interpolants that preserve qualitative properties like monotonicity and convexity [1], [5]. Besides the obvious value of this for plotting, these interpolants are the key to the efficient location of events [6].

Comparatively little attention has been devoted to low order pairs of explicit Runge - Kutta formulas. We shall mention some pairs that have been proposed, and explain why the pair we propose is either more efficient, more reliable, or has better stability. We have chosen to base our pair on a three stage, third order formula because among the minimal cost formulas, it is arguably the best with respect to stability. Also, this is the highest order for which the "free" shape preserving interpolants are available. Much of the solution of PDEs by semi-discretization is done with a single formula and fixed step size. We observe that the automatic control of step size with an efficient pair such as ours involves little cost per step. Not only does the control pay for itself by providing the most efficient step size, but it also avoids step sizes that lead to instability.

2. THIRD ORDER FORMULA

We wish to solve the initial value problem for a system of ordinary differential equations

$$y'(x) = f(x, y(x)), \qquad a \le x \le b, \tag{2.1a}$$

$$y(a) \text{ given.} \tag{2.1b}$$

Approximations $\hat{y}_n$ are computed to $y(x_n)$ on a mesh $a = x_0 < x_1 < \ldots < x_N = b$. Starting with $\hat{y}_0$ equal to the given value $y(a)$, an explicit Runge - Kutta formula of s stages advances from $\hat{y}_n$ to $\hat{y}_{n+1}$ by a recipe of the form

$$\hat{y}_{n+1} = \hat{y}_n + h \sum_{i=1}^{s} \hat{b}_i k_i \tag{2.2}$$

[1]This work was supported by the Applied Mathematical Sciences program of the Office of Energy Research under DOE grant DE-FG05-86ER25024.

where

$$k_1 = f(x_n, \hat{y}_n) \tag{2.3a}$$

$$k_i = f(x_n + c_i h, \hat{y}_n + h\sum_{j=1}^{i-1} a_{i,j} k_j), \quad i = 2, ..., s. \tag{2.3b}$$

Here

$$c_i = \sum_{j=1}^{i-1} a_{i,j}.$$

The local solution at x_n is the solution $u(x)$ of (2.1a) that has the value $\hat{y}_n$ at x_n. The local error of the Runge - Kutta formula at x_n is $u(x_n + h) - \hat{y}_{n+1}$. Taylor expansion about x_n leads for smooth functions f to

$$u(x_n + h) - \hat{y}_{n+1} = \sum_{i=1}^{\infty} h^i \sum_{j=1}^{n_i} \hat{\alpha}_j^{(i)} D_j^{(i)} \tag{2.4}$$

Here the $D_j^{(i)}$ are elementary differentials, sums of products of partial derivatives of f, that depend only on the problem. The coefficients $\hat{\alpha}_j^{(i)}$ depend only on the coefficients $a_{i,j}$ and $\hat{b}_i$ defining the Runge - Kutta formula. Expressions for them may be found in [3].

The Runge - Kutta formula (2.2) is of order three if the equations of condition

$$\hat{\alpha}_1^{(1)} = 0, \ \hat{\alpha}_1^{(2)} = 0, \ \hat{\alpha}_1^{(3)} = 0, \ \hat{\alpha}_2^{(3)} = 0 \tag{2.5}$$

are satisfied. This states that the non - zero terms in (2.4) begin with the power h^4. It is known that at least three stages s are required for order 3. To keep the cost per step down, we restrict our attention to the minimum number of stages. There is a two parameter family of three stage, third order Runge - Kutta formulas [9]. Naturally we wish to choose the parameters to get the best possible formula. All such formulas have the same stability region, so we give our attention to the accuracy of the formula. Because of the presence of the problem - dependent elementary differentials in the expression for the local error, there is no choice of parameters best for all problems. It is conventional to make the coefficients $\hat{\alpha}_j^{(4)}$ of the leading terms in the local error (2.4), the truncation error coefficients, small in some sense so as to have a formula that is accurate for the "typical" problem. Several norms of the vector of truncation error coefficients are seen. In our selection of a formula of order two, we use the Euclidean norm, but in the present situation the choice of norm is not important because the two free parameters do not influence the value of $\hat{\alpha}_4^{(4)}$, which is $-\frac{1}{24}$. The best that can be done is to make the remaining coefficients small. An attractive possibility is to make some of these coefficients zero because then the formula will be of order 4 for some problems. In [9] Ralston presents the formula

$$\begin{array}{c|cccc} 0 & & & & \\ \frac{1}{2} & \frac{1}{2} & & & \\ \frac{3}{4} & 0 & \frac{3}{4} & & \\ \hline 1 & \frac{2}{9} & \frac{1}{3} & \frac{4}{9} & \end{array} \tag{2.6}$$

This formula does not quite minimize the Euclidean norm of the truncation error coefficients, but the difference is too small to have any practical significance. The truncation error coefficients $\hat{\alpha}_2^{(4)}$ and $\hat{\alpha}_3^{(4)}$ are zero. The formula has distinct c_i, which is advantageous for

a reason we take up below. A minor advantage is that the formula has "nice" coefficients. Because the formula seems as good as any, we have adopted it as the third order formula of the pair we derive.

3. Second Order Formula

Now we wish to reuse the stages formed in the evaluation of the formula of order three to get a result y_{n+1} of order two. Fehlberg [4] and Dormand and Prince [2] give pairs in which the first two stages are used for this purpose:

$$y_{n+1} = y_n + h(b_1 k_1 + b_2 k_2)$$

This fails to exploit the situation in two ways. All two stage, second order Runge - Kutta formulas have the same stability region, just as all three stage, third order formulas do. In this way of proceeding, the stability regions are not well matched, and there is nothing that can be done about it. With only two stages there is little flexibility for achieving an error estimate of high quality.

Additional flexibility is gained by realizing that most steps are a success, and if a step is a success, the first stage of the next step is always $f(x_{n+1}, \hat{y}_{n+1})$. If we add the stage

$$k_4 = f(x_n + h,\ \hat{y}_n + h\sum_{j=1}^{3} \hat{b}_j k_j),$$

and consider second order formulas of the form

$$y_{n+1} = \hat{y}_n + h\sum_{i=1}^{4} b_i k_i, \tag{3.1}$$

we gain flexibility at very little additional cost. This approach is now called FSAL, First Same As Last. In this approach we must specify which of the pair of formulas will be used to advance the step. We prefer the higher order formula because it is more accurate and more stable. This choice, called local extrapolation, is seen in all the popular codes now, but in [4] Fehlberg chose to advance with his second order formula. To get a more accurate second order formula, he used three stages, and the stage gained as described here is used for the construction of a third order formula.

Because we consider FSAL formulas, we evaluate f at $x_n + h$. One of the reasons we chose the Ralston third order formula is that it does not evaluate at this point. Thus our pair will sample f at four values of x in each step, and further, the two formulas do not share all these samples. We believe that this provides a little more robustness than, say, the Dormand - Prince pair which samples at three values, and that it improves the reliability of the estimate of the error of the second order formula.

Because all the stages used in the formula (3.1) have been specified by our choice of the third order formula and our decision to use FSAL along with local extrapolation, we need only select the coefficients b_i. The equations of condition

$$\alpha_1^{(1)} = 0,\ \alpha_1^{(2)} = 0$$

leave us with two free parameters. We select the parameters so as to obtain a pair of formulas of high quality. Prince and Dormand [8] discuss suitable criteria. In addition to their criteria, we add the natural desire for an accurate second order formula, namely that $\| \alpha^{(3)} \|_2$ be "small". The construction makes it clear that we could make this quantity vanish, i.e., make the formula of order 3. Other measures of quality that we take up prevent

this degeneration, but it is also important to avoid having either of the two truncation error coefficients $\alpha_1^{(3)}$ and $\alpha_2^{(3)}$ vanish. If one were to vanish, the formula would be of third order for some problems. This erratic behavior causes difficulties for step size adjustment algorithms. We have restricted our attention to formulas such that the two truncation error coefficients are the same, and non-zero, in order that the behavior of the formula be as uniform as possible. In the truncation error expansion (2.4) we want the leading term to dominate so that the formula "looks like" a second order formula for even comparatively large step sizes h. As a measure of this we use

$$B = \frac{\| \alpha^{(4)} \|_2}{\| \alpha^{(3)} \|_2}.$$

The error in the second order formula is estimated by comparison to the third order formula. Subtracting the Taylor series expansions of their local errors shows that an accurate estimate of the error will be provided by formulas for which the measure

$$C = \frac{\| \alpha^{(4)} - \hat{\alpha}^{(4)} \|_2}{\| \alpha^{(3)} \|_2}$$

is "small". Our goal was to choose the parameters so that B and C are of similar size and "small". As a final measure of quality, we sought parameters that would yield a second order formula with a stability region that includes that of the third order formula, but is not a lot bigger.

Certain of the constraints mentioned can be translated into mathematical constraints on the parameters, but most are sufficiently vague that one must explore the space of parameters in a heuristic way. After finding a "best" formula, we modified the coefficients slightly to obtain a pair of high quality that involved only "nice" coefficients. Our result is the line to be added to the array of (2.6)

$$\begin{array}{c|cccc} & & & & \\ \hline \frac{7}{24} & \frac{1}{4} & \frac{1}{3} & \frac{1}{8} \end{array}$$

Naturally we should compare our new pair to those of Dormand and Prince and Fehlberg. The Dormand - Prince pair has a two stage, second order formula combined with a three stage, third order formula. The stability regions of both are then fully specified, and consultation of the literature, e.g.[7, p.227], shows that the regions are not well matched. In contrast, our second order formula has a stability region that includes that of the third order formula. It is not greatly larger, but the details do not matter in view of the fact that we advance the integration with the third order formula. Dormand and Prince [3] agree that a poor match of stability regions "could cause severe problems in step size control, resulting in a loss of computational efficiency." Fehlberg's second order formula has a stability region that matches that of his third order formula very closely. Unfortunately, this is because his second order formula has truncation error coefficients that are so small that the formula "looks like" a third order formula. For the pair to behave as expected, and in particular for the error estimate to be accurate, the step size must be small enough that the leading term in the error expansion dominates. With this pair, the step size must be very small for this to be true, so small that a higher order formula would usually be more efficient. In contrast, the second order formula of Dormand and Prince is considerably less accurate than ours, which makes it less efficient. These facts are quantified by the measures of quality displayed in the following table.

Method	$\| \alpha^{(3)} \|_2$	B	C
Bogacki-Shampine 3(2)	2.94×10^{-2}	1.35	1.38
Dormand-Prince 3(2)	1.72×10^{-1}	0.81	1.01
Fehlberg 2(3)	6.70×10^{-4}	72.5	4.66

Table 1. Measures of quality for some Runge - Kutta pairs of orders two and three.

References

1. R.W. Brankin and I. Gladwell, *Use of Slope Preserving Interpolants for Plotting Solutions of ODEs*, IMA J. Numer. Anal..
2. J.R. Dormand, M.A. Lockyer, N.E. McGorrigan and P.J. Prince, *Global Error Estimation with Runge - Kutta Triples*, Comp. and Maths. with Applics..
3. J.R. Dormand, and P.J. Prince, *A Family of Embedded Runge - Kutta Formulae*, J. of Comp. and Appl. Math. **1** (1980), 19–26.
4. E. Fehlberg, *Klassische Runge - Kutta - Formeln vierter und niedrigerer Ordnung mit Schrittweiten - Kontrolle und ihre Anwendung auf Warmeleitungsprobleme*, Computing **6** (1970), 61–71.
5. I. Gladwell, L.F. Shampine, L.S. Baca, and R.W. Brankin, *Practical Aspects of Interpolation in Runge - Kutta Codes*, SIAM J. Sci., Stat. Comp., **3** (1987), 322–341.
6. I. Gladwell, L.F. Shampine, and R.W. Brankin, *Locating Special Events when Solving ODEs*, Appl. Math. Lett., **1** (1988), 153–156.
7. J.D. Lambert, "Computational Methods in Ordinary Differential Equations," Wiley, New York, 1973.
8. P.J. Prince, and J.R. Dormand, *High Order Embedded Runge - Kutta Formulae*, J. of Comp. and Appl. Math. **1** (1981), 67–75.
9. A. Ralston, "A First Course in Numerical Analysis," McGraw-Hill, New York, 1965.

Mathematics Department, Southern Methodist University, Dallas, TX 75275