Local Extrapolation in the Solution of Ordinary Differential Equations

By L. F. Shampine*

Abstract. The local errors being estimated in the solution of an initial value problem can be added in to make the solution more accurate but this is not always advisable. A rule for deciding when to extrapolate is studied for one-step methods. Some observations about the correctness of local error estimators and extrapolation of multistep methods are also made.

Introduction. In a recent paper [1], the author and H. A. Watts studied the efficiency of a number of local error estimators for Runge-Kutta methods. Local error is not the only possibility for controlling error in codes for the initial value problem, but it is almost universally used because it is the most practical. There are two other applications of local error estimates which are important. One is to make a locally optimal choice of step size. The other is the subject of this paper; one can add in the estimated error to improve (hopefully) the accuracy of his computations. This local extrapolation asymptotically raises the order of the method by one. Still, extrapolation is not necessarily advantageous. The extrapolated value can be less accurate and stability can be seriously affected. We have seen working codes which never extrapolate, always extrapolate, and which provide it as an option. We shall give a simple test which circumvents stability problems for one-step methods and which appears to be effective at choosing to extrapolate only if the accuracy is enhanced. The test is applicable to multistep methods but there are difficulties in this case.

We first make some observations about local error estimators and their use in extrapolation. In [1], we were unable to interpret some Runge-Kutta formulas of Zonneveld in terms of local error, but we can interpret them very naturally in this context. We also study two other variants of Runge-Kutta which are candidates for an efficient Runge-Kutta code. Some interesting observations about error estimators for multistep methods are made too. We consider extrapolation of Adams formulas and point out a relation which does not seem as well known as it ought to be. Two rather well-known predictor-corrector pairs are shown by counterexample to have local error estimators which are *not* asymptotically correct. We provide some computations of stability regions for Adams methods which may be of general interest.

Local Error Estimators. In [1], we surveyed local error estimators for Runge-Kutta methods and used a set of formulas of Zonneveld [2] to illustrate a different

way of measuring error. Motivated by Taylor series methods, he estimates the last term of the Taylor expansion that has been taken into account. For example, his second-order formula is

$$k_0 = f(x_0, y_0), \quad k_1 = f(x_0, y_0 + hk_0), \quad y_1 = y_0 + h(k_0 + k_1)/2,$$

along with the estimate of the term of order h^2 going into $y(x_1)$,

$$\text{term } O(h^2) \doteq h(k_1 - k_0)/2.$$

We describe this in terms of the first-order formula $y_1^* = y_0 + hk_0$. Its local error is

$$\text{local error} = y(x_1) - y_1^* = y_1 - y_1^* + O(h^3) = h(k_1 - k_0)/2 + O(h^3).$$

Moreover,

$$y_1^* + h(k_1 - k_0)/2 = y_1^* + (y_1 - y_1^*) = y_1.$$

That is, we say his formula is a *first*-order formula with an asymptotically correct local error estimator and the formula is always extrapolated to become second-order. This is true of all of Zonneveld's formulas. His formula of order p can be described as a formula of order $p - 1$ combined with a local error estimator to extrapolate to order p. Since his step adjustment is a locally optimal one based on order $p - 1$, it is clear this is the proper way to view his formulas. This observation not only brings his set of formulas into the same framework as the others studied but his codes also serve as examples of working codes which always extrapolate. We study the formula he describes as fifth-order below by deducing the underlying fourth-order formula. It requires seven evaluations per step with the error estimate.

In [1], we contrasted the classical Runge-Kutta fourth-order scheme with doubling error estimator to a procedure using a scheme of England. The text [3] gives a classical code of this type which always extrapolates (the constants in the error estimator are incorrect). Two additional schemes which are quite efficient have since come to our attention. Fehlberg [4] gives a fourth-order procedure using six function evaluations per step with an asymptotically correct local error estimator. He explicitly derives the formula by deriving formulas of orders four and five and using their difference as an estimate of the error in the lower order formula. Of course, the parameters are chosen so that most function evaluations are used in both formulas to keep the total cost down to six evaluations per step. Shintani [5] has given an effective scheme using seven evaluations per double step with a similar derivation. Along with England's scheme, and perhaps Zonneveld's, we feel these are the principal possibilities for an efficient Runge-Kutta code. Their fair comparison is not an easy matter. Shampine and Watts [1] prefer to use England's scheme instead of doubling, but local extrapolation is not considered in their comparison. Examination of the stability regions with respect to the effect of extrapolating was partly responsible for this paper. Extrapolation substantially increases the stability regions of Fehlberg's and Zonneveld's schemes and decreases the others. Attempting to prevent this stability loss for England's scheme led to the device we present which is completely successful in this context.

The only device we know of in serious use for estimating local errors of multistep methods is Milne's device of comparing predicted and corrected values [6, p. 256].

A very important example is the combination of Adams-Bashforth predictor and Adams-Moulton corrector. There is an interesting point about extrapolation in this case which appears not to be well known. The Adams-Bashforth formula of order k is

$$y_n = y_{n-1} + h \sum_{j=0}^{k-1} \gamma_j \nabla^j f_{n-1}$$

where

$$f_{n-1} = f(x_{n-1}, y_{n-1}),$$

$$\nabla^j = j\text{th backward difference},$$

$$\gamma_j = (-1)^j \int_0^1 \binom{-s}{j} ds.$$

Similarly, the Adams-Moulton formula of order k is

$$y_n = y_{n-1} + h \sum_{j=0}^{k-1} \gamma_j^* \nabla^j f_n$$

where

$$\gamma_j^* = (-1)^j \int_0^1 \binom{-s+1}{j} ds.$$

To distinguish the Adams-Bashforth value, we shall write y_n^P in what follows. If one presumes the values $y_{n-1}, y_{n-2}, \cdots$ are exactly equal to $y(x_{n-1}), y(x_{n-2}), \cdots$, a smooth solution of $y' = f(x, y)$, he finds

$$y(x_n) - y_n^P = \gamma_k h^{k+1} y^{(k+1)}(\xi) + O(h^{k+2}),$$

$$y(x_n) - y_n = \gamma_k^* h^{k+1} y^{(k+1)}(\eta) + O(h^{k+2}).$$

Then a little manipulation shows

$$\text{local error} = y(x_n) - y_n = \gamma_k^*(y_n - y_n^P)/(\gamma_k - \gamma_k^*) + O(h^{k+2}).$$

The first term is the computable local error estimate. It is an easy exercise to show that $y_n = y_n^P + h\gamma_{k-1}\nabla^k f_n$, once one realizes that $\gamma_0^* = \gamma_0$, $\gamma_j^* = \gamma_j - \gamma_{j-1}$ for $j \geqq 1$. If one were to use this Adams-Bashforth predictor of order k and an Adams-Moulton corrector of order $k + 1$, he would find $y_n^E = y_n^P + h\gamma_k \nabla^k f_n$ where we use y_n^E to distinguish the result of the corrector of higher order. What has this to do with extrapolation? To estimate the error, we must use predictor and corrector of the same order as we did above and then if we extrapolate by forming

$$y_n + \frac{\gamma_k^*}{\gamma_k - \gamma_k^*}(y_n - y_n^P) = y_n + \frac{\gamma_k^*}{\gamma_{k-1}}(h\gamma_{k-1}\nabla^k f_n)$$

$$= y_n^P + h\gamma_{k-1}\nabla^k f_n + h\gamma_k^* \nabla^k f_n = y_n^E,$$

we find this is the same as using a corrector of one higher order. As we understand it, the variable order, variable step Adams code of F. T. Krogh [7], which has received well deserved attention, always extrapolates in this way.

An interesting point about Milne's device is that it is almost always applied with the assumption above that the memorized values are exact, which they obviously are not. One ought to be suspicious that the errors actually present might invalidate the application. Henrici hints at this when he says in [6, p. 257] that "To justify a similar device for calculating $y^{(p+1)}$ in the general case where the values $y_n, y_{n+1}, \cdots,$ y_{n+k-1} are not assumed to be exact, it seems necessary to assume that the left-hand terms of the predictor and corrector formulas should be the same $\cdots$." He goes on to give a complete proof of the asymptotic correctness of the error estimator previously stated for the Adams methods which takes into account the propagated errors. In this very important case, the usual plausibility argument happens to lead to the correct conclusion but this is not always so.

The midpoint-trapezoidal rule combination has been frequently used in articles to illustrate Milne's device but the estimator is not asymptotically correct! A counter-example is readily constructed. The rule in PECE form is

$$y_{n+1}^P = y_{n-1} + 2hf(x_n, y_n),$$

$$y_{n+1} = y_n + h(f(x_{n+1}, y_{n+1}^P) + f(x_n, y_n))/2,$$

$$\text{est} = (y_{n+1}^P - y_{n+1})/5.$$

Apply this to $y' = y$, $y(0) = 1$ with a constant step of h. Start the process with the exact values $y_0 = 1$, $y_1 = e^h$. The local error at x_n requires the solution of

$$u' = u, \qquad u(x_n) = y_n$$

for then, by definition,

$$\text{local error} = \text{l.e.} = u(x_{n+1}) - y_{n+1} = e^h y_n - y_{n+1}.$$

Because y_0, y_1 are exact, the local error at x_1 is asymptotically correct. At x_2 however, there is error present and a straightforward calculation shows that

$$\text{l.e.} = -h^3/12 + O(h^4),$$

$$\text{est} = (y_3^P - y_3)/5 = -h^3/10 + O(h^4).$$

A similar observation of considerably more significance is that Hamming's formulas [8]

$$y_{n+1}^P = y_{n-3} + 4h[2y_n' - y_{n-1}' + 2y_{n-2}']/3,$$

$$y_{n+1} = [9y_n - y_{n-2}]/8 + 3h[y_{n+1}^{P'} - 2y_n' - y_{n-1}']/8,$$

$$\text{est} = 9(y_{n+1}^P - y_{n+1})/121$$

are incorrect in the same way. The same counterexample suffices though the calculations are more tedious. Starting with exact values, one finds

$$\text{l.e.} = e^h y_4 - y_5 = -9h^5/320 + O(h^6)$$

$$\text{est} = 9(y_5^P - y_5)/121 = -1049h^5/38720 + O(h^6).$$

In decimal form, the coefficients are respectively, -0.028125, -0.0270919. Since the error estimate is not asymptotically correct, we recommend the method be abandoned. Hamming qualifiedly advocates extrapolation though our observation says

it is not justified in general. The implementation in the IBM SSP always extrapolates. Apparently, the estimate is nearly correct in most cases and the effect of being incorrect quite difficult to spot. It is interesting that later Hamming [9, p. 205] advocates predictor-corrector combinations of the form envisioned by Henrici but for reasons not associated with their error estimator.

When to Extrapolate.   Experience seems to show that the benefit gained from local extrapolation is quite modest and it is the fact that it costs practically nothing which makes it worthwhile. With any of the methods discussed, we can write the result of a step as $y_{n+1} = y_n + h\Phi_n$. If $u(x)$ is the solution of the differential equation such that $u(x_n) = y_n$, we have

$$u(x_{n+1}) = y_n + h\Phi_n + \text{l.e.}$$

Let us denote the local error estimate by est. We propose that if $|\text{est}| \leq \alpha\,|h\Phi_n|$, then extrapolate by forming $y_{n+1} + \text{est}$ and otherwise do not. Here, α is a constant and we suggest a value of $1/16$.

This test is a natural one on the accuracy and asymptotic behavior. If the method is of order p, the term $h\Phi_n$ is no smaller than $O(h^p)$, except at isolated points and in very special cases, and the local error is $O(h^{p+1})$. Except in rare instances, the test is always passed asymptotically and one always extrapolates as he should. If the test is failed, one has good grounds for suspecting that the asymptotic basis of the computations is insufficiently valid to permit extrapolation. The value of α does not seem to matter much in this argument; it seems plausible to choose it so that the increment has a few digits unaffected by extrapolation.

Besides attempting to extrapolate only when the accuracy will be improved, we also want to prevent any loss in stability. For one-step methods, this test in effect defines a new method and we can compute the stability region associated with it in the usual way. The regions depend on the parameter α of course and we seek an α such that the modified extrapolation does not affect the stability region in a serious way. We have made these plots for various α and the various methods. The most important of them may be found in the report [10]. Here, for the sake of brevity, we describe the results another way. We computed the stability regions in polar coordinates. For a given angle θ, let R_α denote the distance along the radius to the boundary of the stability region for the given α and method. A value of $\alpha = 0$ corresponds to never extrapolating, and $\alpha = \infty$ to always extrapolating. Among other quantities, we computed the average values of R_∞/R_0, $R_{1/16}/R_0$. The first quantity measures, roughly, whether always extrapolating is desirable or not. The second quantity shows whether or not our device with $\alpha = 1/16$ is effective in preserving the stability properties of the method. We found the average ratio and the plots both show that $\alpha = 1/16$ is sufficiently small that the stability region with this modified extrapolation is virtually identical to that without extrapolation at all. Thus, we are completely successful, in the sense of stability regions, in removing stability difficulties associated with local extrapolation. We remark that $\alpha = 1/16$ is sufficiently small but all smaller α give regions still closer to the regions without extrapolation so one can use smaller α as he prefers. With this device we recommend that one extrapolate when using these methods.

Method	average R_∞/R_0	average $R_{1/16}/R_0$
Classical with doubling	0.89	1.00
England	0.70	1.00
Fehlberg	1.15	1.00
Shintani	0.92	1.00
Zonneveld	1.43	1.00
Adams order 1	1.51	1.00

We have included in this table the Adams-Bashforth-Moulton PECE combination of order one which is a one-step method. This is an important procedure for starting variable order Adams codes such as Gear's [3] and Krogh's [7]. The device described can obviously be applied to multistep methods in general, but there are conceptual difficulties. The analysis of the test in terms of stability regions is not applicable because, with memory, this device does not lead to a fixed difference scheme. Moreover, one does not then have a rigorous basis for the local error estimate because an asymptotic expansion for the error is present only when h is sufficiently small that one is essentially always extrapolating. Because of this, we only consider the options of never extrapolating or always extrapolating; then stability regions are defined in the standard way. We have computed the stability regions for the Adams PECE combinations of orders one through eleven with and without extrapolation. All plots are found in the report [10]. The following table gives the average ratio of the radius for always extrapolating to never extrapolating.

Order	average R_∞/R_0
1	1.53
2	1.12
3	1.08
4	1.06
5	1.05
6	1.05
7	1.05
8	1.05
9	1.03
10	1.00
11	.91

For order one, the average ratio R_∞/R_0 differs somewhat from the value in the preceding table. Because of the expense of the Adams computations, many fewer points were obtained on the boundary of the region than for the one-step computations. Extrapolation seems to increase stability on the average for the Adams methods as well as improving the accuracy; so we recommend it be used, at least for orders through 10.

Applied Mathematics Division 1722
Sandia Laboratories
Albuquerque, New Mexico 87115

1. L. F. Shampine & H. A. Watts, "Comparing error estimators for Runge-Kutta methods," *Math. Comp.*, v. 25, 1971, pp. 445–455.

2. J. A. Zonneveld, *Automatic Numerical Integration*, Math. Centre Tracts, no. 8, Mathematisch Centrum, Amsterdam, 1964. MR **30**, #1612.

3. C. W. Gear, *Numerical Initial Value Problems in Ordinary Differential Equations*, Prentice-Hall, Englewood Cliffs, N. J., 1971.

4. E. Fehlberg, "Klassische Runge-Kutta-Formeln vierter und niedrigerer Ordnung mit Schrittweiten-Kontrolle und ihre Anwendung auf Wärmeleitungsprobleme," *Computing (Arch. Elektron. Rechnen)*, v. 6, 1970, pp. 61–71. MR **43** #5728.

5. H. Shintani, "Two-step processes by one-step methods of order 3 and of order 4," *J. Sci. Hiroshima Univ. Ser. A-I Math.*, v. 30, 1966, pp. 183–195. MR **34** #7032.

6. P. Henrici, *Discrete Variable Methods in Ordinary Differential Equations*, Wiley, New York, 1962. MR **24** #B1772.

7. F. T. Krogh, VODQ/SVDQ/DVDQ-*Variable Order Integrators for the Numerical Solution of Ordinary Differential Equations*, TU Doc. No. CP-2308, NPO-11643, May 1969; *On Testing A Subroutine for the Numerical Integration of Ordinary Differential Equations*, JPL Internal Document, Section 314, TM No. 217 (revised), October 1970, Jet Propulsion Laboratory, Pasadena, Calif.

8. R. W. Hamming, "Stable predictor-corrector methods for ordinary differential equations," *J. Assoc. Comput. Math.*, v. 6, 1959, pp. 37–47. MR **21** #973.

9. R. W. Hamming, *Numerical Methods for Scientists and Engineers*, Internat. Series in Pure and Appl. Math., McGraw-Hill, New York, 1962. MR **25** #735.

10. L. F. Shampine, *Stability Regions for Extrapolated Runge-Kutta and Adams Methods*, SC-RR-72 0223, March 1972.