THE HYPERBOLIC KEPLER EQUATION (AND THE ELLIPTIC EQUATION REVISITED)

R. H. GOODING and A. W. ODELL

Royal Aerospace Establishment, Farnborough, Hants, England

(Received: 16 November 1987; accepted: 13 October 1988)

Abstract. A procedure is developed that, in two iterations, solves the hyperbolic Kepler's equation in a very efficient manner, and to an accuracy that proves to be always better than 10^{-20} (relative truncation error). Earlier work on the elliptic equation has been extended by the development of a new procedure that solves to a maximum relative error of 10^{-14}.

1. Introduction

In an earlier paper (Odell and Gooding, 1986), we considered the classical Kepler equation for elliptic orbits, and recommended two particular procedures for its solution. This paper was a greatly shortened version of the authors' monograph (Gooding and Odell, 1985), and all references to our 'previous work' in the present paper should be taken to cite the two references just given. To complement the previous work, we have now studied the hyperbolic equation, and the present paper is a shortened version of the recent RAE report by Gooding (1987). Though there is a very extensive literature on the elliptic equation, several papers having appeared since our previous work, little has been published on the hyperbolic equation. Burkardt and Danby (1983) consider it briefly, before passing to a generalized equation expressed in terms of universal variables, whilst other authors, such as Bergam and Prussing (1982), only consider hyperbolic orbits in the context of the universal (generalized) equation. (*Postscript:* a recent paper (Boltz, 1987) is, however, entirely devoted to the hyperbolic equation.)

The use of universal variables is aesthetically very attractive and we discussed the universal equation briefly in our previous work, referring in particular to the recent paper by Shepperd (1985). In parallel with the present work, one of us (Gooding, 1988) has been studying the use of universal elements in conversion algorithms (to and from position and velocity), but a conclusion of that study is that the best way to proceed, inside the relevant algorithm, is still to solve the elliptic equation, the hyperbolic equation, or Barker's equation, as appropriate.

The hyperbolic Kepler equation is

$$e \sinh H - H = M_h, \tag{1}$$

where e (eccentricity) and M_h (hyperbolic mean anomaly) are assumed to be known, and H (hyperbolic eccentric anomaly, often denoted by F) is to be determined. Since the range of e is from unity (rectilinear hyperbolic orbit) to infinity (uniform linear motion, for zero attractive force), there is some advantage in rewriting equation (1) as

$$\sinh H - gH = L, \tag{2}$$

Celestial Mechanics **44** (1988), 267–282.

where

$$g = 1/e, \qquad L = M_h/e;$$

we also define

$$g_1 = 1 - g.$$

Later we shall find further advantage in reformulating the equation so that $\sinh H$, rather than H, is the quantity to be determined.

Our approach to the iterative solution of Equation (1), or equivalently (2), has been to adhere as far as possible to the philosophy underlying the final procedure that we previously recommended. In particular, we sought a combination of starting formula and iteration process accurate enough for the resulting procedure to be certain to meet a given accuracy goal after a fixed small number (preferably two) of iterations; assuming a smoothly continuous starter over the (e, M_h)–data space, the output (H or $\sinh H$) must then also be smoothly continuous. ('Smooth continuity' is meant to be synonymous with the term 'smooth portability' discussed in the previous work.) The accuracy goal was expressed as a ceiling for the relative error in the output quantity, the ceiling of 10^{-13} being selected; as in the previous work, this was related to the computer used for most of the work, but the goal has not been the same, since for the elliptic equation the value of 10^{-13}(rad) had been an *absolute*-error ceiling.

Section 2 describes two procedures that solve for H, and Section 3 describes two procedures that solve for S $(= \sinh H)$. As the previously-developed best iterator providing quartic convergence was incorporated in all four procedures, Sections 2 and 3 mainly provide descriptions of the starters. Section 4 is devoted to computer implementation and results for the last (and best) of the four procedures, and Section 5 describes a new procedure developed for the elliptic equation, such that the relative-error goal used for the hyperbolic equation can be met. Also included in Section 5 is a discussion of an alternative iteration method, due to Laguerre and recently recommended by Conway (1986); though it only provides cubic convergence, it is much more robust than our standard iterator, as it comes into its own when a good starter is not available.

The recommended hyperbolic procedure meets the goal of 10^{-13} relative accuracy with the greatest of ease, since it provides 20-decimal-digit accuracy in all cases. The new elliptic procedure does not do so well, the accuracy provided being 14 digits.

2. Procedures Solving for *H*

Our initial approach was to follow EKEPL2 (the final procedure developed in the previous work) as closely as possible. For our first hyperbolic procedure, therefore, we chose the starting formula (assuming $L \geqslant 0$)

$$H_0 = gH_{01} + g_1 \sinh^{-1} L; \tag{3}$$

here H_{01} is a special starter, for $g = 1$, defined (to give smooth continuity) by

$$H_{01} = \begin{cases} (6M_h)^{1/3} & \text{if} \quad 0 \leqslant M_h \leqslant 1/6 \qquad (4a) \\ \ln 2(M_h + 1/3) + 1 & \text{if} \quad M_h \geqslant 1/6. \qquad (4b) \end{cases}$$

This first procedure gave disappointing results, however, the guaranteed accuracy after two iterations being no better than seven decimal digits (relative truncation error), as shown in the detailed analysis by Gooding (1987). A third iteration would increase the accuracy to at least 26 digits, but the use of a third iteration would reduce the efficiency. Thus our initial approach was not very productive, except that the analysis brought to light a small defect that is described in the next paragraph; this defect has also been present in EKEPL2, which is why further attention was in due course given to the elliptic equation as well as the hyperbolic.

For small values of M_h, Equation (2) gives

$$H^3 + 6g_1 H - 6L \approx 0; \tag{5}$$

this can be further truncated to (4a), which is therefore an excellent starter when $g_1 = 0$. For $g_1 > 0$, however, the neglect of the second term of (5) can lead to a serious overestimate of H, which the interpolation (3) does little to remedy when g_1 is only slightly greater than zero (e only slightly greater than unity). The overestimation is never so gross as to affect convergence adversely, so long as this statement is interpreted in terms of truncation error (whether absolute or relative), but the effect on *rounding* error is another matter entirely, as shown in the detailed analysis. Even here there is no problem with absolute error, because the iterator works so well, but relative (rounding) error can only be reduced at a rate (per iteration) determined by the computer word-length.

The defect referred to was not picked up in the previous work, because the analysis was then conducted almost entirely in terms of *absolute* error, this being possible because of the periodic nature of $E - M$ for the elliptic equation; thus absolute rounding error is bounded over the full (infinite) range of M. In consequence, we did not look at very small non-zero values of M; had we done so, the problem would have come to light in the plotting of rounding error in Fig. 6 of Odell and Gooding (1986).

With a view to eliminating the small defect described, and to devise a procedure in which two iterations would always suffice, we abandoned our initial approach to the solution of (2), in favour of one based on the approximation (5). The new approach, as in our previous consideration of the approximating cubic for the elliptic equation, follows the notation of Ng (1979).

We define H_{00} as the starter, which is a good approximation for small M_h and arbitrary g, given by the (unique) solution of the equation (cf. (5))

$$H_{00}^3 + 3qH_{00} - 2r = 0, \tag{6}$$

where $q = 2g_1$ and $r = 3L$. As before, we note that (6) is the classical cubic equation, for which we can do better than Ng by expressing the solution with a single cube root, as

$$H_{00} = s - q/s, \tag{7}$$

where (assuming M_h, and hence r, to be non-negative)

$$s = [(r^2 + q^3)^{1/2} + r]^{1/3}.$$

This expression for H_{00} is still not optimal, however, as unnecessary rounding error will be experienced when $s^2 \approx q$; some algebraic manipulation gives us the optimal solution as

$$H_{00} = \frac{2r}{s^2 + q + (q/s)^2}. \tag{8}$$

Since $\sinh^{-1} L$ is appropriate, as a starter, for large values of M_h (just as it was for small values of g), we can use (8) as the basis for an overall starter. Thus H_0 is defined by

$$H_0 = (M_h \sinh^{-1} L + H_{00})/(M_h + 1). \tag{9}$$

On coupling Equation (9) to the usual iterator, we have a procedure that is free of the deficiencies of the first procedure. In particular, there is no rounding-error problem, and the maximum relative error (due to truncation) after two iterations is only 7×10^{-14}.

3. Procedures Solving for S ($= \sinh H$)

The development of the second procedure of Section 2 might have ended our study of the hyperbolic equation, but an alternative approach was thought to be worth investigating. In this, equation (1) is not just rewritten as (2), but more significantly it is rewritten as

$$S - g \sinh^{-1} S = L, \tag{10}$$

where

$$S = \sinh H. \tag{11}$$

Then S, instead of H, is the quantity to be solved for.

There are at least two advantages in solving Equation (10), as opposed to (1) or (2). First, it is more efficient, since the only use to be made of H is normally via $\sinh H$ and $\cosh H$; moreover, there is no loss of accuracy in deriving $\cosh H$ as $\sqrt{(1 + S^2)}$, in complete contrast to the loss that can occur if $\cos E$ is derived from $\sin E$. Secondly, there is a gain in precision with the use of S, rather than H, for large values of $|S|$; thus the evaluation of the composite $\sinh(\sinh^{-1})$ function, like the composite exp(ln), cannot be relied upon to be the identity function when the argument is large. It is to be noted, also, how great is the resemblance between the reformulated equation and the standard elliptic equation, since g has the same range as e for an ellipse, whilst $\sinh^{-1} x$, like $\sin x$, expands as $x - x^3/6 + O(x^5)$. As a final introductory point, we remark that the change of variable from H to S is bound to affect the operation of the iteration process; thus if we chose to solve both Equations (1) and (10), with starters (H_0 and S_0) related by (11), this relation would not be preserved during iteration.

The first approach to a starter for Equation (10) was based on the direct modification of the successful solution of Equation (2) via the cubic Equation (6). Thus we solve (6) for S_{00}, rather than H_{00}, if we replace the definitions of q and r by

$$q = 2g_1/g \quad \text{and} \quad r = 3L/g. \tag{12}$$

In view of the possibly-zero denominators in (12), however, it is better if we write the cubic equation in the homogeneous form

$$aS_{00}^3 + 3bS_{00} - 2c = 0, \tag{13}$$

where $a = g$, $b = 2g_1(= gq)$ and $c = 3L(= gr)$. In extending S_{00}, the starter appropriate to small L (still, for convenience, assumed to be positive) to an overall starter, the most obvious analogy seemed to be with S_0 given by

$$S_0 = (L^2 + S_{00})/(L + 1), \tag{14}$$

but this gave disappointing results. It was in due course decided, therefore, to investigate a different type of starter.

One of the possible starters (the eleventh) discussed in our previous work was a very complicated one, constructed on the rationale that it should have the right form in the region of awkward convergence, whilst at the same time matching the formal e-series expansion (of the solution to the elliptic equation) to terms of order e^3. By 'having the right form' etc., we just meant that, for $e = 1$ and small M, the starter should behave like $\sqrt[3]{6M}$, since the more stringent constraint (imposed by the cubic-equation approximation, cf. (5), for $e \neq 1$) would then be met by the series matching. It was decided to apply this rationale to (the hyperbolic) Equation (10), but to develop the desired new starter in a straightforward and systematic manner by basing it on Lagrange's expansion theorem (Whittaker and Watson, 1940).

The objective may be summarized as being an expression of the form

$$S_0 = L + gB^{-1/3} \sinh^{-1} L, \tag{15}$$

where $B = 1 +$ terms in g, g^2, etc. as required. Clearly

$$B = F(S)/F(L), \tag{16}$$

where F is the function $(\sinh^{-1})^{-3}$ (a somewhat unfortunate notation as the two negative signs have different meanings). We use the expansion theorem to express $F(S)$ as a function of L (with parameter g), and to help in this we define $l = \sinh^{-1} L$. Then the theorem gives

$$F(S) = F(L) + \sum_{n=1}^{\infty} \frac{g^n}{n!} D^{(n-1)}\{l^n DF(L)\} \tag{17}$$

where D is the operator d/dL.

On performing the necessary differentiations and dividing by $F(L)$, we find that (17) leads to

$$B = (1 - g\Lambda)^3 + \tfrac{1}{2}g^2 l\Lambda^3\{3L(1 - g\Lambda) + gl\Lambda^2(1 - 2L^2)\} + O(g^4), \tag{18}$$

where $\Lambda = \operatorname{sech} l$. But B has to behave like $l^2/6$, when $g = 1$ and l is small; so we give up the formally-accurate g^3 term in (18), writing instead

$$B = (1 - g\Lambda)^3 + g^2 lL\Lambda^3(9 - 8g)/6 + O(g^3). \tag{19}$$

Substitution of (19) in (15) gives the required starter. Coupled with the usual iteration process, it works so well (as we will see in the next section) that the resulting procedure enjoys our unqualified recommendation for use in solving the hyperbolic Kepler equation.

4. Implementation and Results for the Recommended Procedure

The Fortran–77 function SHKEPL has been written to implement the recommended solution procedure. Its arguments are L and g_1, and it is listed in Appendix A. The second argument is g_1, rather than g, to minimize rounding error in the vicinity of $(g, L) = (1,0)$, when $S - g \sinh^{-1} S$ has to be computed by the special subordinate function SHMKEP, listed in Appendix B; SHMKEP operates in the same way as the function (EMKEPL) we gave before, except that it is rather more efficient since $S - g \sinh^{-1} S$ is computed as $g_1 S + g(S - \sinh^{-1} S)$ rather than $(S - \sinh^{-1} S) + g_1 \sinh^{-1} S$. Two other (non-standard) subordinate functions are used by SHKEPL: DCUBRT for obtaining a cube root; and DASINH for the inverse sinh.

The computer used for testing SHKEPL has mainly been a PRIME 750, as in the previous study, but a Cray 1S was used for the most accurate results. Though the input for SHKEPL is actually L and g_1, the test data consisted of a wide range of values for H and e, H being the source for the reference value of S, from which the nominally input L was obtained via equation (10). From the value, S_i, extracted from SHKEPL after i iterations, we have the relative error, σ_i, given by

$$\sigma_i = (S_i - S)/S. \tag{20}$$

For each e, the range of H extended from 10^{-30} to 10^4, at a fixed geometric interval, the maximum value of $|\sigma_i|$ obtained being a measure of the accuracy of SHKEPL with i iterations. (For normal use, and as listed in Appendix A, i is fixed at 2, but the values 0 and 1 were also of interest.) A very wide range of e was tested, extending from unity to 10^{100}.

Figure 1 provides plots of $\log|\sigma_{\max}|$ against $\log e$, for $i = 0$ (starter), 1 and 2 (standard SHKEPL). The range of e, here, extends only from 1 to 10, as the accuracy is so good for higher values of e. The accuracy is, in fact, striking, since it is evident that SHKEPL gives at least 20 significant figures correct (for S) in all circumstances. If the function is restricted to a single iteration, the number of significant figures reduces to five – this is compatible with the quartic nature of the iterator.

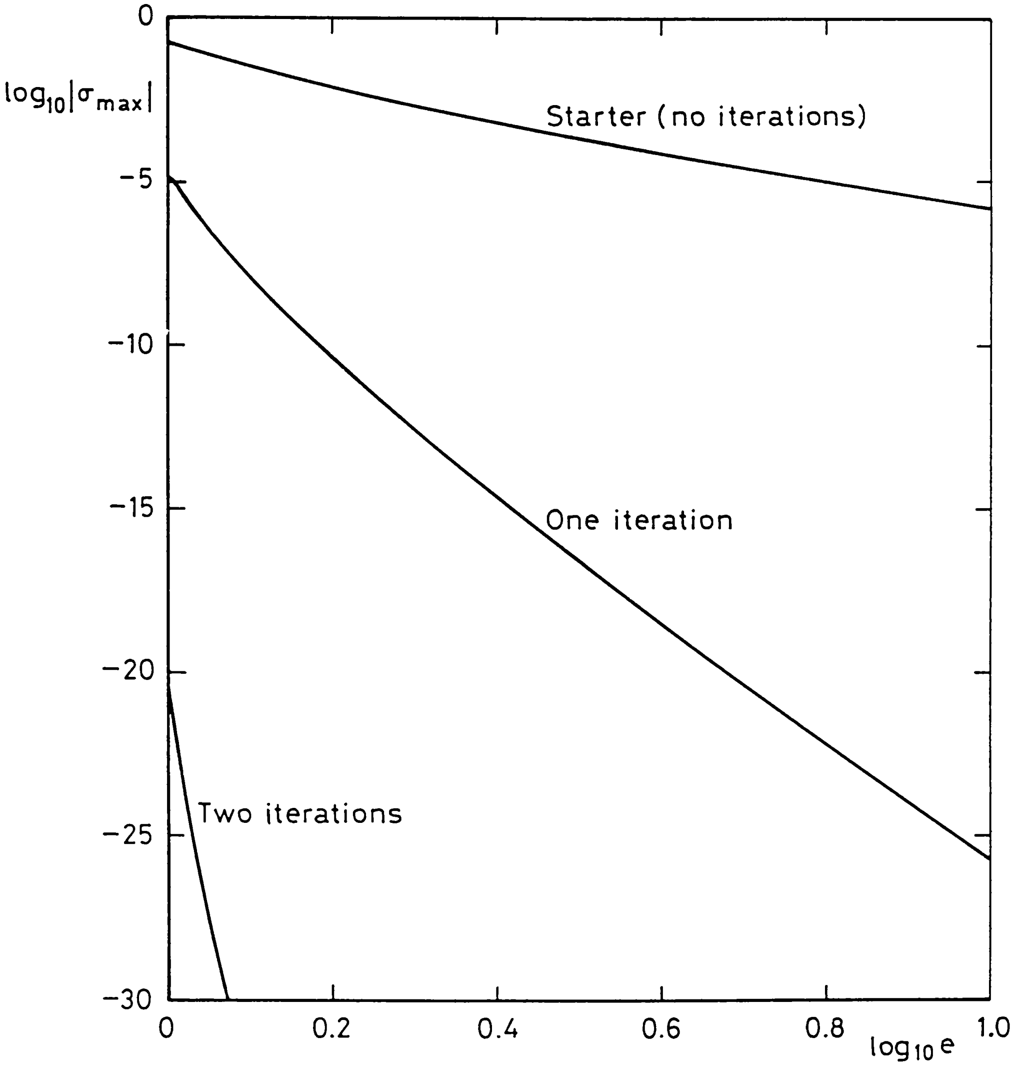


Fig. 1. Maximum relative truncation error for the solution procedure SHKEPL, for the reformulated hyperbolic equation ($e \leqslant 10$).

It is of theoretical interest to note that SHKEPL also works without difficulty when $e \leqslant -1$. This makes it applicable to the solution of Equation (10) for orbits under an inverse-square-law *repulsive* force.

5. The Elliptic Equation Revisited

In our previous study we established the merits of two particular procedures, EKEPL1 and EKEPL2, for solving the standard Kepler equation

$$E - e \sin E = M. \tag{21}$$

The second (and more accurate) of these procedures combined a number of desirable features, as described in our papers; in particular, the incorporation of a bilinear formula in the starter made it extremely efficient. We did not observe, however, that

EKEPL2 suffers from the relative-rounding-error defect, remarked upon here for the first procedure described in Section 2. This may be regarded as a very minor weakness, which it does not seem possible to remedy without loss of efficiency; for completeness, however, we now describe a new procedure, modelled on the second procedure of Section 2, that is free of the defect and retains all the merits of EKEPL2 other than efficiency. We also take the opportunity to make some comments on the application of Laguerre iteration to Kepler's equation, in the light of the recent paper by Conway (1986). (*Postscript*: we wish to draw attention, also, to the work of Mikkola (1987), who reflects the cubic approximation to Kepler's equation by transforming the unknown quantity from E to sin $\frac{1}{3}$E.)

We have seen, in the context of the hyperbolic equation, that the relative-rounding-error defect could be eliminated by use of a cubic approximation, for small H, valid for all e and not just $e = 1$. We adopt the same policy for the elliptic equation, requiring solution of the equation, cf (6),

$$E_{00}^3 + 3qE_{00} - 2r = 0, \tag{22}$$

given in our previous work, where now

$$q = 2e_1/e \quad \text{and} \quad r = 3M/e, \quad \text{with } e_1 = 1 - e.$$

Again we prefer to solve the homogeneous equation, parallel to (13); the function, DCBSOL, for doing this has (like DCUBRT and DASINH) been listed by Gooding (1987).

The solution of (22) for E_{00} provides a suitable starter for small $M(\geqslant 0)$ and any $e(0 \leqslant e \leqslant 1)$, so for arbitrary M over the range $[0, \pi]$ we use the formula

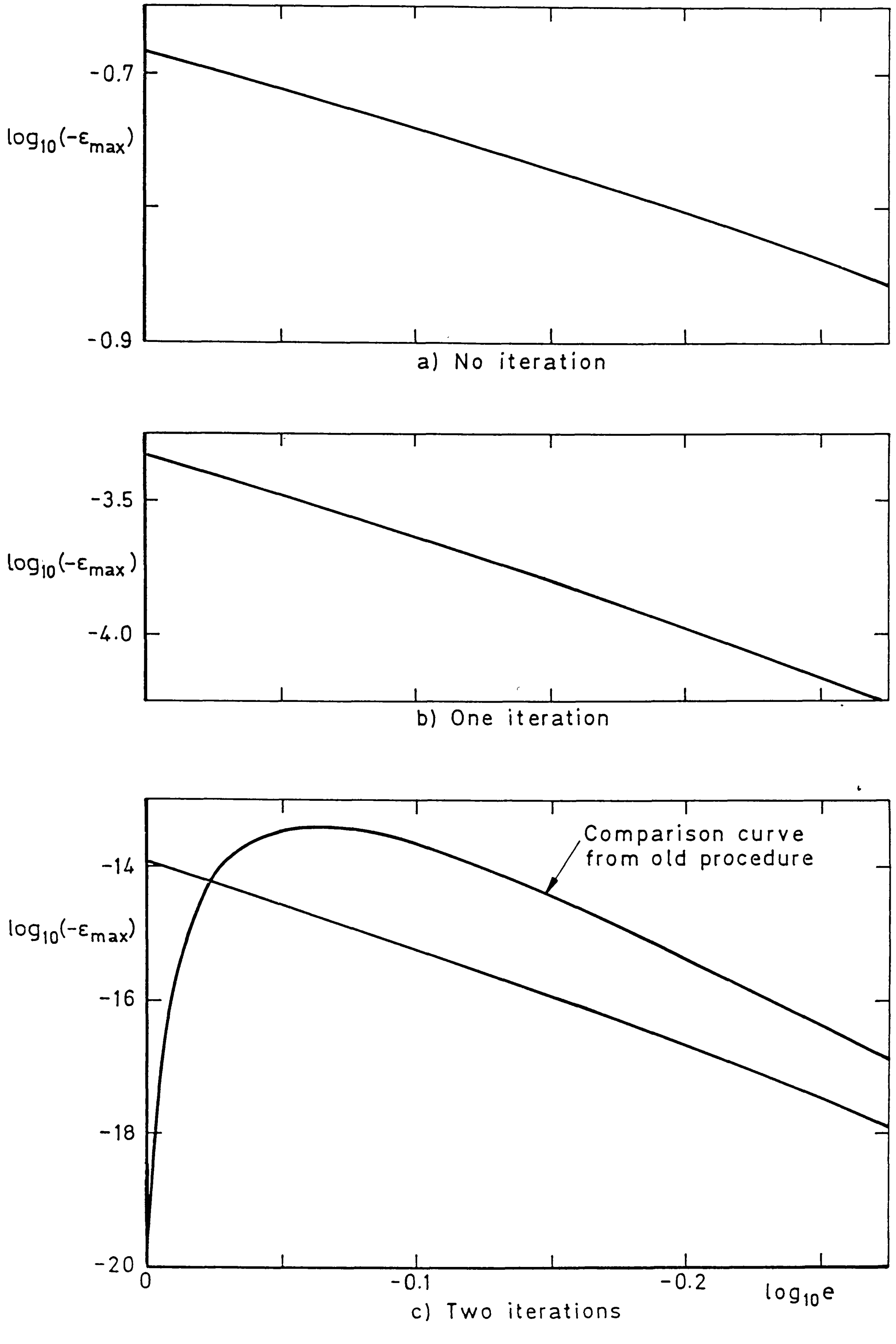
$$E_0 = \frac{M^2 + (\pi - M)E_{00}}{\pi}. \tag{23}$$

Coupled to our usual iterator, this led to a new procedure, EKEPL3, of arguments (as for EKEPL1 and EKEPL2) M and e. To minimize rounding error in the awkward region where $(e, M) \approx (1, 0)$, the second argument was then changed to e_1 and the resulting procedure given the name EKEPL. The Fortran-77 function EKEPL is listed in Appendix C; it uses the subordinate function EMKEP in the awkward region, but this function is so like the original EMKEPL that it is not listed here.

Figure 2 provides, for the solution procedure EKEPL, plots of $\log|\varepsilon_{\max}|$, where

$$\varepsilon_i = (E_i - E)/E, \tag{24}$$

for a (decreasing) range of e from unity to $10^{-0.275}$ (≈ 0.253). To provide a comparison with the old procedure (EKEPL2), the corresponding curve for that procedure is also given, for the definitive $i = 2$. It is seen that the new procedure does better, except when $e > 0.95$ ($\log e > -0.023$), and there is an apparent paradox here, since it was for precisely the high values of e (approaching unity) that EKEPL2 was defective. The paradox is easily resolved, however, since the defect relates to rounding error, whereas Figure 2 only covers truncation error: for $e \approx 1$, both procedures give very small truncation error when M is small (i.e. in the 'awkward region') and the curves of Figure 2 are detemined by much larger values of M; it is a tribute to the efficacy of the bilinear



a) No iteration

b) One iteration

c) Two iterations

Fig. 2. Maximum relative error (truncation) for the new solution procedure, EKEPL, for Kepler's classical equation.

function, used in EKEPL2, that under these circumstances EKEPL2 does so much better than the new EKEPL.

Figure 2 indicates that the accuracy for EKEPL is never worse than about 14 significant figures. Though good, this is inferior to the accuracy of SHKEPL. Since SHKEPL uses a starter based on Lagrange's expansion theorem, it might be thought that, as for the reformulated hyperbolic equation, we would do much better via this theorem than via the cubic-equation-based starter. This is not so, however; results for a procedure based on the expansion theorem, derived in exactly the same way as for the hyperbolic equation, are actually worse than those given by EKEPL. Despite this, we give the equations corresponding to (15) and (19), for completeness; they are

$$E_0 = M + eB^{-1/3} \sin M$$

and

$$B = (1 - e\cos M)^3 + e^2 \sin^2 M\,(9 - 8e)/6 + O(e^3).$$

We now discuss the merits of replacing our quartic generalized-Newton iteration formula by one of the Laguerre formulae, which, though they only give cubic convergence, are shown by Conway (1986) to be much more robust than the Newton formulae.

The general Laguerre formula, for iterating to a root of the equation $f(x) = 0$, may be written

$$\delta = -\frac{Nf}{f'\{1 + \sqrt{[(N-1)^2 - N(N-1)ff''/f'^2]}\}}, \tag{25}$$

where $\delta = x_i - x_{i-1}$ and f is shorthand for $f(x_{i-1})$, etc.; N may be identified as the degree of the polynomial that is matched to f. Thus N is normally taken to be a small integer ($\geqslant 2$, since for $N = 1$ the formula reduces to the Newton–Raphson formula, of only quadratic convergence), but it does not have to be restricted in this way – the formula is sound with N large, infinite, or even non-integral. Conway expresses the denominator of (25) as $f' \pm \sqrt{[(N-1)^2 f'^2 - N(N-1)ff'']}$; this provides a result in the (unrealistic) circumstance that $f' = 0$, but at the expense of the sign ambiguity (resolved, when $f' \neq 0$, by giving the square root the same sign as f').

There would be no advantage in substituting one of the Laguerre formulae for our standard iterator in either of the procedures EKEPL2 or EKEPL, or in any of the procedures developed for the hyperbolic equation, since the starters in all these procedures are good enough for the full (quartic) power of the existing iterator to operate. For EKEPL1 it is another matter, however, since the possibly slow convergence here (infinitely slow in the limit) is completely overcome with one of the Laguerre formulae. The advantage of iteration by (25) is even more marked with the starter $E_0 = M$, for which the Newton formulae can lead to divergence.

Conway demonstrates the success of the Laguerre iterator in solving Kepler's equation, with the 'somewhat arbitrary choice' of $N = 5$ in Equation (25). This value is essentially a compromise, since in straightforward cases convergence is usually more rapid with N infinite. In the awkward region (where the Newton formulae fail or are slow), however, much the best results are given with $N = 3$, and the reason for this has been given by Gooding (1987). The essence of the matter is the following: for a function approximating to the polynomial $x^n = A(>0)$, and an iteration step from a gross overestimate of the nth root, the only value of N, in (25), that makes the step a quadratic (as opposed to linear) one is $N = n$; for Kepler's equation, the appropriate value of n is 3, so that $N = 3$ is the natural choice. What happens in practice, using $N = 3$ and a poor initial value (and assuming an awkward-region problem still), is that iteration steps are initially quadratic; the process then slows down for a step or two, but finally homes in with the full rapidity of its nominally cubic nature. When the starter is $E_0 = M$, which underestimates E, the convergence is improved by taking the first step with N infinite, before switching (because E_1 then overestimates E, M being small) to $N = 3$. This is of particular significance, since the first step then has the simple analytical outcome

$$E_1 = M + \frac{e \sin M}{\sqrt{(1 - 2e \cos M + e^2)}}. \tag{26}$$

The right-hand side of (26) is just the starter for EKEPL1; it is an attractive formula, and was first used by Brown (1931).

The conclusions in regard to the Laguerre formulae are as follows. When a good starter is available, they give no advantage. When the starter is poor, on the other hand, there is an immense advantage to be had. The impact on our previous work is that the procedure EKEPL1 is greatly improved if the Halley iterator is replaced by the Laguerre iterator with $N = 3$.

6. Conclusion

We have sought to complement our previous work on Kepler's (elliptic) equation by applying the same philosophy to the hyperbolic equation. A number of solution procedures were developed during the new work, but one of these seems superior in all respects and we unreservedly recommend it. It is based on a reformulation of the equation, such that $\sinh H$ rather than H (hyperbolic mean anomaly) is determined directly; implemented in Fortran-77 under the name SHKEPL, it is listed in Appendix A.

The procedure SHKEPL operates with a starting formula derived by use of Lagrange's expansion theorem and with the quartic iteration process developed in our previous work. Its accuracy is such that, in the absence of rounding error, two iterations (built-in) should give 20 decimal digits correct in all cases. The effects of rounding error have been held to a minimum, such that at most one (decimal) digit of precision should be lost when the computer's word-length does not exceed 20 digits.

During the study, it was recognized that a particular relative-rounding-error defect, identified in the first procedure developed, would also apply to the more accurate of the two solution procedures recommended for the elliptic equation. Though it is scarcely conceivable that the defect could be of consequence in practice, we have developed an alternative procedure, of comparable accuracy but not so efficient; given the name EKEPL, it is listed in Appendix C. The comments of the preceding paragraph (on SHKEPL) apply to EKEPL, except that the nominal worst-case accuracy (after two iterations) is only 14 digits.

For the elliptic equation, which has to be solved so much more often than the hyperbolic equation, we recommend all three solution procedures (the two old ones, EKEPL1 and EKEPL2, and the new one, EKEPL) as being appropriate in different circumstances. A significant improvement in the least sophisticated of them (EKEPL1) can be made, however, if the Halley iterator is replaced by the Laguerre iterator with $N = 3$.

We have not produced a 'universal procedure', for the universally formulated Kepler's equation, because, in spite of the mathematical elegance of such a formulation, we regard it as of little practical value. For numerical work that is accurate and efficient, it will always be necessary to solve different equations for the ellipse, parabola and hyperbola. However, this in no way hinders the development of outwardly universal algorithms for conversion between position and velocity, on the one hand, and a universal set of orbital elements, on the other; such conversion algorithms constitute the subject matter for a parallel paper (Gooding, 1988).

In the context of universal computation, we wish to end with a remark on the recent response by Danby (1987) to our previous comment (Odell and Gooding, 1986) on the disadvantage of using the Stumpff function, $S(x)$, when x corresponds to a multirevolution angle in an elliptic orbit. As Danby observes, the computation must be based on recurrence formulae, after the angle has been reduced (by factors of 4) to a suitable magnitude. It is unfortunate that three of his four recurrence formulae (16) are incorrectly stated, the correct versions being:

$$c_0(4x) = 2[c_0(x)]^2 - 1, \qquad c_1(4x) = c_0(x)c_1(x),$$

$$c_2(4x) = \tfrac{1}{2}[c_1(x)]^2 \quad \text{and} \quad c_3(4x) = \tfrac{1}{4}[c_3(x) + c_1(x)c_2(x)].$$

Considering just the first of these relations, we see that whenever $c_0(x)$ is close to unity, in particular while $|x|$ is small, the rounding error will be roughly quadrupled in each application of the formula. The build-up will not be so rapid as this all the way from $x = 10^{-1}$ to 10^6 (to follow Danby's example), but the overall effect will still be the loss of more than three decimal digits in $c_0(10^6)$. If $\cos E$ is evaluated, on the other hand, with $E = 10^3$ rad (because $x = E^2$), the intrinsic loss (from storage of E itself) is at least one digit less. Danby's technique is certainly viable (a much more rapid build-up of error might have been expected intuitively), but how much simpler and more efficient to recognise an elliptic orbit as such, and apply old-fashioned range reduction!

Appendix A

THE SHKEPL PROCEDURE

```
      DOUBLE PRECISION FUNCTION SHKEPL (EL, G1)
C          EQUATION   EL = SHKEPL + (G1 - 1)*DASINH(SHKEPL),
C          WITH G1 IN RANGE 0 TO 1  INCLUSIVE, SOLVED ACCURATELY.
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      PARAMETER (SW=0.5D0, AHALF=0.5D0, ASIXTH=AHALF/3D0,
     1 ATHIRD=ASIXTH*2D0)
      S = EL
      IF (EL.EQ.0D0) GO TO 2
C1         STARTER BASED ON LAGRANGE'S THEOREM
      G = 1D0 - G1
      CL = DSQRT(1D0 + EL**2)
      AL = DASINH(EL)
      W = G**2*AL/CL**3
      S = 1D0 - G/CL
      S = EL + G*AL/DCUBRT(S**3 + W*EL*(1.5D0 - G/0.75D0))
C2         TWO ITERATIONS (AT MOST) OF HALLEY-THEN-NEWTON PROCESS
      DO 1 ITER=1,2
      S0 = S*S
      S1 = S0 + 1D0
      S2 = DSQRT(S1)
      S3 = S1*S2
      FDD = G*S/S3
      FDDD = G*(1D0 - 2D0*S0)/(S1*S3)
      IF (ASIXTH*S0 + G1 .GE. SW)  THEN
        F = (S - G*DASINH(S)) - EL
        FD = 1D0 - G/S2
       ELSE
        F = SHMKEP(G1, S) - EL
        FD = (S0/(S2 + 1D0) + G1)/S2
      END IF
      DS = F*FD/(AHALF*F*FDD - FD*FD)
      STEMP = S + DS
      IF (STEMP.EQ.S) GO TO 2
      F = F + DS*(FD + AHALF*DS*(FDD + ATHIRD*DS*FDDD))
      FD = FD + DS*(FDD + AHALF*DS*FDDD)
    1 S = STEMP - F/FD
    2 SHKEPL = S
      RETURN
      END
```

Appendix B

AN UNSOPHISTICATED SHMKEP PROCEDURE

```
      DOUBLE PRECISION FUNCTION SHMKEP (G1, S)
C          ACCURATE COMPUTATION OF  S - (1 -  G1)*DASINH(S)
C          WHEN (G1, S) IS CLOSE TO (0, 0)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      G = 1D0 - G1
      T = S/(1D0 + DSQRT(1D0 + S*S))
      TSQ = T*T
      X = S*(G1 + G*TSQ)
      TERM = 2D0*G*T
      TWOI1 = 1D0
    1 TWOI1 = TWOI1 + 2D0
      TERM = TERM*TSQ
      X0 = X
      X = X - TERM/TWOI1
      IF (X.NE.X0) GO TO 1
      SHMKEP = X
      RETURN
      END
```

Appendix C

THE EKEPL PROCEDURE

```
      DOUBLE PRECISION FUNCTION EKEPL(EM, E1)
C          KEPLER'S EQUATION, EM = EKEPL - (1 - E1)*DSIN(EKEPL),
C          WITH E1 IN RANGE 1 TO 0 INCLUSIVE, SOLVED ACCURATELY
C          (BASED ON EKEPL3, BUT ENTERING E1 NOT E)
          IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      PARAMETER (PI=3.1415926535897932384626433832 8D0,TWOPI=2D0*PI,
     1 PINEG=-PI, SW=0.25D0, AHALF=0.5D0, ATHIRD=AHALF/1.5D0)
C1         RANGE-REDUCE EM TO LIE IN RANGE -PI TO PI
      EMR = DMOD(EM,TWOPI)
      IF (EMR.LT.PINEG) EMR = EMR + TWOPI
      IF (EMR.GT.PI) EMR = EMR - TWOPI
      EE = EMR
      IF (EE) 1,4,2
    1 EE = -EE
C          (EMR IS RANGE-REDUCED EM & EE IS ABSOLUTE VALUE OF EMR)
C2         STARTER BY FIRST SOLVING CUBIC EQUATION
    2 E = 1D0 - E1
      W = DCBSOL(E, 2D0*E1, 3D0*EE)
C3         EFFECTIVELY INTERPOLATE IN EMR (ABSOLUTE VALUE)
      EE = (EE*EE + (PI - EE)*W)/PI
      IF (EMR.LT.0D0) EE = -EE
C4         DO TWO ITERATIONS OF HALLEY, EACH FOLLOWED BY NEWTON
      DO 3 ITER=1,2
      FDD = E*DSIN(EE)
      FDDD = E*DCOS(EE)
      IF (EE*EE/6D0 + E1 .GE. SW)  THEN
        F = (EE - FDD) - EMR
        FD = 1D0 - FDDD
       ELSE
        F = EMKEP(E1,EE) - EMR
        FD = 2D0*E*DSIN(AHALF*EE)**2 + E1
      END IF
      DEE = F*FD/(AHALF*F*FDD - FD*FD)
      F = F + DEE*(FD + AHALF*DEE*(FDD + ATHIRD*DEE*FDDD))
C*         TO REDUCE THE DANGER OF UNDERFLOW REPLACE THE LAST LINE BY
C*    W = FD + AHALF*DEE*(FDD + ATHIRD*DEE*FDDD)
      FD = FD + DEE*(FDD + AHALF*DEE*FDDD)
    3 EE = EE + DEE - F/FD
C*         IF REPLACING AS ABOVE, THEN ALSO REPLACE THE LAST LINE BY
C*  3 EE = EE - (F - DEE*(FD - W))/FD
C5         RANGE-EXPAND
    4 EKEPL = EE + (EM - EMR)
      RETURN
      END
```

References

Bergam, M. J. and Prussing, J. E.: 1982, 'Comparison of Starting Values for Iterative Solutions to a Universal Kepler's Equation', *J. Astr. Sci.* **30**, 75–84.

Boltz, F. W.: 1987, 'Inverse Solution of Kepler's Equation for Hyperbolic Orbits', *J. Astr. Sci.* **35**, 347–358.

Brown, E. M.: 1931, 'On a Method of solving Kepler's Equation', *Mon. Not. R. Astron. Soc.* **92**, 104.

Burkardt, T. M. and Danby, J. M. A.: 1983, 'The Solution of Kepler's Equation II', *Celest. Mech.* **31**, 317–328.

Conway, B. A.: 1986, 'An improved Algoritm due to Laguerre for the Solution of Kepler's Equation', *Celest. Mech.*, **39**, 199–211.

Danby, J. M. A.: 1987, 'The Solution of Kepler's Equation, III', *Celest. Mech.* **40**, 303–312.

Gooding, R. H.: 1987, 'Solution of the Hyperbolic Kepler's Equation,' RAE Technical Report 87042.

Gooding, R. H.: 1988, 'On Universal Elements, and Conversion Procedures to and from Position and Velocity', *Celest. Mech.*, this issue, pp. 283–298.

Gooding, R. H. and Odell, A. W.: 1985, 'A Monograph on Kepler's Equation', RAE Technical Report 85080.

Mikkola, S.: 1987, 'A Cubic Approximation for Kepler's Equation', *Celest. Mech.* **40**, 329–334.

Ng, E. W.: 1979, 'A General Algorithm for the Solution of Kepler's Equation for Elliptic Orbits', *Celest. Mech.* **20**, 243–249.

Odell, A. W. and Gooding, R. H.: 1986, 'Procedures for solving Kepler's Equation', *Celest. Mech.* **38**, 307–334.

Shepperd, S. W.: 1985, 'Universal Keplerian State Transition Matrix', *Celest. Mech.* **35**, 129–144.

Whittaker, E. T. and Watson, G. N.: 1940, *A Course in Modern Analysis*, Cambridge, University Press.