IMPLICIT SINGLE-SEQUENCE METHODS FOR INTEGRATING ORBITS

EDGAR EVERHART

Dept. of Physics and Astronomy, University of Denver, Colo., U.S.A.

(Received 24 May, 1973)

Abstract. The solutions of $\ddot{x} = F(x, t)$, and also $\dot{x} = F(x, t)$, are developed in truncated series in time t whose coefficients are found empirically. The series ending in the t^6 term yields a position at a final prechosen time that is accurate through 9th order in the sequence size. This is achieved by using Gauss-Radau and Gauss-Lobatto spacings for the several substeps within each sequence. This time-series method is the same in principle as implicit Runge-Kutta methods, and the present algorithm generates coefficients for families of implicit Runge-Kutta forms, including some not described previously. In some orders these methods are unconditionally stable (A-stable). In the time-series formulation the implicit system converges rapidly. For integrating a test orbit the method is found to be about twice as fast as high-order explicit Runge-Kutta-Nyström-Fehlberg methods at the same accuracies. Both the Cowell and the Encke equations are solved for the test orbit, the latter being 35% faster. It is shown that the Encke equations are particularly well-adapted to treating close encounters when used with a single-sequence integrator (such as this one) provided that the reference orbit is re-initialized at the start of each sequence. This use of Encke equations is compared with the use of regularized Cowell equations.

1. Introduction

The method of integrating $\ddot{x} = F(x, t)$ and $\dot{x} = F(x, t)$ described here was developed from time-series expansions as a practical and fast single-sequence integrator. Later this was found to be related to the implicit Runge-Kutta method, although the formulations and algorithms of the two methods are entirely different.

From the time-series approach we are concerned with methods wherein the series is fitted to the function F at several computed points, and in particular to those cases where these points are not evenly spaced. Such methods have been developed by Wielen (1967) and Aarseth (1972). They fit an empirical polynomial in time through the forces found at several previous steps which can be unevenly spaced. Integrating this fitted curve they predict the position at the next step, determine the force there, and then correct this position. Theirs is a multi-step method with variable step size.

Our method is similar in that we also integrate a time series found by fitting an empirical curve through forces at several unevenly-spaced points, but it is different in several respects: (1) It is self-starting. (2) All the points are at the current position or in advance of it. (3) There are several forward substeps taken during an integration sequence before the final corrected position is found. It is a single-sequence method.

In the above respects our method has a pattern like that of Runge-Kutta integrations. In fact, the most extensive comparisons to be presented are with the high-order explicit Runge-Kutta-Nyström methods developed by Fehlberg (1972).

An attractive aspect of our method is that one can obtain accuracies several orders higher than would be expected from the order of the fitting polynomial. This is achiev-

Celestial Mechanics **10** (1974) 35–55.

ed by adopting Gauss-Radau or Gauss-Lobatto spacings for the substeps. As in Butcher's (1964) implicit Runge-Kutta method and Beaudet's (1972) multi-off-grid method, we apply the principles of Gaussian quadratures to integrating differential equations.

Section 2 below derives the equations fundamental to the method, and Section 3 describes the integration procedure. Section 4 discusses Radau and Lobatto spacings, giving a detailed example of their applicability. There is also a discussion of why Gauss-Legendre spacings are not as useful here. Section 5 shows that the same procedures apply to both first- and second-order differential equations. An analysis shows that there is unconditional stability (A-stability) in the Lobatto cases when solving $\dot{x}=F(x, t)$. Section 6 considers the correspondence with implicit Runge-Kutta methods, showing that the present algorithm develops the coefficients for four families of such methods. One of these families is A-stable and another is the same as one described by Butcher (1964). Section 7 describes numerical tests and comparisons with other methods of orbit integration. Finally, Section 7 points out that single-sequence integrators are particularly suitable for solving the Encke orbit equations because they allow rectification at the start of every sequence. The advantage shows in the case of orbits involving close encounters, and this is compared with the alternative of using regularized Cowell equations.

2. Fundamental Equations

The orbit equations of celestial mechanics are of the form

$$\ddot{x} = F(x, y, z, x_i, y_i, z_i, x_j, y_j, z_j, \ldots, t), \tag{1}$$

where the function F, which may be called the force, depends on time t and the position x, y, z of the body whose path is to be integrated, as well as the positions of other bodies, as identified by subscripts $i, j, \ldots$. There are 3 such equations for each body. It is sufficient to treat the solution of the class IIS equation (S for special, since $\dot{x}$ is absent)

$$\ddot{x} = F(x, t), \tag{2}$$

since the extension to any number of simultaneous equations of the form of Equation (1) is a very well known procedure. The class I equation is treated in Section 5.

At the start of a sequence we reset time $t_1=0$, and the initial position x_1, velocity $\dot{x}_1$, and force F_1 are known. A time-series expansion of F about time zero is

$$\ddot{x} = F = F_1 + A_1 t + A_2 t^2 + A_3 t^3 + \cdots + A_N t^N. \tag{3}$$

Integrating Equation (3) one has

$$x = x_1 + \dot{x}_1 t + F_1 t^2/2 + A_1 t^3/6 + \cdots + A_N t^{N+2}/\big((N+1)\,(N+2)\big), \tag{4}$$

$$\dot{x} = \dot{x}_1 + F_1 t + A_1 t^2/2 + A_2 t^3/3 + \cdots + A_N t^{N+1}/(N+1). \tag{5}$$

The truncated series of Equation (3) is not a Taylor series because the coefficients A

are not chosen to represent F as well as possible for all values of t, but instead, as will be shown in Section 4, they are chosen so that the truncated expressions in Equations (4) and (5) calculate x and $\dot{x}$ as accurately as possible at a particular prechosen time T.

The only unknowns in the solution are the A-values. The overall pattern for finding these is that of a single-sequence method, somewhat analogous to Runge-Kutta procedures. One explores the value of the function F at several unequally-spaced substeps $t_2, t_3, t_4, \ldots$ ahead of $t_1 = 0$. From this information the values A_n are developed such that Equations (4) and (5) are accurate at time T, which is the end of the sequence.

Let $F_n(x_n, t_n)$ be the force at one of the substeps where the time is t_n and the position is $x_n(t_n)$. To develop formulas for relating A-values to these F-values we need the auxiliary expansion

$$F = F_1 + \alpha_1 t + \alpha_2 t(t - t_2) + \alpha_3 t(t - t_2)(t - t_3) + \cdots . \tag{6}$$

This series and the expressions below are all terminated in a manner consistent with the number of terms kept in Equation (3). Equation (6) truncates short at each of the times t_n. Thus $F_2 = F_1 + \alpha_1 t_2$, and $F_3 = F_1 + \alpha_1 t_3 + \alpha_2 t_3 (t_3 - t_2)$, etc. Using the abbreviation $t_{nj} = t_n - t_j$, one finds

$$\begin{aligned}
\alpha_1 &= (F_2 - F_1)/t_2 , \\
\alpha_2 &= \big((F_3 - F_1)/t_3 - \alpha_1\big)/t_{32} , \\
\alpha_3 &= \big(\big((F_4 - F_1)/t_4 - \alpha_1\big)/t_{42} - \alpha_2\big)/t_{43} , \\
\alpha_4 &= \big(\big(\big((F_5 - F_1)/t_5 - \alpha_1\big)/t_{52} - \alpha_2\big)/t_{53} - \alpha_3\big)/t_{54} .
\end{aligned} \tag{7}$$

Of course the α-values are divided differences. For example, $\alpha_3 = [0, t_2, t_3, t_4]$. The standard divided-difference notation would be cumbersome here, and the compact α-notation is used instead. Each new force adds another α-value without changing any of the previous α's. The relationship between the α-values and the A-values is found by equating like powers of t in Equations (3) and (6). The result is

$$\begin{aligned}
A_1 &= \alpha_1 + (-t_2)\,\alpha_2 + (t_2 t_3)\,\alpha_3 + \cdots = c_{11}\alpha_1 + c_{21}\alpha_2 + c_{31}\alpha_3 + \cdots \\
A_2 &= \alpha_2 + (-t_2 - t_3)\,\alpha_3 + \cdots = c_{22}\alpha_2 + c_{32}\alpha_3 + \cdots \\
A_3 &= \alpha_3 + \cdots = c_{33}\alpha_3 + \cdots
\end{aligned} \tag{8}$$

The c-coefficients have the same recursion relationships as Stirling numbers of the first kind. Thus,

$$\begin{aligned}
c_{ii} &= 1 , \\
c_{i1} &= -t_i c_{i-1,1} , && i > 1 , \\
c_{ij} &= c_{i-1,j-1} - t_i c_{i-1,j} , && 1 < j < i .
\end{aligned} \tag{9}$$

They are also the coefficients of a polynomial expressed in terms of its roots. Thus a cubic with roots t_2, t_3, t_4 is

$$(-t_2 t_3 t_4) + (t_2 t_3 + t_3 t_4 + t_4 t_2)\, t + (-t_2 - t_3 - t_4)\, t^2 + (1)\, t^3 = 0, \tag{10}$$

where the quantities in parentheses are c_{41}, c_{42}, c_{43}, and c_{44}.

In the integration to be described one α-value is found with each substep in a pass through a sequence. As each is found another A-value is known from Equation (8) and each previous A-value is corrected.

3. Integration Algorithm

3.1. Predictors and correctors

To illustrate the procedure we consider integrating a sequence of 3 substeps at times t_2, t_3, t_4 which, in general, are not uniformly spaced, and where the end of the sequence T need not coincide with t_4. At the starting time $t_1 = 0$ we know the initial values x_1, $\dot{x}_1$, and F_1. In this class IIS procedure there are three predictor equations, one for each substep,

$$x_2 = x_1 + \dot{x}_1 t_2 + F_1 t_2^2/2 + [A_1 t_2^3/6 + A_2 t_2^4/12 + A_3 t_2^5/20], \tag{11}$$

$$x_3 = x_1 + \dot{x}_1 t_3 + F_1 t_3^2/2 + A_1 t_3^3/6 + [A_2 t_3^4/12 + A_3 t_3^5/20], \tag{12}$$

$$x_4 = x_1 + \dot{x}_1 t_4 + F_1 t_4^2/2 + A_1 t_4^3/6 + A_2 t_4^4/12 + [A_3 t_4^5/20], \tag{13}$$

and two corrector equations for finding position and velocity at the end of the sequence

$$x(T) = x_1 + \dot{x}_1 T + F_1 T^2/2 + A_1 T^3/6 + A_2 T^4/12 + A_3 T^5/20, \tag{14}$$

$$\dot{x}(T) = \dot{x}_1 + F_1 T + A_1 T^2/2 + A_2 T^3/3 + A_3 T^4/4. \tag{15}$$

This system is implicit because the terms in square brackets are not known when they are first needed. However, fairly good estimates can be made, because the α-values change slowly from one sequence to the next. Thus if we know α_1 from the previous sequence and the second previous sequence, we can predict α_1 for the current sequence fairly well. The sequence lengths T may all be different, but it is not difficult to fit a straight line through 2 back points (or a parabola through 3 back points) and extrapolate to the current sequence. In almost all cases, including the present 3-substep example, a linear extrapolation suffices for α_1. For α_2 and α_3 one does no fitting; it is enough simply to use the corresponding α-value from the previous sequence in the current sequence until a new α-value is found. Approximate values for A_1, A_2, and A_3 are found with the aid of Equation (8) from these estimated α-values. Thus in starting the iteration, reasonable values of the bracketed terms are in hand, and a fair initial prediction of x_2 is obtained from Equation (11).

Using this value of x_2 one calculates $F_2(x_2, t_2)$ and a new value of α_1 from Equation (7). At this time A_1 can be improved as in Equation (8), discarding the contribution from the extrapolated value of α_1 and including the contribution from this newly-found value of α_1.

Next x_3 is found from Equation (12), using this considerably improved value of A_1 and, as before, the estimated values of A_2 and A_3. Finding F_3 from x_3, one obtains α_2 with Equation (7). Then A_2 is improved as in Equation (8), using this new α_2 instead of the old one, and retaining still the estimated value of α_3. In like manner A_1 is improved.

The same process with the next substep finds x_4, F_4, α_3, and new value for A_3, and still further-improved values of A_2 and A_1.

3.2. Iteration

Usually it is profitable to iterate, passing through the sequence a second time. Of course the α- and A-values used in the predictors are far more accurate than they were in the first pass. At every substep all the A-values are immediately improved as each better α-value becomes available. Only rarely, as in an 11th order example to be discussed later, is it worthwhile to make 3 passes through the sequence.

Finally, new values of position and velocity at the end of the sequence are found using Equations (14) and (15).

The use of information from previous sequences in the first pass through the sequence makes possible a rapid solution of an implicit system. However, in the second (and final) pass one uses only values within the current sequence, and the method is basically of a single-sequence nature.

3.3. Starting

In the case of the very first sequence, where extrapolated α-values from previous sequences are not available, these estimated α-values are taken as zero, which makes the bracketed terms in Equations (11)–(13) zero initially. High accuracy may be maintained by making 4 passes through the sequence in this case, or by taking a somewhat smaller sequence length T to start the integration.

3.4. Sequence size control

Sequence lengths T are controlled by monitoring the size of the highest-order term in the x-expansion. A control parameter C is established that is the desired size of this last term, which is $A_3T^5/20$ in the case above. Let $H=(|A_{3x}|+|A_{3y}|+|A_{3z}|)/20$, assuming a three-dimensional integration. The next sequence should be chosen of length $T=(C/H)^{1/5}$ to achieve such a control. The subscript 3, the divisor 20, and the exponent 1/5 are changed appropriately in other cases.

3.5. Preliminary tests

Later in this paper we describe the test problem and explain how the error is evaluated. However, for now, regard Figure 1 as plotting the relative integration error in position vs computation time for an integration requiring very many sequences. A line is traced out on this figure using a range of controls C on the sizes T of the sequences.

Line (1) on the figure is the result of a test integration where the substeps were taken equally spaced within the sequence, with $t_2=T/3$, $t_3=2T/3$, $t_4=T$. For line (1) only a single pass was made through each sequence and no use was made of estimated values from previous sequences, that is, the bracketed terms in Equations (11)–(13) were deleted. The slope of the line is a rough measure of integration order. It is desirable for the slope to be high and for the line to lie as far as possible towards the lower left part of the figure. Evidently line (1) shows a poor result.

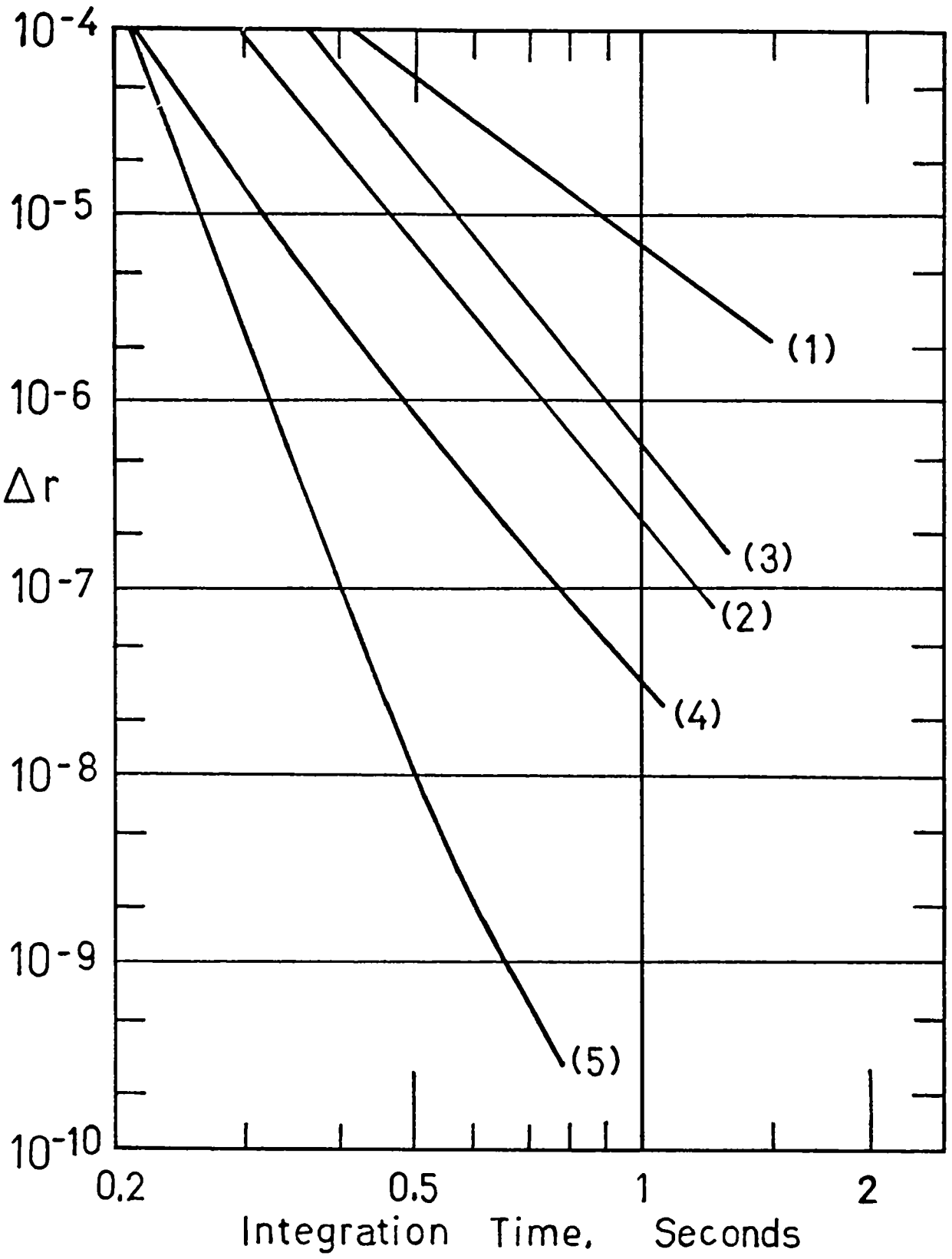

Fig. 1. Relative error Δr in position vs computing time for a test orbit using present method with 3 substeps per sequence. (1) Uniform substeps, no alpha prediction, single pass. (2) Uniform substeps, with alpha prediction, single pass. (3) Uniform substeps, with alpha prediction, double pass. (4) Radau spacings, with alpha prediction, single pass. (5) Radau spacings, with alpha prediction, double pass.

Line (2) on the figure is the same single-pass procedure as in line (1), except that full use was made of estimated values of the bracketed terms, extrapolating the α-values from previous sequences. This takes almost no extra computation and improves the accuracy considerable.

Line (3) is the same as line (2), except that there are two passes through each sequence. The second pass adds 40% to the computation time per sequence, and we see that line (3) is not as good as line (2). This means that sufficiently high accuracy has been reached in the first pass, and the second pass, though it improves the accuracy per sequence, is not worth the additional computation time. In other words, at the same accuracy it is more efficient to take slightly smaller sequences in a single pass than to take larger sequences and pass twice through each. With even spacings of the substeps in this 5th-order integration, one pass per sequence is optimum.

However, it is possible to improve the results in this 3-substep case remarkably, as in lines (4) and (5) of Figure 1, by choosing a special spacing for the substeps, as described

the next section. A single pass through each sequence then results in line (4). Two passes, which together take 140% as long as a single pass, improve the convergence. There is a large net gain, as seen by line (5), which indicates 7th-order accuracy. When 3 passes were tried (taking 180% of the computation time per sequence of the single pass time), the result was not as good as line (5), the slight additional accuracy not being worth the extra computation time. Thus two passes per sequence are optimum with the special spacings in this case.

4. Gauss-Radau and Gauss-Lobatto Spacings

As in quadratures, much higher orders of accuracy may be obtained with a given number of function evaluations by adopting Gaussian spacings for the substeps. When the first force evaluation is at the fixed point $t=0$ and the last evaluation is at the optimum position, then the appropriate spacings are Gauss-Radau. These correspond to odd orders of integration accuracy. When the first evaluation is at $t=0$ and the last is at $t=T$, then Gauss-Lobatto spacings are appropriate, and the integration order is even. The classical justification for this is well known for quadratures, but is certainly less familiar for integrating second-order differential equations. Such a case is discussed next. A side light is that the proof illustrates the connection between the roots of Hilbert matrices and Legendre polynomials.

4.1. Seventh-order example integrating $\ddot{x}=F(x,t)$

The example of the preceding section, Equations (11)–(14), appears to be of the 5th order in t. Without adding more terms we shall find spacings for t_2, t_3, t_4 such that the results for both x and $\dot{x}$ are accurate through 7th order.

Suppose that two more evaluations of force were to be made at any two additional times t_5 and t_6 and that α_4 and α_5 were determined from these. We can calculate the improvements in x and $\dot{x}$ that would result from this additional information. Two additional orders of accuracy are achieved by finding values for t_2, t_3, t_4 such that these improvements are zero. With these spacings there is no need to carry out the two additional force evaluations.

If α_4 and α_5 were known, we could calculate, using Equations (8), new values A'_4 and A'_5 and also improve the three A-values in hand to get A'_3, A'_2, and A'_1. Here primes identify the 'improved' values.

The improved x-value would be described by a series in primed A-values ending with the term $A'_5 T^7/42$. Subtracting the old x-value (as in Equation (14)) from this improved value we find the improvement Δx in position. Thus,

$$\Delta x = (A'_1 - A_1)\, T^3/6 + (A'_2 - A_2)\, T^4/12 + (A'_3 - A_3)\, T^5/20 + A'_4 T^6/30 + A'_5 T^7/42 . \tag{16}$$

This equation is to undergo several steps of manipulation. First, using Equations (8) suitably extended, it is expressed in terms of the α-values and $c_{41}, \ldots, c_{43}, c_{51}, \ldots, c_{54}$.

Then the recursion relationships, Equations (9), are used to reduce the terms in $c_{51}, \ldots c_{54}$ to expressions involving only c_{41}, c_{42}, c_{43}. A third step introduces dimensionle times

$$h_2 = t_2/T, \qquad h_3 = t_3/T, \quad \ldots, \tag{17}$$

and a fourth step introduces dimensionless c-values indentified by primes. Thus $c_{41} = -t_2 t_3 t_4$, then $c'_{41} = -h_2 h_3 h_4$ with the same recursion relationships. The resu of all the manipulation is that Equation (16) can be written

$$\begin{aligned} \Delta x = (\alpha_4 - t_5 \alpha_5)\, T^6 [c'_{41}/6 + c'_{42}/12 + c'_{43}/20 + 1/30] \\ + \alpha_5 T^7 [c'_{41}/12 + c'_{42}/20 + c'_{43}/30 + 1/42]. \end{aligned}$$

One notes that Δx can be set equal to zero regardless of the unknown values of α_4, α_5 and t_5, Two condition equations are

$$c'_{41}/6 + c'_{42}/12 + c'_{43}/20 + 1/30 = 0, \tag{18}$$
$$c'_{41}/12 + c'_{42}/20 + c'_{43}/30 + 1/42 = 0. \tag{19}$$

A similar development shows that the improvement in velocity $\Delta \dot{x}$ is zero to three more orders with the three conditions

$$c'_{41}/2 + c'_{42}/3 + c'_{43}/4 + 1/5 = 0, \tag{20}$$
$$c'_{41}/3 + c'_{42}/4 + c'_{43}/5 + 1/6 = 0. \tag{21}$$
$$c'_{41}/4 + c'_{42}/5 + c'_{43}/6 + 1/7 = 0. \tag{22}$$

Equation (18) is redundant, being the difference between Equations (21) and (22), and Equation (19) is likewise redundant. The condition on the c'-values is

$$\begin{pmatrix} 1/2 & 1/3 & 1/4 \\ 1/3 & 1/4 & 1/5 \\ 1/4 & 1/5 & 1/6 \end{pmatrix} \begin{pmatrix} c'_{41} \\ c'_{42} \\ c'_{43} \end{pmatrix} = \begin{pmatrix} -1/5 \\ -1/6 \\ -1/7 \end{pmatrix}, \tag{23}$$

which is recognized as one of the Hilbert matrix forms. The solution is

$$\begin{aligned} c'_{41} &= -4/35 = -h_2 h_3 h_4, \\ c'_{42} &= \quad 6/7 \quad = h_2 h_3 + h_3 h_4 + h_2 h_4, \\ c'_{43} &= -12/7 = -h_2 - h_3 - h_4, \end{aligned} \tag{24}$$

and the polynomial in the form of Equation (10) whose roots give the spacings is

$$(-4/35) + (6/7)\, h + (-12/7)\, h^2 + h^3 = 0. \tag{25}$$

The roots are

$$\begin{aligned} h_2 &= t_2/T = 0.21234\ 05382\ 39\ldots \\ h_3 &= t_3/T = 0.59053\ 31355\ 59\ldots \\ h_4 &= t_4/T = 0.91141\ 20404\ 88\ldots. \end{aligned} \tag{26}$$

ith these spacings the integration should be accurate through 7th order for both x and $\dot{x}$. The practical demonstration is seen in Figure 1. Line (4) is the result of adopting these spacings in a single pass through each sequence. However, with a double pass through each sequence the accuracy at constant computation time is improved by another factor of 100, as seen by line (5) on the figure. The steepness of the line indicates that 7th order is achieved in practice. The computation time per sequence is the same for lines (3) and (5). As mentioned already, two passes/sequence is optimum in this case.

For Radau quadratures Abramowitz and Stegun (1964) give an equation involving Legendre polynomials, $P_4(u)+P_3(u)=u+1$, whose roots are the Gauss-Radau spacings (for the appropriate order to compare with our case). Divided by $u+1$ this is a polynomial in u, and the roots are within the range -1 to $+1$. Since the range in h is 0 to $+1$, one must substitute $u=2h-1$. When this is done the resulting polynomial in h is the same as Equation (25); hence Equations (26) are the Radau spacings. The demonstration that even orders of integration require Lobatto spacings is similar.

Although Hilbert matrices are ill-conditioned, their roots are rational fractions, and no difficulty was experienced in finding these fractions exactly. Indeed, as seen above, an alternative way of finding these fractions exactly is through the coefficients of Legendre polynomials.

4.2. Tabulated spacings

Table I gives the spacings h for integration of $\ddot{x}=F(x, t)$ from orders 5th through 15th. In certain cases the polynomial in the form of Equation (25) can be solved in terms of quadratic surds, and the exact spacing can be given. Much more extensive tables of Radau and Lobatto spacings are given by Stroud and Secrest (1966), except in the range -1 to $+1$, instead of the range 0 to $+1$ as used here.

4.3. Practical considerations

The odd orders of integration are more simply programmed than the even orders. An 8th-order method with Lobatto spacings and a 9th-order method with Radau spacings both require four α-determinations in each of the two passes through a sequence. These α-determinations and the attendant updating of the A-values become more complicated in quantum jumps as another α-value is added to the sequence. One might as well use the higher order for a given number of α-determinations, and this recommends the odd-order cases. In all cases, even and odd, the number of force calculations in a double pass through a sequence is the same as the order of the integration.

When looking for programming errors with this method it is useful to examine the α-values. The sequence $\alpha_1, \alpha_2, \alpha_3, \ldots$ should generally be decreasing in size. An incorrect equation or procedural error causes large erratic fluctuations in such a sequence. In practical programming, one would nest the series in Equations (11)–(15).

4.4. The Gauss-Legendre case

By avoiding the force evaluation at $t=0$, one could adopt Gauss-Legendre spacings

TABLE I

Spacings for substeps $h_n = t_n/T$ within sequences of length T for integration with accuracies through the orders given. All these apply to integration of $\dot{x} = F(x, t)$ and $\dot{x} = F(t)$, and for orders of 5th or higher they apply to $\ddot{x} = F(x, t)$. Where they are known, analytic expression are given in brackets. In every case $h_1 = 0$. Except that the numbers here are fitted to the range 0 to +1, these spacings are the same as those for Gauss-Radau and Gauss-Lobatto quadratures

Order	h_n
3	$h_2 = 0.66667$
4	$h_2 = 0.5$ $h_3 = 1.$
5	$[h_2 h_3 = (6 \pm 6^{1/2})/10]$ $h_2 = 0.35505\ 10257$ $h_3 = 0.84494\ 89743$
6	$[h_2 h_3 = (5 \pm 5^{1/2})/10]$ $h_2 = 0.27639\ 32022\ 50$ $h_3 = 0.72360\ 67977\ 50$ $h_4 = 1.$
7	$h_2 = 0.21234\ 05382\ 39153$ $h_3 = 0.59053\ 31355\ 59265$ $h_4 = 0.91141\ 20404\ 87296$
8	$[h_2 h_4 = (7 \pm 21^{1/2})/14]$ $h_2 = 0.17267\ 31646\ 46011$ $h_3 = 0.5$ $h_4 = 0.82732\ 68353\ 53989$ $h_5 = 1.$
9	$h_2 = 0.13975\ 98643\ 43780\ 55$ $h_3 = 0.41640\ 95676\ 31083\ 18$ $h_4 = 0.72315\ 69863\ 61876\ 17$ $h_5 = 0.94289\ 58038\ 85482\ 32$
10	$[h_2, h_3, h_4, h_5 = 1/2 \pm ((7 \pm 7^{1/2}2)/84)^{1/2}]$ $h_2 = 0.11747\ 23380\ 35267\ 65$ $h_3 = 0.35738\ 42417\ 59677\ 45$ $h_4 = 0.64261\ 57582\ 40322\ 55$ $h_5 = 0.88252\ 76619\ 64732\ 35$ $h_6 = 1.$
11	$h_2 = 0.09853\ 50857\ 98826\ 42612$ $h_3 = 0.30453\ 57266\ 46363\ 90549$ $h_4 = 0.56202\ 51897\ 52613\ 85599$ $h_5 = 0.80198\ 65821\ 26391\ 82746$ $h_6 = 0.96019\ 01429\ 48531\ 25766$
12	$[h_2, h_3, h_5, h_6 = 1/2 \pm ((15 \pm 15^{1/2}2)/132)^{1/2}]$ $h_2 = 0.08488\ 80518\ 60716\ 53506$ $h_3 = 0.26557\ 56032\ 64642\ 89310$ $h_4 = 0.5$ $h_5 = 0.73442\ 43967\ 35357\ 10690$ $h_6 = 0.91511\ 19481\ 39283\ 46494$ $h_7 = 1.$

Table I (Continued)

Order	h_n
13	$h_2 = 0.07305\ 43286\ 80258\ 38515$
	$h_3 = 0.23076\ 61379\ 69945\ 49908$
	$h_4 = 0.44132\ 84812\ 28449\ 86792$
	$h_5 = 0.66301\ 53097\ 18845\ 70090$
	$h_6 = 0.85192\ 14003\ 31515\ 70815$
	$h_7 = 0.97068\ 35728\ 40215\ 10803$
14	$h_2 = 0.06412\ 99257\ 45196\ 69233\ 12771$
	$h_3 = 0.20414\ 99092\ 83428\ 84892\ 77446$
	$h_4 = 0.39535\ 03910\ 48760\ 56561\ 56714$
	$h_5 = 0.60464\ 96089\ 51239\ 43438\ 43286$
	$h_6 = 0.79585\ 00907\ 16571\ 15107\ 22554$
	$h_7 = 0.93587\ 00742\ 54803\ 30766\ 87229$
	$h_8 = 1.$
15	$h_2 = 0.05626\ 25605\ 26922\ 14646\ 56522$
	$h_3 = 0.18024\ 06917\ 36892\ 36498\ 75799$
	$h_4 = 0.35262\ 47171\ 13169\ 63737\ 39078$
	$h_5 = 0.54715\ 36263\ 30555\ 38300\ 14486$
	$h_6 = 0.73421\ 01772\ 15410\ 53152\ 32106$
	$h_7 = 0.88532\ 09468\ 39095\ 76809\ 03598$
	$h_8 = 0.97752\ 06135\ 61287\ 50189\ 11745$

and, in principle, achieve a given order of accuracy with one less force evaluation. The disadvantage is that F_1 would not appear as an explicit factor in Equations (11)–(15), its place being taken by an implicit quantity A_0, whose value depends on the forces at every substep. This would enlarge by one low-order term each of the bracketed expressions that are subject to iterative improvement in the predictors. The accuracy of the predictors would be decreased. The additional force evaluation at $t=0$ (which is not repeated in the iteration) avoids this difficulty. Butcher (1964) discusses Gauss-Legendre spacings in comparison to Gauss-Radau in his implicit Runge-Kutta methods, and he also points out the advantage of the latter in reducing the number of implicit terms for a given accuracy.

5. First-Order Differential Equation and Stability

Application to solve $\dot{x}=F(x, t)$, the class I equation, is illustrated in a 3-substep case that permits 7th-order integration. The set of predictor equations is

$$\begin{aligned} x_2 &= x_1 + F_1 t_2 + [A_1 t_2^2/2 + A_2 t_2^3/3 + A_3 t_2^4/4], \\ x_3 &= x_1 + F_1 t_3 + A_1 t_3^2/2 + [A_2 t_3^3/3 + A_3 t_3^4/4], \\ x_4 &= x_1 + F_1 t_4 + A_1 t_4^2/2 + A_2 t_4^3/3 + [A_3 t_4^4/4], \end{aligned} \tag{27}$$

and the final position at time T is obtained with an error $O(T^8)$ from

$$x(T) = x_1 + F_1 T + A_1 T^2/2 + A_2 T^3/3 + A_3 T^4/4. \tag{28}$$

The integration procedure is the same as that described in Section 3, and the spacings for 7th order should be used in Table I.

With the test equation $\dot{x}=\lambda x$, a stability analyis was made, adapting the methods in Section 3.6.2. of Lapidus and Seinfeld (1971). Assuming iteration to convergence our 4th-order method yields $x(T)=x_1(1+\lambda T/2+(\lambda T)^2/12)/(1-\lambda T/2+(\lambda T)^2/12)$. The fraction is the (2, 2) Padé approximant to $e^{\lambda T}$, as listed in Table 3.2 by Lapidus and Seinfeld. The characteristic root is less than unit magnitude for all negative λT, and the method is unconditionally stable (A-stable). The 6th-order gave the (3, 3) Padé approximant, again with the same stability. Thus the even-order methods (Lobatto class I) appear to be A-stable. Tests with the 3rd- and 5th-order versions developed the (2, 1) and (3, 2) Padé approximants, and the 7th-order version would be expected to yield the (4, 3) Padé approximant. The regions of stability are of the form $0>\lambda T>S$, where $S=-6$, -11.84, and -19.1, respectively, in the 3 cases. Evidently the Radau class I methods are conditionally stable, but with very large regions for stability, particularly in the higher orders. Stability in the class IIS differential equation has not been studied.

Note added in proof. Preliminary tests with standard test equations of class I show a favorable comparison with other single-sequence methods. However, when the problem can be solved directly as class II without reducing to first order, the results are up to 10^6 times more accurate at the same number of function evaluations. For the class II equation it is profitable to use integration to orders at least as high as 15th.

The method of this paper is found to be adapted to solving any number of simultaneous 2nd-order differential equations of the form $\ddot{x}=F(\dot{x}, x, t)$ without reduction to first order. The modification involves adding velocity predictors to the algorithm as in Equations (11)–(15) and using these in the evaluation of the function F at each substep. Practical tests are most encouraging. Apparently the method can be extended to solving simultaneous nonlinear differential equations of any order with no restrictions on the lower-order derivatives appearing in the force function.

6. Implicit Runge-Kutta Methods

We show how the present algorithm generates coefficients for implicit Runge-Kutta integrators.

6.1. Class I families

Butcher (1964) described several families of implicit Runge-Kutta procedures. For his 5th-order Radau version (his type I), he found $F_1(x_1, 0)$, $F_2(x_2, h_2 T)$ and $F_3(x_3, h_3 T)$, where

$$x_2 = x_1 + T(a_{21}F_1 + [a_{22}F_2 + a_{23}F_3]),$$
$$x_3 = x_1 + T(a_{31}F_1 + a_{32}F_2 + [a_{33}F_3]). \tag{29}$$

Then

$$x(T) = x_1 + T(w_1F_1 + w_2F_2 + w_3F_3)$$

determines the final position. These implicit equations require iterative solution. (The explicit Runge-Kutta version would delete the bracketed terms, but then is only of 3rd order.)

The same form can be developed from our 5th-order Radau case for $\dot{x}=F(x, t)$. Using Equations (7) and (8), one evaluates $F_1+A_1T/2+A_2T^2/3$ in terms of F_1, F_2, F_3 and equates this to $w_1F_1+w_2F_2+w_3F_3$. This determines the w-values. Similar evaluations of $h_nF_1+h_n^2A_1T/2+h_n^3A_2T^2/3$ are set equal to $a_{n1}F_1+a_{n2}F_2+a_{n3}F_3$ to find the a-matrix. The coefficients so found for this 5th-order case are the same as those already given by Butcher (1964). For this 'type I' he lists members up through 5th order for this Radau class I family. The next member is of 7th order, and the coefficients (developed by the time-series algorithm described above) are given here as the second set in Table II. This Radau class I family has a large region for stability, as noted already. High order members of this family have been tabulated by Glasmacher and Sommer (1966).

When applied to the Lobatto class I method of 6th order, the same process generates the implicit Runge-Kutta coefficients listed as the first set in Table II. Besides this example, we have examined the coefficients of the 4th- and 8th-order members. Unlike Butcher's Lobatto family, which is different in form, all members of the present Lobatto class I family are A-stable. This is pointed out by Chipman (1971) and Ehle (1969), who have already discussed the first entry in Table II.

The implicit Legendre family of Butcher is also A-stable and this family is extended by the work of Glasmacher and Sommer (1966). We did not investigate Gauss-Legendre spacings for the reasons discussed in Section 4.4.

6.2. Class IIS families

An analogous procedure generates implicit Runge-Kutta coefficients for solving the special 2nd-order differential equation directly. The forces $F_n(x_n, h_nT)$ are evaluated at

$$x_n = x_1 + \dot{x}_1 h_n T + T^2(a_{n1}F_1 + a_{n2}F_2 + \cdots).$$

Then

$$x(T) = x_1 + \dot{x}_1 T + T^2(w_1F_1 + w_2F_2 + \cdots) \tag{30}$$

and

$$\dot{x}(T) = \dot{x}_1 + T(u_1F_1 + u_2F_2 + \cdots)$$

determine the final position and velocity. The coefficients a, w, and u are found by comparing these equations with Equations (11)–(15). The lowest order of the Lobatto class IIS family is the 6th-order member shown in Table II. For the Radau class IIS family there is a 5th-order member, but the 7th-order member is shown in Table II. The 8th- or 9th-order cases of all four families, Lobatto and Radau, class I and class IIS, were also worked out in testing the program that generates the coefficients.

6.3. Discussion

A by-product of this comparison is a criterion for choosing sequence sizes for implicit Runge-Kutta integrators. For example, in the 6th-order Lobatto class I case one

TABLE II

Coefficients for implicit Runge-Kutta integrators. The 6th- or 7th- order member is given each family – clas Lobatto and Radau, class IIS Lobatto and Radau. Horizontal sums equal h_n for class I and $h_n^2/2$ for class II

Class I, Lobatto 6th Order. A-stable

$h_1 =$	0	0	0	0	0
$h_2 = (5-\sqrt{5})/10$		$(11+\sqrt{5})/120$	$(25-\sqrt{5})/120$	$(25-13\sqrt{5})/120$	$(-1+\sqrt{5})/12($
$h_3 = (5+\sqrt{5})/10$		$(11-\sqrt{5})/120$	$(25+13\sqrt{5})/120$	$(25+\sqrt{5})/120$	$(-1-\sqrt{5})/12($
$h_4 =$	1	1/12	5/12	5/12	1/12
w		1/12	5/12	5/12	1/12

Control: $A_3T^4/4 = 5T(F_4 - F_1 + \sqrt{5}(F_2 - F_3))/4$

Class I, Radau 7th Order. Stable for $0 > \lambda T > -19.1$. Obtain numerical values for h_2, h_3, h_4 from Table I

h_1	0.0	0.0	0.0	0.0
h_2	0.08668 21048 99833	0.13980 78018 10003	−0.01792 97950 24201	0.00378 04265 53518
h_3	0.04645 25771 93484	0.37670 14480 28600	0.18130 08949 07813	−0.01392 17845 70632
h_4	0.07240 46765 07266	0.30304 19425 59723	0.42850 26895 66694	0.10746 27318 53612
w	0.06250 00000 00000	0.32884 43199 80060	0.38819 34688 43172	0.22046 22111 76768

Control: $A_3T^4/4 = T(-2.1875\ F_1 + 4.453202790\ F_2 - 3.488522645\ F_3 + 1.222819855\ F_4)$

Class IIS, Lobatto 6th Order

$h_1 =$	0	0	0	0	0
$h_2 = (5-\sqrt{5})/10$		$(125-27\sqrt{5})/3000$	1/50	$(53-25\sqrt{5})/600$	$\sqrt{5}/15($
$h_3 = (5+\sqrt{5})/10$		$(125+27\sqrt{5})/3000$	$(53+25\sqrt{5})/600$	1/50	$-\sqrt{5}/15($
$h_4 =$	1	1/12	$(5+\sqrt{5})/24$	$(5-\sqrt{5})/24$	0
w		1/12	$(5+\sqrt{5})/24$	$(5-\sqrt{5})/24$	0
u		1/12	5/12	5/12	1/12

Control: $A_3T^5/20 = T^2(F_4 - F_1 + \sqrt{5}(F_2 - F_3))/4$

Class IIS, Radau 7th Order. Obtain numerical values for h_2, h_3, h_4 from Table I

h_1	0.0	0.0	0.0	0.0
h_2	0.01292 89608 95268	0.01115 01356 03964	−0.00195 38089 83474	0.00041 89645 74088
h_3	0.03754 00899 66293	0.12188 77209 19233	0.01612 02500 91054	−0.00118 33688 79851
h_4	0.05636 14364 35399	0.23145 03384 08384	0.12214 15033 99353	0.00538 26755 29472
w	0.06250 00000 00000	0.25901 73400 78606	0.15895 23624 83586	0.01953 02974 37809
u	0.06250 00000 00000	0.32884 43199 80060	0.38819 34688 43172	0.22046 22111 76768

Control: $A_3T^5/20 = T^2(-0.4375\ F_1 + 0.890640558\ F_2 - 0.697704529\ F_3 + 0.244563971\ F_4)$

finds that $A_3T^4/4 = 5T[F_4 - F_1 + \sqrt{5}(F_2 - F_3)]/4$. This is the highest-order term retained in the time-series expansion and is useful for a sequence-size control, as discussed in Section 3.4. Table II lists these controls.

Lapidus and Seinfeld describe tests indicating that implicit Runge-Kutta methods

may not converge rapidly enough to compete with explicit methods. However, the four families of implicit Runge-Kutta discussed here are basically equivalent to the time-series method and should converge just as rapidly. Thus extrapolated α-values could be used in Equation (6) to predict the F-values, and so on.

Two points in favor of the time-series formulation may be mentioned. First, there is an efficient algorithm such that the iteration converges accurately in two passes. The time trials to be described in the next section are most encouraging. Second, the values of h_n in Table I are sufficient for integration to any order up through 15th of either the 1st- or 2nd-order differential equation. Changing only the h-values and a few indices one can write a general program for integrating either equation to any order. In contrast, the implicit Runge-Kutta versions require a far more numerous set of coefficients, which are different for each equation order and each integration order.

6.4. Multi-off-grid integration

Beaudet (1972) has described a system of integration, developed from multi-step predictor-corrector forms, that uses spacings analogous to Gauss-Legendre for substeps within each sequence. It is a multi-sequence method where each sequence is the same length. The predictors and correctors use information from previous sequences. His system is explicit. It is not self-starting, and there is no freedom to change sequence size each time. The point of contact with our system is the use of Gaussian-type spacings to achieve enhanced accuracy with few function evaluations in solving 1st- and 2nd-order differential equations.

7. Tests and Comparisons

7.1. The differential equations

The test orbit was that of a hypothetical asteroid revolving about the Sun between or near the orbits of Jupiter and Saturn, being perturbed by these planets. Two systems of equations were used:

The Cowell equations are of the form

$$\ddot{x} = -\mu x/r^3 + f_x, \tag{31}$$

where $\mathbf{r}(x, y, z)$ is the position of the asteroid, and μ is the mass factor. The first term is the direct force due to the primary mass (Sun) at the center of coordinates, and

$$f_x = \sum_j m_j[(x_j - x)/\varrho_j^3 - x_j/r_j^3] \tag{32}$$

is the x-component of the perturbational force due to all other masses m_j at positions $\mathbf{r}_j(x_j, y_j, z_j)$. Here $\varrho_j = |\mathbf{r}_j - \mathbf{r}|$.

The Encke differential equations calculate the difference between the actual orbit $\mathbf{r}$ and a conic reference orbit $\mathbf{R}(X, Y, Z)$. Let ξ be the x-component of this difference. Then

$$\ddot{\xi} = \ddot{x} - \ddot{X} = (-\mu x/r^3 + \mu X/R^3) + f_x, \tag{33}$$

where the term in parentheses may be called the 'Encke force'. The Encke force and f_x are both small. In our application the reference **R** is rectified (re-fitted) at the start of every sequence. The Encke equations are discussed more fully in Section 8. The system is complicated, but it does allow for large sequence lengths.

7.2. The test problem

The hypothetical asteroid is started in a circular orbit outside Jupiter's orbit. It is orbit No. 96 from our recent study of random orbits in the solar system, Everhart (1973). The initial inclination is 6.9° and the initial circular radius is 6.05 AU (atronomical units). Jupiter and Saturn are placed in unchanging elliptical orbits that are found analytically. Only the asteroid's orbit is integrated. The path is followed for 68.23 years, which is more than 4 orbit revolutions for the asteroid. During this time it makes two approaches to Jupiter with 1.0 AU and one to Saturn within 3.0 AU. At the end of this time the integration is stopped and the asteroid's orbital elements are determined. Then the asteroid's velocity vector is reversed, as are the directions of the two planets. The integration is restarted and carried back for precisely the same length of time to the starting point. The errors in closure Δx, Δy, Δz are evaluated as

$$\Delta r = (\Delta x^2 + \Delta y^2 + \Delta z^2)^{1/2} . \tag{34}$$

In the unit system employed $\mathbf{r}(x, y, z)$ happened to have nearly unit magnitude, so that Δr is approximately the relative error in closure of position. The relative error in closure for velocity is found to be comparable with Δr.

7.3. Discussion of error criteria

It may be questioned whether the reversibility test is appropriate. In the case of a symmetric integrator that retraces its steps one might expect a small error of closure, whereas the error might be large at the reversal point. However, the Gauss-Radau spacings used in most of the tests are not symmetric, and the nature of the restarting and sequence-size control is such that the return sequencies do not fall on the outbound sequences. In the test problem the correct values at the reversal point are known to high accuracy, and the error there was printed out and compared with the error of closure. In every case the error at the reversal point was comparable to the error of closure (both in position and velocity) within a factor usually less than 3, neither systematically higher or lower. The same rough equivalence between the two error criteria holds for the Runge-Kutta-Nyström-Fehlberg methods to be described later. Changing the error criterion would not change the figures significantly.

7.4. Tests using the Encke equation

The tests shown in Figure 1 used the Encke equations in the 3-substep case. The relative error Δr is plotted vs the computation time measured on a CDC 6600 computer in single precision (14 decimal digits). Line (5) of Figure 1 results from using Radau spacings and passing through the sequence twice.

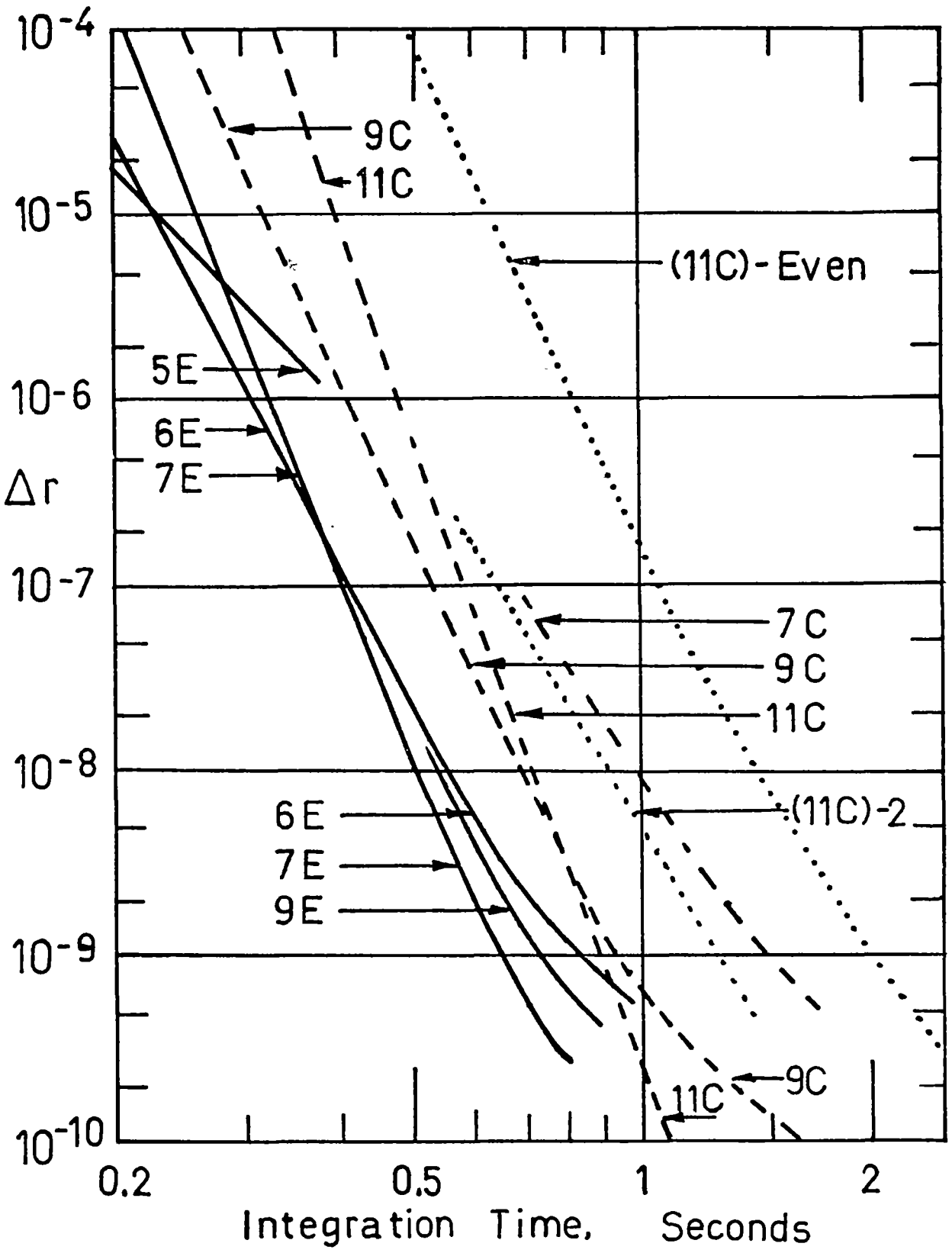

Fig. 2. Error vs computation time plots for present methods. Solid lines 5E, 6E, 7E, and 9E solve the Encke equation to orders 5, 6, 7, and 9. Dashed lines 7C, 9C, and 11C solve the Cowell equation to orders 7, 9, and 11. All these are for two passes through the sequence, except 11C, which is for three passes. The dotted line marked (11C)-2 is as in 11C, except with two passes in the sequence. The above cases use the appropriate Radau or Lobatto spacings for the substeps. The dotted line '(11C)-Even' is the same program as for 11C except with evenly spaced substeps.

The solid lines 5E, 6E, 7E, and 9E on Figure 2 are the result of integrating the Encke equation to nominal orders 5, 6, 7, and 9. Order 7 is about optimum. The slope of the lines on the figure depends on the effective order of the integration, which may not be the same as the nominal order. The lines become steeper as the effective integration order increases. The curvature at the lower ends is caused by round-off error.

7.5. Tests using the Cowell equation

The dashed lines 7C, 9C, and 11C of Figure 2 describe tests integrating the Cowell equation to nominal orders 7, 9, and 11. Generally, 9th order is best. There are several lines for nominal 11th order on the figure. The dotted line (11C)-2 is for the usual 2 passes per sequence. With a slope intermediate between 7C and 9C, the effective order is about 8. The dashed line 11C is the result of 3 passes through the sequence. Although

each sequence then takes 35% more time, there is a large net gain. Method 11C is limited by round-off error near $\Delta r = 10^{-11}$, this being due to the 14-digit precision.

Finally, the dotted line marked '(11C)-Even' on Figure 2 is for the same program as 11C (including the 3 pass per sequence) except that the substeps were evenly spaced. At a computation time of 1 s there is an advantage in accuracy of $2\frac{1}{2}$ orders of magnitude in using the Radau spacings.

7.6. Comparisons with other methods

On Figure 3 three of the best of the present methods are drawn again. Here 7E is the Encke solution of 7th order, and 9C and 11C are the Cowell solutions of 9th and 11th order.

We believe that the new high-order explicit Runge-Kutta-Nyström-Fehlberg methods are among the best of other single-sequence procedures. They also solve the equation $\ddot{x} = F(x, t)$ directly without reducing to first order, and the necessary coefficients and sequence-size control have been worked out by Fehlberg (1972). They are delightfully easy to program. The line marked FE6 is Fehlberg's 6th-order method RKN6(7) applied to the Encke equation, and FC8 in his 8th-order method RKN8(9) applied to the Cowell equation. Among his methods we find these orders to be optimum for the present test. Fehlberg's 4th-order method RKN4(5) in the Encke case, FE4, and the Cowell case, FC4, are also shown. Methods 7E and 9C are twice as fast as FE6 and FC8 at the same accuracies in these tests.

The line marked XC6 is for a time series method of 6th order solving the Cowell equation. It has a similar structure to methods developed by Wielen (1967) and Aarseth (1972) for N-body integrations. The line YE5 is for a 5th-order time-series method we used in a study of the evolution of orbits in the solar system, Everhart (1973). It solves the Encke equation. The perturbational force is fitted empirically as in the Wielen-Aarseth method, and the Encke force is expanded analytically.

8. Encke Equations and Close Encounters

The Encke formulation is not often chosen by celestial mechanicians. In its classical form it is more complicated and generally slower than the Cowell method. In multi-step methods it is particularly cumbersome, since the integration must be restarted every time the reference orbit is rectified. In single-step (single-sequence) methods, where there is no difficulty in starting, some advantages appear if the reference orbit is rectified anew at the start of each sequence, redefining $\mathbf{R} = \mathbf{r}$ and $\dot{\mathbf{R}} = \dot{\mathbf{r}}$ at $t_1 = 0$. Under this circumstance the two terms $-\mu x/r^3$ and $\mu X/R^3$ in Equation (33) very nearly cancel, and their sum is less than f_x. The right side of Equation (33) is much smaller than the right side of the Cowell Equation (31), which has the large term $-\mu x/r^3$.

Even with these advantages, which do, in fact, allow larger sequence lengths, the Encke equations are no obvious improvement over the simpler Cowell system, because the reference orbit position and velocity involve substantial additional pro-

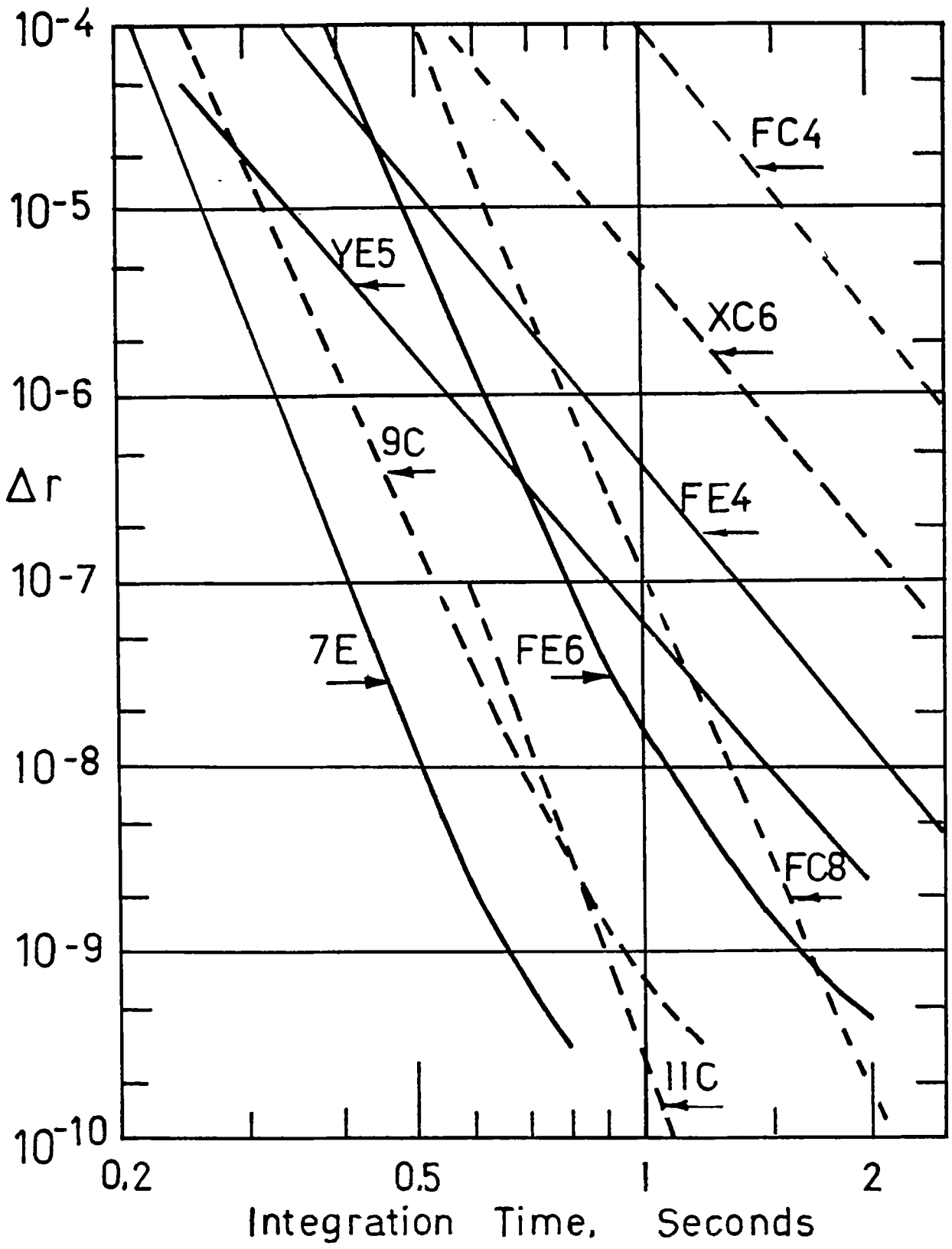

Fig. 3. Comparisons of error vs computing time for several methods integrating the same test orbit. Numbers refer to the nominal order of the integration. Methods solving the Encke equation are solid lines and include an 'E' in the label, and those solving the Cowell equation are dashed lines with a 'C'. Lines 7E, 9C, and 11C are for the present method, FE6, FE4, FC8, and FC4 are Fehlberg's high order Runge-Kutta-Nyström methods, and XC6 and YE5 are time-series methods described in the text.

gramming. The best method to handle the reference is to use the universal variables introduced by Stumpff (1947) and developed further by Herrick (1965), since they allow the reference to be projected ahead easily in rectangular coordinates. The most efficient notation and formulation, in our opinion, are those by Goodyear (1965) and Pitkin (1966). Only with these methods (and ingenuity in avoiding solving Kepler's equation in carrying the reference along) do Encke's equations show an improvement over Cowell's. The improvement, a 35% decrease in computation time in the present tests, is marginal justification for all the complications.

It may be that the significant advantage of Encke's equations compared to Cowell's is concerned with close encounters and regularization. Whenever the path of a comet brings it near the Sun, the term $-\mu x/r^3$ becomes large, and whenever it passes near Jupiter the term $-m_j x/\varrho_j^3$ dominates enormously. These are $1/r^2$ poles in the differential equation that causes difficulties. The way to remove such a pole in the Cowell

equations is to regularize them. Usually time is transformed by $d\psi = dt/r$, where ψ is a pseudo-time and r is the distance to the troublesome pole. Frequently the coordinates are transformed as well, using a transformation such as that of Kustaanheimo and Stiefel. Regularization and its advantages are reviewed by Bettis and Szebehely (1972).

In the Encke formulation the $1/r^2$ pole in the reference orbit is handled analytically. This is best done in a regularized manner with universal variables, so that one does use regularization in that sense. However, the ξ-equation, which is integrated numerically, is concerned only with perturbations and small departures from the reference, and it does not require regularization. In order to show this we expand the right side of Equation (33) in a Maclaurin time series about $t_1 = 0$ (using subscript 1 for initial values). Expanding the terms $-\mu x/r^3 + f_x$ in such a series is considerably more complicated than the expansion of $-\mu x/r^3$ alone because of the interaction of the f_x term. To the result we add the expansion of $+\mu X/R^3$, and all terms that do not depend on the perturbation force **f** disappear. Through terms in t^2 one finds

$$\ddot{\xi} = (\mu/r_1^3)\,[-f_{x1} + 3x_1\mathbf{f}_1 \cdot \mathbf{r}_1/r_1^2]\,t^2/2 + \cdots + f_{x1} + \dot{f}_{x1}t + \ddot{f}_{x1}t^2/2 + \cdots. \tag{35}$$

The second series is the perturbational force, and the first series, of which only the leading term is shown, is the Encke force. This term includes the constant factor μ/r_1^3 that is large when r_1 is small. However, the factor in square brackets has the dimension and magnitude of the perturbing force, a quantity that is ordinarily small. The factor $t^2/2$ is always very small when rectification is carried out at the start of every sequence. We see that the Encke force under these conditions is always small.

The Encke formulation requires a transformation when the integration path passes near one of the secondary masses. When a comet is about to pass very near Jupiter, one should transform to Jovicentric coordinates, identifying Jupiter, temporarily, as the primary mass. The reference conic is then referred to Jupiter.

There are certain advantages in this Encke system compared to the regularized Cowell system. First, with the Encke equations one can solve the orbit with $3N$ second-order differential equations of the form of Equation (1), where N is the number of bodies whose path is integrated. However, in regularizing the Cowell equations, first derivatives are introduced, and the system is usually solved as $6N$ first-order equations. Since one second-order differential equation is solved just as easily as one first-order equation (at least by the present method), it would appear that regularization doubles the work in this respect. Second, in the Encke formulation, time can be retained as the independent variable for all bodies. In the regularized Cowell case, one or more pseudo-times ψ must be related to time by integrating additional first-order equations of the form $\dot{\psi} = 1/r(\psi, t)$.

The author's experience in Monte Carlo studies of the interaction of comets and asteroids with the solar system, Everhart (1972, 1973), have involved calculation of more than a million orbits by the Encke method using single-sequence integrators. The method has proved to be fast and accurate in handling close encounters.

Acknowledgements

Discussion of Radau and Lobatto quadratures as well as implicit Runge-Kutta methods with Mr W. H. Goodyear has been very helpful. There have been useful communications with Prof. J. M. A. Danby, Prof. Leon Lapidus, Prof. Dale G. Bettis, Prof. J. C. Butcher, Dr Myron Lecar, and Dr Paul Beaudet. The author appreciates the detailed comments of his colleagues, Prof. Hershel Neumann, Dr Janet Rountree Lesh, and Dr Robert Stocker. The problem of efficient integration methods arose in connection with a study of comet debris and gas between the planets in the solar system, and the author is grateful to the National Center for Atmospheric Research, sponsored by the National Science Foundation, for the computer time used in this study.

References

Aarseth, S. J.: 1972, in M. Lecar (ed.), *The Gravitational N-Body Problem*, D. Reidel Publ. Co., Dordrecht, p. 373.

Abramowitz, M. and Stegun, L. A.: 1964, *Handbook of Mathematical Functions* (U.S.) National Bureau of Standards Appl. Math. Sers. **55** (U.S. Government Printing Office, Washington) 9th Printing 1971, Sec. 25.4.31–32.

Beaudet, P. R.: 1972, *Bull. Am. Astron. Soc.* **4**, 417 (abstract). Am. Astron. Soc. Div. Dynamical Astron./SIAM Meeting, Austin, Texas, Oct, 1972, (Proceedings to be published).

Bettis, D. G. and Szebehely, V. S.: 1972, in M. Lecar (ed.), *The Gravitational N-Body Problem*, D. Reidel Publ. Co., Dordrecht, p. 388.

Butcher, J. C.: 1964, *Math. Comp.* **18**, 233.

Chipman, F. H.: 1971, Waterloo Univ. Res. Rept. CSRR 2042.

Ehle, B. L.: 1969, Waterloo Univ. Res. Rept. CSRR 2010.

Everhart, E.: 1972, *Astrophys. Letters* **10**, 131.

Everhart, E.: 1973, *Astron. J.* **78**, 316.

Fehlberg, E.: 1972. Nat. Aero. and Space Adm. (U.S.A.) Tech Rpt. NASA TR R-381.

Glasmacher, W. and Sommer, D.: 1966, *Implizite Runge-Kutta Formeln*, Westdeutscherverlag-Köln und Opladen.

Goodyear, W. H.: 1965, *Astron. J.* **70**, 189.

Herrick, S. J.: 1965, *Astron. J.* **70**, 309.

Lapidus, L. and Seinfeld, J. H.: 1971, *Numerical solution of Ordinary Differential Equations*, Academic Press, New York, pp. 59–63, pp. 129–136.

Pitkin, E. T.: 1966, *J. Astronaut. Sci.* **13**, 204. See his Equations (1)–(9). Use $n = 3$ in his Equation (5).

Stroud, A. H. and Secrest, D.: 1966, *Gaussian Quadrature Formulas*, Prentice-Hall, Inc., Englewood Cliffs, N.J., U.S.A., Tables 11 (Lobatto) and 12 (Radau).

Stumpff, K.: 1947, *Astron. Nachr.* **275**, 108.

Weilen, R.: 1967, *Veröff. Astron. Rechen-Inst. Heidelberg* **10**.