JOURNAL OF COMPUTATIONAL PHYSICS 41, 295–308 (1981)

Comparison of Numerical Methods for the Integration of the Black Hole Geodesic Equations

N. A. SHARP*

Institute of Astronomy, Madingley Road, Cambridge CB3 0HA, England

Received February 26, 1980; revised August 29, 1980

We compare four different techniques for the numerical integration of ordinary differential equations, and their performance on two initial value problems which are timelike geodesics around a rotating black hole.

1. INTRODUCTION

Black holes are now an accepted part of theoretical astrophysics. Because there is much support for the "no-hair" conjecture, that any physically existing black hole will be a member of the four-parameter generalised Kerr–Newman family of solutions to Einstein's equations [1–3] their geodesic structure has been much studied [4]. For astrophysical purposes, the timelike geodesics are of greatest importance, since many accretion models assume geodesic motion of the infalling matter, and the null geodesics are also needed in order to follow the propagation of radiation out to distant observers. Spacelike geodesics—the paths that would be followed by tachyons, if they exist—are used to help to elucidate the structure of the spacetime, particularly with reference to analytic continuations and "wormholes" (see, e.g., Fuller and Wheeler [5] and Brigman [6]). Because of the complexity of the differential equations governing the geodesic paths in this metric [4], most analysis has concentrated on various specialisations or particular symmetries, and the calculation of a general trajectory requires numerical integration.

After the equations are repeated for convenience, their calculational form is optimised. Possible numerical methods are discussed, and then compared by using two representative timelike tracks. All calculations were performed on an IBM 370/165 at the University of Cambridge.

* Current address: Center for Relativity, University of Texas, Austin, Tex. 78712.

0021-9991/81/060295-14$02.00/0

581/41/2-5

2. The Equations and Their Calculational Form

In the coordinate system t, r, θ, ϕ, generally known as the Boyer–Lindquist coordinate system (after Boyer and Lindquist [7]), the metric of the generalised Kerr–Newman solution is

$$ds^2 = \frac{Z}{\Delta}\,dr^2 + Z d\theta^2 + \frac{\sin^2\theta}{Z}\,[a dt - (r^2 + a^2)\,d\phi]^2 - \frac{\Delta}{Z}\,[dt - a\sin^2\theta\,d\phi]^2, \tag{1}$$

where $Z = r^2 + a^2\cos^2\theta$, $\Delta = r^2 - 2Mr + a^2 + Q^2 + P^2$, and M, a, Q and P are the hole's mass, specific angular momentum, electric charge and magnetic monopole moment. Throughout, we use geometrical units with $G = c = 1$. A complete set of first integrals exists for the geodesics:

$$\begin{aligned}
Z^2\dot{r}^2 &= R = [E(r^2 + a^2) - a\Phi - eQr]^2 - \Delta(m^2r^2 + K),\\
Z^2\dot{\theta}^2 &= \Theta = K - m^2a^2\cos^2\theta - \sin^{-2}\theta[Ea\sin^2\theta - \Phi + eP\cos\theta]^2,\\
Z\dot{\phi} &= \Phi\left[\sin^{-2}\theta - \frac{a^2}{\Delta}\right] + aE\left[\frac{r^2 + a^2}{\Delta} - 1\right] - e\left[\frac{Qra}{\Delta} + \frac{P\cos\theta}{\sin^2\theta}\right],\\
Z\dot{t} &= E\left[\frac{(r^2 + a^2)^2}{\Delta} - a^2\sin^2\theta\right] - a\Phi\left[\frac{r^2 + a^2}{\Delta} - 1\right]\\
&\quad - e\left[\frac{Qr(r^2 + a^2)}{\Delta} + aP\cos\theta\right].
\end{aligned} \tag{2}$$

There are four constants of the motion: m, the mass; E, the energy at infinity; Φ, the azimuthal angular momentum; and K, the generalisation of the total angular momentum; in addition to the particle's charge, e. The dot represents differentiation with respect to an affine parameter, λ. For numerical work, the appearance of squares in the equations for $\dot{r}$ and $\dot{\theta}$ means that the following second derivatives are required:

$$\begin{aligned}
Z^2\ddot{r} &= [E(r^2 + a^2) - a\Phi - eQr][2rE - eQ] - [m^2r^2 + K][r - M] - m^2r\Delta - \dot{r}Z\dot{Z},\\
Z^2\ddot{\theta} &= (m^2 - E^2)\,a^2\cos\theta\sin\theta + eaEP\sin\theta\\
&\quad + \sin^{-3}\theta(eP\cos\theta - \Phi)(eP - \Phi\cos\theta) - \dot{\theta}Z\dot{Z}.
\end{aligned} \tag{3}$$

These equations are valid directly for timelike paths with $m^2 > 0$ and for spacelike tracks with $m^2 < 0$, and also for null geodesics with $m^2 = 0$ by setting $m = e = 0$, $\Phi = 1$ (essentially adjusting the affine parameter), and relating E and K to the impact parameters. The affine parameter λ can be normalized to be the proper time τ for ordinary massive particles.

It is probable that an astrophysical black hole will be rotating, but without charge and without a magnetic monopole moment (particularly since this latter does not appear to exist in nature). In addition, the environment will probably be electrically neutral. For this case, we specialise to $e = Q = P = 0$. This leads to some simplifications, which are most marked in the equations

$$\begin{aligned} \Delta Z\dot{\phi} &= \Phi(Z - 2Mr)\sin^{-2}\theta + 2MraE, \\ \Delta Z\dot{t} &= E[Z(r^2 + a^2) + 2Mra^2\sin^2\theta] - 2Mra\Phi. \end{aligned} \tag{4}$$

These equations construct a geodesic by its affine parameter; i.e., they follow a particle in its own proper time. For many-body calculations of a bundle of particles, the viewpoint of the observer-at-infinity must be taken, and this means following the evolution in intervals of the coordinate time t. In addition, for single particles the use of coordinate time concentrates computing power at the parts of the trajectory closer to the hole, because of time-dilation. When stepping in proper time, the coordinate jumps near the event horizon become larger and larger, leading to inaccuracies. Our primary astrophysical concern is with the black hole exterior, and so, using a prime to denote d/dt, the new set of equations is given from the old set by

$$\begin{aligned} \lambda' = 1/\dot{t}, \qquad \phi' = \dot{\phi}/\dot{t}, \qquad \theta' = \dot{\theta}/\dot{t}, \qquad r' = \dot{r}/\dot{t}; \\ r'' = (\ddot{r} - \ddot{t}r')/\dot{t}^2, \qquad \theta'' = (\ddot{\theta} - \ddot{t}\theta')/\dot{t}^2. \end{aligned} \tag{5}$$

The derivatives $\dot{t}$ and $\ddot{t}$, which latter is derived from $\dot{t}$ as $\ddot{r}$ and $\ddot{\theta}$ were derived from $\dot{r}$ and $\dot{\theta}$, are now necessary intermediate functions. It is not, of course, essential to keep track of the affine parameter λ, but the extra work involved is small and the information is useful.

It is easiest to reduce the system of equations (5) to a canonical set of six, first-order, coupled, nonlinear ordinary differential equations

$$y^{i\prime} = f^i(t, y^j), \qquad i = 1, 6, \tag{6a}$$

with the identifications

$$\begin{aligned} y_{-1} = r' = y^{2\prime}, \qquad y^3 = \theta' = y^{4\prime}, \qquad y^5 = \phi; \\ y^2 = r, \qquad y^4 = \theta, \qquad y^6 = \lambda \end{aligned} \tag{6b}$$

although there is in fact no explicit dependence on t. It might seem better to calculate r' and θ' from the first integrals of the motion, using r'' or θ'' only to cope with the points where r' or θ' approach zero. Unfortunately, it turns out to be virtually impossible to monitor the turning point sufficiently well and to compute accurately enough in this way. Since we need a general method of solution it is more straightforward, just as accurate, and only a little slower, to use the full set of six equations (6). The first integrals can be used as checks on any errors, if desired.

The encoding of the set (6) for computer use is further simplified by two normalisations and some analytic cancellations. The normalisations are $M = 1$, so that the coordinates, hole parameters, and constants of the motion are in units of the hole's mass, and $m^2 = \pm 1$ so that the constants of the motion are further reduced to values per unit particle mass. The analytic cancellations arise because $\ddot{r}$, $\ddot{\theta}$ and $\ddot{t}$ are all derived from first derivative equations containing a factor of $1/Z$, and the terms in $\dot{Z}$ thus cancel out. Furthermore, the event horizon of the black hole is defined as the shell where $\Delta = 0$, which leads to problems because of the appearance of factors of $1/\Delta$. As many of these as possible are removed, for example, by calculating ϕ' as $(\Delta\dot{\phi}) \div (\Delta\dot{t})$.

3. Numerical Methods

(a) *Introduction*

The initial value problem defined by the set of coupled, first-order ordinary differential equations (6), with appropriate initial conditions $y^i(t_0)$, belongs to a class whose solution has occupied a great deal of time and analysis. All the numerical methods currently of use are of the type known as discrete variable methods. Simple methods use a constant stepsize, but generally it is more efficient to allow it to vary.

A good introduction to the theoretical basis of the numerical techniques is the book edited by Hall and Watt [8]. Comparisons of the methods, and of practical implementations of the methods, have appeared often, but particularly useful articles are those by Enright and Hull [9] and by Shampine *et al.* [10]. These references do not mention Taylor series methods (see below).

The best method for any particular problem depends on the behaviour of the equation set over the region of interest, and on the attributes required of the solution with regard to accuracy and speed. All the methods considered produce an estimate of the expected error in the solution at each step, and adjust themselves to keep this error below a specified value. Most tests use a mixed relative–absolute error criterion, where the error is controlled so that $|\text{Estimated error}| \leqslant \text{Maximum relative error allowed} \times |\text{Solution}| + \text{Maximum absolute error allowed}$. The techniques implemented here all set the allowable relative and absolute errors to the same tolerance. Unfortunately, such *local* error control at each step does not prevent the accumulation of inaccuracies so that the error at the end point may be unacceptably large. Despite progress towards methods which allow global error control, currently the best way to check the solution is to solve the same problem with different allowed local errors and to compare the answers obtained. Since more stringent limits increase the computer time required, the loosest constraint compatible with the required accuracy should be used.

Although general comparisons are useful guides to the relative merits of the available methods, choosing the best way to solve a given problem must involve test runs on a selection of initial conditions, two of which are presented below. For these

comparisons, all the routines were pre-compiled and stored as load modules in a FORTRAN subroutine library. This was to eliminate overhead and to make the comparisons fairer, since some methods involve many more lines of program than others. For convenience, the indices on the system $y^{i\prime} = f^i(t, y^i)$ will be dropped. Recall that y_n is the approximation to $y(t_n)$.

(b) *Runge–Kutta (R) Methods*

The Runge–Kutta routine considered was an explicit five-stage, fourth-order method due to Merson [11] (see also [8, 12]), which attempts to control the local error by varying the local steplength. It was taken from the N.A.G. library of numerical methods.

It is unfortunate that a better method, such as a Runge–Kutta–Fehlberg approach, was not readily available at the time of this work, because the Merson algorithm is known to be unreliable and inefficient for nonlinear problems [9]. However, it sill provides some basis for comparisons via DASCRU (a Merson method), RK4 and RKF4 of [9], and RKF4 and RKF45 of [10]. In general, these methods were a priori not expected to be highly competitive, because of the complex and expensive derivative evaluations required by Eqs. (2), (3), and because of the requirement of fairly high accuracy. It was therefore felt not to be worth implementing a better approach solely for the current study.

(c) *Adams (A) Method*

This was an Adams–Bashforth–Moulton predictor–corrector method of variable order, again taken from the N.A.G. library. It uses a divided difference representation of the required polynomials, and adjusts them according to the prescription of Krogh [13]. The error is controlled by changing both the stepsize and the order. For further details of this approach, see, e.g., [8, 10, 12, 14].

(d) *Taylor Series (T) Methods*

The Taylor series

$$y(t_{n+1}) = y(t_n) + \Delta t_{n+1} y'(t_n) + \frac{\Delta t_{n+1}^2}{2!} y''(t_n) + \cdots \tag{7}$$

gives what must be conceptually the simplest way to advance the solution from t_n to t_{n+1}. By truncating and approximating $y(t_n) = y_n$, we get the further approximation

$$y_{n+1} = y_n + \Delta t_{n+1} f_n + \frac{\Delta t_{n+1}^2}{2!} f_n' + \cdots + \frac{\Delta t_{n+1}^s}{s!} f_n^{(s)}. \tag{8}$$

The method can be described as an application of the process of analytic continuation. One generates a sequence of Taylor expansions which are valid successively in overlapping intervals, the totality of which intervals completely covers the desired domain of integration.

Assuming appropriate differentiability, such formulae are self-starting and allow easy change of stepsize Δt_n and of the number of terms used, s. The biggest problem faced by this idea is obtaining the higher derivatives of f, which is why most works ignore the Taylor series approach. However, it has been shown that reducing the original equations to a certain canonical form enables the automatic derivation of recurrence relations from which the unknowns may be calculated [15, 16]. This method has been implemented as an algebraic manipulation system [17], which takes as input the differential equations to be solved, and produces as output the FORTRAN subroutines to solve them. This system should be more widely disseminated.

(e) *Extrapolation—The Gragg–Bulirsch–Stoer (G) Method*

The basic idea of extrapolation routines is fairly straightforward. Evaluate an approximation to the value $y(t_{n+1})$ by using a simple rule and a given stepsize h_0. Evaluate further approximations using a sequence of decreasing stepsizes h_i. By assuming the form of an interpolating function to these values, it is possible to extrapolate to a better answer. Thus, combining results of low accuracy yields a highly accurate approximation. The particular assumption required is the functional form of the error, $F(h_i) = y_{n+1,h_i} - y(t_{n+1})$. The details of this method (first suggested by Richardson[1] [18, 19]) are rather complex, and are justified by a great deal of analysis. It is much like a Runge–Kutta method of variable order.

The routine used is due to Prof. Dr. R. Bulirsch, and was kindly provided by Dr. S. Aarseth. It uses a modified midpoint rule

$$\begin{aligned} y(t; h) &= \tfrac{1}{2}\{y_n + y_{n-1} + hf(t, y_n)\}, \\ y_{n+2} &= y_n + 2hf(t_n, y_n) \end{aligned} \tag{9}$$

and an error estimate which is a rational function (the division of two power series in h). The sequence h_i is taken to be $h_0 \times (1, \frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \frac{1}{6}, \frac{1}{8}, \ldots)$. The details are in [20–22]. The disadvantage of this particular routine is that, whilst adjusting the time-step, it outputs values at points which may not be the desired step-points. It must be (gently) coerced by the driving program to provide the desired evenly spaced results. This negates one of this method's greatest advantages.

[1] Lewis Fry Richardson (1881–1953) is an interesting and eccentric figure who devoted much of his life to studying armed conflicts ("Arms and Insecurity; a Mathematical Study of the Causes and Origins of War," ed. N. Rashevsky and E. Trucco, and "Statistics of Deadly Quarrels," ed. Q. Wright and C. C. Lienau, both published by Boxwood Press, Pittsburgh, 1960), but along the way invented the extrapolation method, started the mathematical study of the weather, did fundamental work on turbulence (note the "Richardson number" in fluid dynamics), and did work on what are nowadays called fractals. See also B. B. Mandelbrot, "Fractals: Form, Chance and Dimension," Freeman, San Francisco, 1977.

4. Two Test Problems–Results

Since all quantities are normalised to the hole's mass and the particle's mass, no units will be quoted. Thus, for example, "$r = 6.7$" is assumed to mean $r = 6.7M$ in geometrical units, and "$E = 0.95$" means $E/m = 0.95M$. (Remember that we have already set $G = c = 1$.)

Available analytic orbit solutions do not in general make good test material, because they nearly all specialise to constancy of some coordinate or other and so are

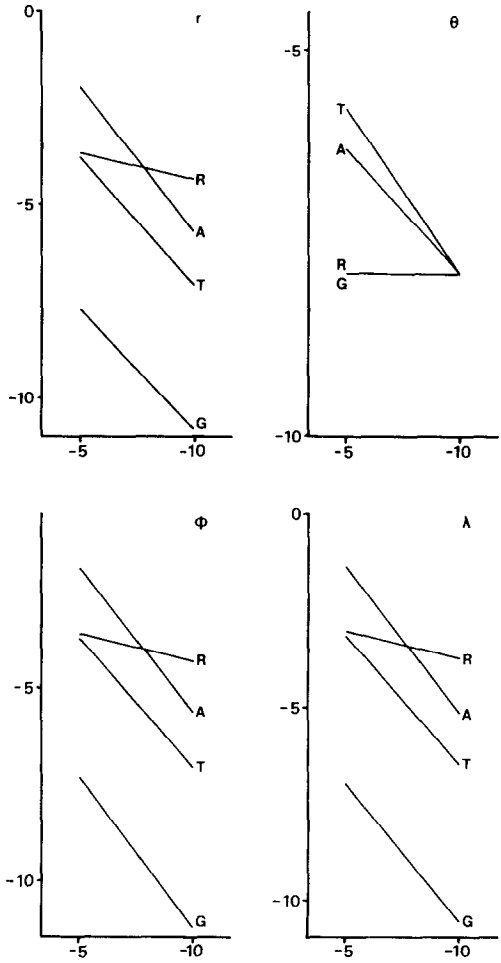

Fig. 1. Maximum error of each method in the four coordinates r, θ, ϕ and λ versus local error request for test one integrated to $t = 400$ (but see text). Logarithmic scales. A: Adams–Bashforth–Moulton, G: Gragg–Bulirsch–Stoer, R: Runge–Kutta, T: Taylor series. Note that the straight lines connect *only two data points*, being used solely to reveal the trend, and do *not* represent the functional form of the dependence.

not truly three dimensional. Thus, for example, orbits in the equatorial plane of the Kerr solution are very well studied. Where the relevant coordinate is constant because its derivative is identically zero, the numerical method cannot fail to be correct, and where the derivative is zero by cancellation, the solution is limited by computer round-off errors and not by the efficacy of the technique. The accuracy of the numerical solutions to the test problems is therefore decided by comparison with a computer-produced "true" solution. This "true" path was calculated, with the benefit of early work on the problem, by using the Adams method with the local error

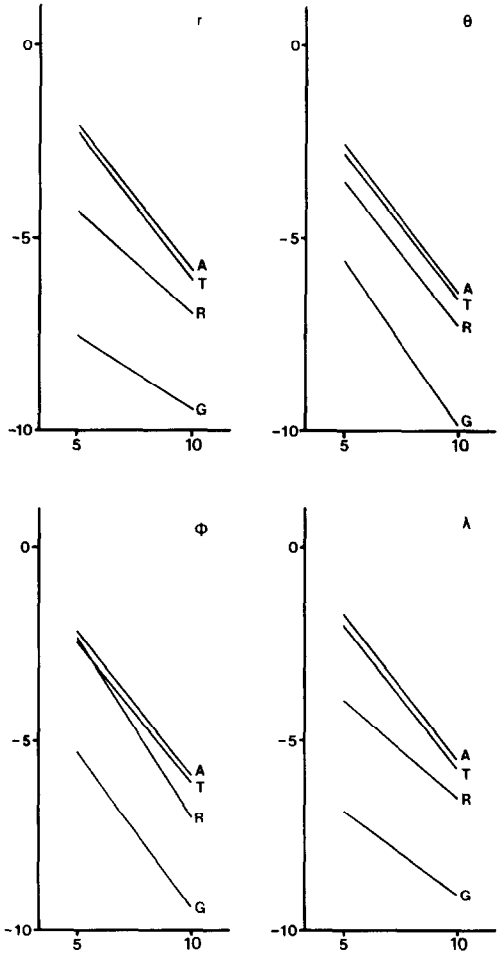

FIG. 2. Maximum error of each method in the four coordinates r, θ, ϕ and λ versus local error request for test two integrated to $t = 2000$. Logarithmic scales. A: Adams–Bashforth–Moulton, G: Gragg–Bulirsch–Stoer, R: Runge–Kutta, T: Taylor series. Note that the straight lines connect *only two data points*, being used solely to reveal the trend, and do *not* represent the functional form of the dependence.

tolerance set at the round-off error level of the computer. It should therefore be as accurate as the machine permits, and, even though it is difficult to verify, it does provide a uniform standard for the performance of the different methods. The behaviour of the errors during the evolution of the solutions gave additional, retrospective, support for the "true" orbits.

The two problems presented here were chosen from the set studied as being representative of the important difficulties. The first path considered is a retrograde orbit in the equatorial plane of the $a = 0.5$ Kerr hole. This spirals in very slowly from $r = 6$, then rapidly plunges most of the way to the event horizon at $r_h = 1.866$, being forced into direct motion by the "dragging of inertial frames" effect. This much-

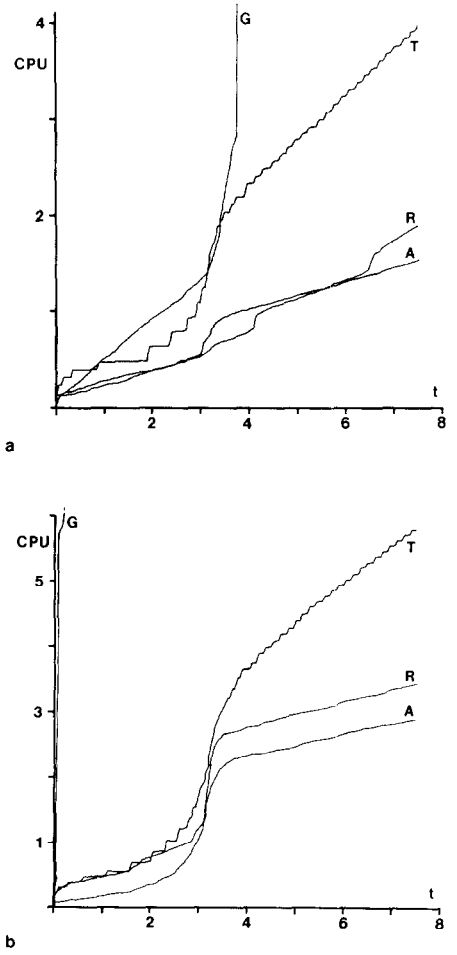

FIG. 3. Central processor time (IBM timer units, $\times 10^{-4}$) taken by each method for test one as a function of coordinate time ($\times 10^{-2}$). (a) Local error 10^{-5}, (b) local error 10^{-10}. A: Adams–Bashforth–Moulton, G: Gragg–Bulirsch–Stoer, R: Runge–Kutta, T: Taylor series.

discussed result of General Relativity can be found in, e.g., [23]. The particle hovers interminably just outside the point of no return. Proper time effectively stops advancing, but the azimuthal coordinate ϕ continues to increase at just the rate given by $\Omega = a/2r_h = 0.134$, which is the angular velocity of the hole. This nicely demonstrates the freezing of the motion to be expected when stepping in coordinate time.

The second orbit used is a bound orbit that neither falls in nor escapes, around an $a = 0.5$ hole. The path is started at $r = 6$ in the equatorial plane, and it loops around and out of this plane—a fully three-dimensional track.

These two test trajectories were run through each of the four integration methods twice, with local accuracy requests of 10^{-5} and 10^{-10}. The deviation from the

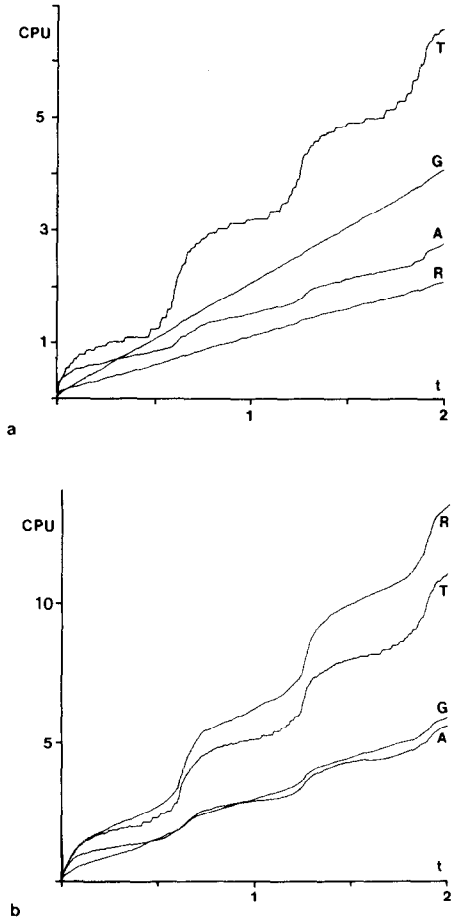

FIG. 4. Central processor time (IBM timer units, $\times 10^{-4}$) taken by each method for test two as a function of coordinate time ($\times 10^{-3}$). (a) Local error 10^{-5}, (b) local error 10^{-10}. A: Adams–Bashforth–Moulton, G: Gragg–Bulirsch–Stoer, R: Runge–Kutta, T: Taylor series.

comparison path was plotted as a function of time, in order to understand better the behaviour and the accuracy of the systems, up until $t = 400$ with output at intervals $\Delta t = 5$ for problem 1, and up until $t = 2000$ with output every $\Delta t = 10$ for problem 2. For problem 1, the curves either change very little, or deviate wildly, beyond $t = 400$, except for the extrapolation (G) routine, which failed at $t = 385$ (see later discussion). At the tighter error request, the G routine was stopped at $t = 190$ for using too much computer time.

Within these ranges, the maximum error in the four variables r, θ, ϕ and λ is plotted against the local error request in Figs. 1 and 2. This gives an indication of the accuracy of the integration technique in the region where it successfully handled the problem. It is very important to remember that there are *only two data points* for each line in these figures, and that these points have been connected by a straight line in order to reveal the trend more clearly. These straight lines do *not* represent the functional form of the dependence of global error on local error request.

Figures 3 and 4 show the central processor (CPU) time taken for the calculations by each routine, as a function of coordinate time t. The time units are "IBM 370 timer units," of which there are said to be 38,400 in one second. These curves show the variation of CPU requirement over different parts of the orbit. There has been no attempt at any normalisation related to the actual accuracy obtained.

5. Discussion

(a) *Test* 1

At the lower accuracy request, beyond $t = 400$, the Runge–Kutta (R) solution allowed the particle to bounce back out of the black hole, but this routine performed much better at higher accuracy. The G routine produced a very interesting (and undocumented) feature. It attempts to estimate the most efficient steplength for the next integration region by considering the behaviour of the approximations during the current region. This estimated value is often larger than the region itself. The ability to travel over large regions in one leap, which is one of the greatest advantages of extrapolation methods, is not used here because the solution is required at fixed, evenly spaced, and quite short, intervals. However, as proper time stopped and very little was happening, the G routine reduced its estimated timestep towards zero, finally setting it as identically zero, requesting an infinite number of sub-intervals, and giving up in disgust when the machine refused to allow such an unreasonable number. This fit of pique seems to be related to the development of mild stiffness.

An extrapolation code repeatedly integrates over the interval with successively smaller stepsizes. If one stepsize is outside the stability region, the next moves closer to or into this region, but takes more steps and so allows more growth of error. Thus stiffness, which reduces the size of the stability region, "chases" the program into smaller and smaller stepsizes, finally provoking it into surrender. This interpretation is supported by the behaviour of the steplength for the Adams–Bashforth–Moulton

(A) method. Since this steplength is adjusted solely by halving or doubling, it is possible by doubling to move from inside the region of relative stability to outside the region of absolute stability. This leads to rapid error growth and halving of the stepsize, and the process repeats. These rapid jiggles are plainly visible in the results. The stiffness must be mild, because the A method was more accurate, and much faster, than a specialised method which uses backward differencing to cope with the stiffness. (This stiff routine performed so badly overall that its results have not been included.)

The Taylor (T) and G methods are globally considerably more accurate for the same local error tolerance. The G method was actually *globally* within, or close to, the local error limit (until it failed as discussed above). The T routine is remarkably good, generally getting errors two orders of magnitude smaller than those of the A system (probably the most widely used), and taking only twice the computer time.

(b) *Test* 2

Much less of note happened during this test, as the orbit is smooth, and a reasonable distance away from the hole. At low tolerance, the R routine copes very well, although it cannot match the accuracy of the G method. At high tolerance, however, the G routine is ideal, being both fast and very accurate. No problems with stiffness here. The CPU time taken by the R routine increases out of all proportion to its gain in accuracy. The A routine performs as well as expected.

6. Conclusions

Principally, we require a certain accuracy over the entire integration for the smallest computer time. Acceptable accuracy is not obtained by specifying this limit for the local error control, and different methods get different global precision for the same quoted local precision.

The R technique is not sufficiently reliable for all sets of initial conditions, nor is it sufficiently fast at high accuracies. The first difficulty, though not the second, would probably be cured by a more sophisticated approach of the sort noted earlier. Such an improved routine would therefore be a "best buy" for quick, low-accuracy results. The T routine is distinctly slower than the A routine for only slightly better accuracy. It is, however, a very robust method which deserves more attention in differential equation studies. Note also, from Figs. 3 and 4, that it often extrapolated from known results at negligible further cost. The G method cannot cope with the slight stiffness which can arise, and so on the basis of these two typical tests, the A routine emerges as the favourite.

However, if we should be interested solely in astrophysical processes in the black hole exterior, then particles approaching the hole would be deemed captured, and the integration could be terminated before the G routine fails. A major reason for the G routine's poorer performance is the complexity of the derivative evaluations, since each step requires many more of them than used by the A method, and so the G

routine was severely handicapped by being forced to produce values at relatively very short intervals, ignoring an extrapolation method's ability to step accurately over much larger regions. Some tests were carried out with each method "given its head," and in these the G program became very competitive, although the large stepsizes made for a very angular "route map." The T program also improved its relative performance. The extrapolation method should be strongly considered, not only for its great accuracy—often better at the less stringent request than the other routines at the tighter constraint—but also because it requires much less storage than the A method, which has to pass the divided difference tables between steps. It might also be worth investigating the most efficient error-control strategy.

ACKNOWLEDGMENTS

I am grateful to the University of Cambridge Computer Laboratory, particularly Ron Laborde, Arthur Norman and Margaret Oakley, and to Guy Morgan for computer graphics. I wish to thank the referees for very helpful advice. I acknowledge an SRC studentship, and a NATO Postdoctoral Fellowship.

REFERENCES

1. B. CARTER, *in* "Black Holes" (C. DeWitt and B. S. DeWitt, Eds.), Gordon & Breach, New York, 1973.
2. B. CARTER, *in* "Proceedings, First Marcel Grossmann Meeting on General Relativity" (R. Ruffini, Ed.), North-Holland, Amsterdam, 1977.
3. N. A. SHARP, "Black Holes: The No Hair Theorems," unpublished essay, University of Cambridge, 1976.
4. N. A. SHARP, *Gen. Relativity Gravitation* **10** (1979), 659.
5. R. W. FULLER AND J. A. WHEELER, *Phys. Rev.* **128** (1962), 919.
6. G. H. BRIGMAN, *Tensor* **25** (1972), 267.
7. R. H. BOYER AND R. W. LINDQUIST, *J. Math. Phys.* **8** (1967), 265.
8. G. HALL AND J. M. WATT, "Modern Numerical Methods for Ordinary Differential Equations," Oxford Univ. Press, London/New York, 1976.
9. W. H. ENRIGHT AND T. E. HULL, *SIAM J. Numer. Anal.* **13** (1976), 944.
10. L. F. SHAMPINE, H. A. WATTS, AND S. M. DAVENPORT, *SIAM Rev.* **18** (1976), 376.
11. R. H. MERSON, *in* "Proceedings, Symposium on Data Processing," Weapons Research Establishment, Salisbury, S. Australia, 1957.
12. J. D. LAMBERT, "Computational Methods in Ordinary Differential Equations," Wiley, New York, 1973.
13. F. T. KROGH, *SIAM J. Numer. Anal.* **10** (1973), 949.
14. C. W. GEAR, "Numerical Initial Value Problems in Ordinary Differential Equations," Prentice–Hall, Englewood Cliffs, N.J., 1971.
15. D. BARTON, I. M. WILLERS, AND R. V. M. ZAHAR, *Comput. J.* **14** (1971), 243.
16. D. BARTON, I. M. WILLERS, AND R. V. M. ZAHAR, *in* "Mathematical Software" (J. R. Rice, Ed.), Academic Press, New York, 1971.
17. A. C. NORMAN, "Taylor User's Manual," University of Cambridge Computer Laboratory publication, 1973.
18. L. F. RICHARDSON, *Phil. Trans. Roy. Soc. London A* **210** (1910), 307.

19. L. F. RICHARDSON, *Phil. Trans. Roy. Soc. London A* **226** (1927), 299.
20. J. STOER, *in* "Proceedings, Conference on the Numerical Solution of Ordinary Differential Equations" (D. G. Bettis, Ed.), Lecture Notes in Mathematics No. 362, Springer-Verlag, Berlin/Heidelberg/New York, 1974.
21. R. BULIRSCH AND J. STOER, *Numer. Math.* **8** (1966), 1.
22. W. B. GRAGG, *SIAM J. Numer. Anal. Ser. B* **2** (1965), 384.
23. C. W. MISNER, K. S. THORNE, AND J. A. WHEELER, "Gravitation," Freeman, New York, 1973.