

AN EFFICIENT NUMERICAL METHOD FOR HIGHLY OSCILLATORY ORDINARY DIFFERENTIAL EQUATIONS*

LINDA R. PETZOLD†

Abstract. A “quasi-envelope” of the solution of highly oscillatory differential equations is defined. For many problems this is a smooth function which can be integrated using much larger steps than are possible for the original problem. Since the definition of the quasi-envelope is a differential equation involving an integral of the original oscillatory problem, it is necessary to integrate the original problem over a cycle of the oscillation (to average the effects of a full cycle). This information can then be extrapolated over a long (giant!) time step. Unless the period is known a priori, it is also necessary to estimate it either early in the integration (if it is fixed) or periodically (if it is slowly varying). Error propagation properties of this technique are investigated, and an automatic program is presented. Numerical results indicate that this technique is much more efficient than conventional ODE methods, for many oscillating problems.

1. Introduction. Differential equations that have highly oscillatory solutions cannot be solved efficiently using conventional methods. In this paper we will present a method which finds the long-term behavior of the solution to an oscillating problem without following the oscillation closely. This technique can handle linear or nonlinear problems, and it is much faster than conventional methods for many oscillating problems.

Several different authors have proposed algorithms to solve oscillating problems. The multirevolution methods [6], [7], [11], [17], [18], which were first introduced by astronomers in 1957 for calculating the orbits of artificial satellites, are the most closely related to the class of methods presented here. The methods developed here differ from the multirevolution methods in several significant ways. First, those authors were concerned with computing future orbits (oscillations) accurately, whereas we are mainly concerned with “ignoring” the details of the oscillation. Secondly, those methods require some physical reference point (for example, node, apogee, or perigee), whereas with the methods given here there is little need for knowledge of the (physical) origin of the oscillations. Additionally, very little theoretical justification for the multirevolution methods has been attempted.

Several other approaches have been taken to solve these problems. In Gautschi [4], methods are derived which are exact for trigonometric polynomials of a given order if the frequency is known in advance. These methods are applicable for stepsizes the order of the period of the oscillation. Linear systems can be treated by using information from the system eigenvalues to transform the system into another linear system which has highly oscillatory coefficients, but of small amplitude so that they can be neglected, as done by Amdursky and Ziv [1]. Linear and nearly linear systems are also treated by Miranker and Wahba [15], where running averages of the solution are computed. Nonlinear problems of the form

$$\frac{dx}{dt} = \frac{A}{\varepsilon} x + g(t, x), \quad A = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix},$$

$\varepsilon > 0$ small, are dealt with by Miranker and Veldhuizen [14], where the solution is approximated by a series of functions of t/ε . Kreiss [9] approaches the problem by choosing initial values so that the resulting solution is smooth.

* Received by the editors February 2, 1979, and in revised form April 17, 1980.

† Applied Mathematics Division, Sandia Laboratories, Livermore, California. This research was supported in part by the U.S. Department of Energy under contract U.S. ENERGY/EY-76-S-02-2383.

The methods given here compute a smooth function which is conceptually similar to the envelope of an oscillatory function. (For details see § 2.) An approach which is similar to ours, in that the envelope of the oscillations is found, is taken by Miranker and Hoppensteadt [13]. This approach is based on the asymptotic representation of the solution. Alternative ways for determining the asymptotic representation of the solution are found in Auslander and Miranker [2].

The class of methods in this paper is motivated by the observation that when a nearly periodic function is sampled at multiples of the period (or near-period) of the oscillation, the resulting sequence of points changes slowly. If the oscillation is very fast compared to the underlying slowly-varying function, then a smooth curve which passes through these points can be defined. This curve will be called the quasi-envelope, because it is intuitively similar to the envelope of an oscillating function. Since the quasi-envelope varies slowly, it can be followed with large steps. The important property of the quasi-envelope is that it agrees with the solution to the original equation at multiples of the period. Thus, the pointwise solution of the differential equation can be recovered from the quasi-envelope and the differential equation by integrating from any multiple of the period for no more than one cycle.

This technique assumes a knowledge of the period or near-period of the oscillation. An algorithm for finding the period, given a reasonably accurate initial estimate, is developed in § 3. This algorithm is based on minimizing a norm of the difference of the periodic part of the solution and the same part displaced by the period.

Some results concerning the order of accuracy of these methods are given in § 4. Proofs of most theorems are omitted here, and can be found in Petzold [16]. The algorithm has been implemented with variable stepsizes and orders. Details of the implementation are given in § 5 and computational results illustrating the efficiency of this method are provided in § 6.

2. The method. Suppose we are given the initial value problem

$$(2.1) \quad y' = f(y, t), \quad y(0) = y_0, \quad 0 \leq t \leq L,$$

with $y(t)$ periodic or nearly periodic with period T . The variable y may be a vector, as long as all of its components are either nearly periodic with period T , or slowly varying (at least one component of y should be oscillating).

We can define a function $z(t)$ (the quasi-envelope) which describes the long-term behavior of y in terms of the values of z on $[0, T)$ by

$$(2.2a) \quad z(t + T) = z(t) + Tg(z(t), t), \quad 0 \leq t \leq L - T,$$

where

$$(2.2b) \quad g(z, t) = \frac{1}{T} [\tilde{y}(t + T, t) - \tilde{y}(t, t)]$$

and

$$(2.2c) \quad \frac{d}{ds} \tilde{y}(t + s, t) = f(\tilde{y}(t + s, t), t + s), \quad \tilde{y}(t, t) = z.$$

It is easy to see that if $z(0) = y(0)$ then $z(KT) = y(KT)$, $0 \leq KT \leq L$, so that z agrees with y at multiples of the period. Since y is nearly periodic, the values of z at the points $\{KT\}$, K an integer, should change slowly. Solving (2.2a) exactly amounts to solving the differential equation (2.1) over the entire interval $[0, L]$, as $g(z, t)$ is based on the solution of the differential equation over one period. If the length of the interval is very

large, in comparison with the period of the oscillation, this can be very time consuming and costly. Thus, the basis of our method is to compute an approximation to z .

It is possible for many problems to define z on $(0, T)$ so that $z(t)$ changes slowly in $[0, L]$ (see [16]), and then an approximation to z can be computed; or, we may consider z to be a point function, and seek to approximate it only on the set $\{KT\}$.

If the quasi-envelope $z(t)$ is a smooth (slowly varying) function, we can approximate z in terms of its values at infrequent points (at intervals $H \gg T$), thus stepping over many periods of the oscillation at a time. In this paper, we will use formulae which are similar to Adams methods, though other types of formulae are easily derived using the same ideas.

As an example, formulae analogous to the Adams-Moulton methods will be derived. We will use the following relations from operator calculus:

$$\begin{aligned} \nabla z(t) &= z(t) - z(t - H), \\ \nabla &= I - e^{-HD} \quad (D \text{ is the differentiation operator}). \end{aligned}$$

Proceeding formally, we have

$$(2.3) \quad g_n = \left(\frac{e^{TD} - I}{T} \right) z_n,$$

$$(2.4) \quad z_n - z_{n-1} = (I - e^{-HD}) z_n.$$

From (2.3) we have

$$(2.5) \quad z_n = \left(\frac{e^{TD} - I}{T} \right)^{-1} g_n.$$

Substituting (2.5) into (2.4) we obtain

$$z_n - z_{n-1} = (I - e^{-HD}) \left(\frac{e^{TD} - I}{T} \right)^{-1} g_n,$$

or

$$(2.6) \quad z_n - z_{n-1} = \nabla \left(\frac{e^{TD} - I}{T} \right)^{-1} g_n.$$

Now $HD = -\log(I - \nabla)$, so $TD = -(T/H) \log(I - \nabla)$, so that

$$\begin{aligned} \left(\frac{e^{TD} - I}{T} \right)^{-1} &= \left(\frac{e^{-(T/H) \log(I - \nabla)} - I}{T} \right)^{-1} \\ &= \left(-\frac{1}{H} \log(I - \nabla) + \frac{1}{H} \frac{T}{H} \frac{1}{2!} \log^2(I - \nabla) + \dots \right)^{-1}. \end{aligned}$$

Substituting this into (2.6) we obtain

$$\begin{aligned} z_n - z_{n-1} &= \nabla H [\log(I - \nabla)]^{-1} \left(-I + \frac{T}{H} \frac{1}{2!} \log(I - \nabla) - \left(\frac{T}{H} \right)^2 \frac{1}{3!} \log^2(I - \nabla) + \dots \right)^{-1} g_n \\ &= H \left\{ -\nabla [\log(I - \nabla)]^{-1} - \frac{1}{2!} \frac{T}{H} \nabla - \frac{1}{12} \left(\frac{T}{H} \right)^2 \nabla \log(I - \nabla) + \dots \right\} g_n. \end{aligned}$$

Then

$$(2.7) \quad z_n - z_{n-1} = H \left\{ \left[I - \frac{1}{2} \nabla - \frac{1}{12} \nabla^2 - \dots \right] - \frac{1}{2} \frac{T}{H} \nabla + \frac{1}{12} \left(\frac{T}{H} \right)^2 \nabla^2 + \dots \right\} g_n.$$

The terms in square brackets are the Adams-Moulton coefficients, and the others become small as T/H becomes small. This then gives an analogue of the Adams-Moulton formulae in terms of the g_n , on truncating the series at ∇^p . Other such formulae may be derived similarly. These formulae, applied to y , have also been derived in [6], [7], [11], [17] and [18]. Following Graff [6], we will call them generalized Adams formulae. An interesting property of the generalized Adams methods (except for the first order method in (2.7)) is that as $H \rightarrow T$, the coefficients in the formulae approach those of the forward Euler method, which is exact (up to errors made by the method which integrates with small steps through one cycle to find $g(z, t)$) for $T = H$.

A geometrical interpretation of the generalized methods is that they find the "envelope" of the periodic function y . It is easy to see this in Fig. 1, which shows one component of y .

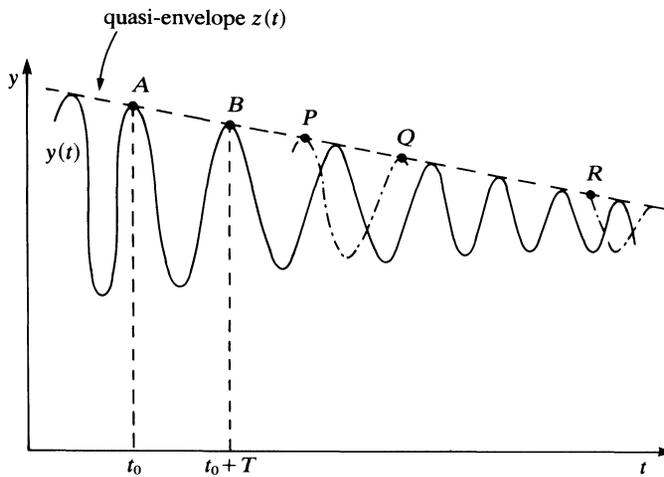


FIG. 1. ODE solution and quasi-envelope.

More precisely, in the method we integrate through one period with small steps to calculate an estimate of the secant line from A to B . (The point A need not be at the top of a cycle, as in the figure.) This secant line is then used (instead of the tangent, as in solving a differential equation), to project the solution to P , which may be many cycles away. Note that, unless H is chosen to be an integral number of periods, the point P may not be anywhere near the solution to the original equation, in the usual sense. Starting with P as the initial value, we integrate for one period along the chain line, using small steps (this solves (2.2c) for \hat{y}). In the picture, the curve that we are finding looks like the upper half of the envelope of the oscillating function.

Some problems have solutions that are composed of oscillations with a slowly varying period, modulated by slowly varying terms. (Here, "period" is defined as the solution to a minimization problem given in § 3.) These problems are handled without having to distinguish them from problems with a constant period of oscillation. This is accomplished by means of a change of independent variable so that in the new variable the period of oscillation is a constant. Then the problem is solved with constant-period formulae, like those just derived.

The change of variables from t to \hat{t} should have the property that the period $T(t)$ of the oscillation becomes a constant τ in the new variable \hat{t} (τ is just a scaling factor). This

is expressed as

$$(2.8) \quad t(\hat{t} + \tau) - t(\hat{t}) = T(t(\hat{t})), \quad t(0) = t_0.$$

We will define $\hat{y}(\hat{t})$ to mean $y(t(\hat{t}))$, and $\hat{z}(\hat{t})$ to mean $z(t(\hat{t}))$. Since we would like to be solving

$$(2.9) \quad \hat{z}(\hat{t} + \tau) = \hat{z}(\hat{t}) + \tau \hat{g}(\hat{z}, \hat{t}),$$

in order to use the constant period formulae, it follows that $\hat{g}(\hat{z}, \hat{t})$ must be defined as

$$(2.10) \quad \hat{g}(\hat{z}, \hat{t}) = \frac{T(t(\hat{t}))}{\tau} g(z(t(\hat{t})), t(\hat{t})).$$

To effect this change of variables, one extra difference equation must be solved. The equations to be solved (instead of (2.2a)) are summarized below.

$$(2.11) \quad \begin{aligned} \hat{z}(\hat{t} + \tau) &= \hat{z}(\hat{t}) + \tau \hat{g}(\hat{z}, \hat{t}), & \hat{z}(0) &= z(t_0), \\ t(\hat{t} + \tau) &= t(\hat{t}) + \tau \left(\frac{T(t(\hat{t}))}{\tau} \right), & t(0) &= t_0. \end{aligned}$$

3. Finding the period. Finding the function $g(z, t)$ in (2.2) involves integrating the differential equation (2.1) over one period (or near-period) of the oscillation. When T is inaccurate, the function $z(t)$, although still defined, may change so rapidly that it might not be possible to follow it with large steps. Therefore, it is important to have an accurate estimate of the period. The procedure described in this section corrects an estimate of the period, once a reasonably accurate initial estimate is given. (In practice, “reasonably accurate” usually means to within about five or ten per cent.)

Suppose we are given a nearly periodic function $y(t)$ defined on an interval $I = [0, L]$, which is the sum of a slowly varying function $r(t)$ and a periodic function $s(t)$ with period T . (In practice, $y(t)$ need not be restricted to have this form, however it is needed for the proof of Theorem 3.1.) By requiring r to be slowly varying, we mean that r does not change much over any interval of length T . Over most of this section, we will assume that y is a scalar, although the procedure is easily extended to a system of functions.

As an example to motivate the algorithm, suppose first that $y(t)$ is periodic with period T . Then $\|y(t) - y(t + T)\|_2 = 0$ on I . This suggests that we might be able to find the period T by minimizing $\|y(t) - y(t + T)\|_2$ over all T in I which are bounded away from zero. Now if y is the solution of a differential equation, it would be impractical to minimize

$$\|y(t) - y(t + T)\|_2 = \int_I (y(t) - y(t + T))^2 dt,$$

with the integral taken over all of I , as that would imply that y is known over all of I . Instead we minimize over the interval from zero to the last estimate of the period. That is, we find

$$\min_{\substack{T^* \cong \epsilon > 0 \\ T^* \in I}} \int_0^{T_m} (y(t) - y(t + T^*))^2 dt,$$

and T_{m+1} is defined as the value of T^* for which the minimum is attained. As there may

be many such values, T_{m+1} will be the one to which the algorithm converges (this will usually be the value closest to T_m).

Some notation which will be used below will now be explained. A function \bar{f} with a bar over it means $f(t + T_m)$ (the subscript m changes after each iteration and should be clear from the context). The symbol f means $f(t)$. All integrals are over the interval $[0, T_m]$ although in practice they may be over any interval of length T_m with the left endpoint fixed throughout the procedure.

In general, we have $y(t) = r(t) + s(t)$, with $r \neq 0$. In this case, we start out with T_0 as before, and with a k -dimensional space of functions $P(k < \infty)$ in which we approximate r . The most obvious space P to take is the space of polynomials of degree $\leq k$. Since a constant shift of a periodic function, that is, $c + s(t)$, is also periodic, the space P should not include constant functions. It is necessary that r may be approximated rather closely over one period by functions in P . At each step of the iteration we would like to find the function P_{m+1} which most closely approximates g on $[0, T_m]$, and a new estimate T_{m+1} of the period. Thus, we must solve two minimization problems at each step.

First we find

$$(3.1) \quad \min_{p_{m+1} \in P} \int_0^{T_m} [y - p_{m+1} - (\bar{y} - \bar{p}_{m+1})]^2 dt.$$

When p_{m+1} is expressed in terms of the basis functions of P , this just leads to k linear equations to determine the coefficients.

Then we find

$$(3.2) \quad \min_{\substack{T_{m+1} \cong \epsilon > 0 \\ T_{m+1} \in I}} \int_0^{T_m} [y - p_{m+1} - (y(t + T_{m+1}) - p_{m+1}(t + T_{m+1}))]^2 dt.$$

At each step of the algorithm a pair $\{p_m, T_m\}$ of quantities which describe r and s are produced.

To describe the computations more precisely, let $\{l_i(t)\}$, $i = 1, \dots, k$ be a basis for P . Write $p_m(t) = \sum_{i=1}^k \alpha_{i,m} l_i(t)$. Then to solve the minimization problem (3.1), take the partial derivatives of the function to be minimized with respect to the coefficients $\alpha_{i,m}$, and set them equal to zero.

$$\frac{\partial}{\partial \alpha_{j,m+1}} \int_0^{T_m} \left(y - \sum_{i=1}^k \alpha_{i,m+1} l_i - \bar{y} + \sum_{i=1}^k \alpha_{i,m+1} \bar{l}_i \right)^2 dt = 0, \quad j = 1, \dots, k.$$

This leads to

$$(3.3) \quad \sum_{i=1}^k \alpha_{i,m+1} \int_0^{T_m} (\bar{l}_i - l_i)(\bar{l}_j - l_j) dt = \int_0^{T_m} (\bar{l}_j - l_j)(\bar{y} - y) dt, \quad j = 1, \dots, k,$$

which is a set of k linear equations that we can solve for the $\alpha_{i,m+1}$, $i = 1, \dots, k$. These equations are nonsingular as long as P excludes periodic functions.

To determine the minimum in (3.2) take the partial derivative of the function to be minimized, with respect to T_{m+1} , and set that equal to zero

$$(3.4) \quad \int_0^{T_m} (\bar{p}'_{m+1} - \bar{y}') (y - p_{m+1} - \bar{y} + \bar{p}_{m+1}) dt = 0.$$

This is a nonlinear equation in the variable T_{m+1} which can be solved by Newton's method if the initial estimate T_0 is close enough to the minimum. This is partly why T_0 needs to be a good estimate, and why the period may be allowed to change only slowly.

This leads to the iteration (one step of (3.5) is done per iteration consisting of (3.3) followed by (3.5)):

$$(3.5) \quad T_{m+1} = T_m - \left[\frac{\int_0^{T_m} (\bar{p}'_{m+1} - \bar{y}') (y - p_{m+1} - \bar{y} + \bar{p}_{m+1}) dt}{\int_0^{T_m} (\bar{p}''_{m+1} - \bar{y}'') (y - p_{m+1} - \bar{y} + \bar{p}_{m+1}) dt + \int_0^{T_m} (\bar{p}'_{m+1} - \bar{y}')^2 dt} \right].$$

The algorithm can be summarized as follows. First, an initial estimate T_0 for the period is given. Then (3.3) is solved for an approximation $p_1(t)$ to $r(t)$. Finally, T_1 is found by (3.5). This process is repeated until the sequence of approximations to the period is determined to have converged. This is not Newton's method for (3.4) because at each iteration m changes, so that p_m changes. It would be possible to fix p_{m+1} in (3.4) and iterate that equation for T_{m+1} until it converges each time, but this appears to be slightly less efficient.

The integrals needed in the iteration can be computed using any sufficiently accurate quadrature formula. They are easily computed using Riemann sums. This procedure may be extended in an obvious way to a system of functions.

In practice, this algorithm converges to the period or near period for a wide range of functions $y(t)$. The following theorem, which is proved in [16], shows that it is locally convergent for a certain class of functions.

THEOREM 3.1. *If $y = r + s$, where $r \in P$ and s is periodic with period T , the iteration defined by (3.3) and (3.5) is stable, if we start within a sufficiently small neighborhood of r and T .*

Using this technique to compute the period, the generalized methods can then be used to solve approximately the resulting equations for the quasi-envelope $z(t)$, with large steps.

4. Error propagation. Errors are introduced into the solution during several phases of the computation. The function $g(z, t)$ in (2.2b) is computed by integrating over one cycle of the oscillation with a numerical method using small steps. The effects of the errors introduced in this computation can be seen by introducing new functions $\hat{z}(t)$ and $\hat{g}(\hat{z}, t)$, which depend on the numerical solution $\hat{y}(t)$ by the small-step method over the entire interval $[0, L]$, in the same way that $z(t)$ and $g(z, t)$ depend on the true solution $y(t)$. Under a few assumptions (the period $T(t)$ is known in advance, and $\hat{y}(t)$ is computed by re-starting at the beginning of each period with the values at the end of the last period), the quasi-envelope $\hat{z}(t)$ of the numerical solution agrees with $\hat{y}(t)$ at multiples of the period. Because the generalized methods compute an approximation to $\hat{z}(t)$ their accuracy is limited by the global errors of the small-step method. Thus, the solution computed by the generalized methods is, in general, no more accurate than the solution would be if the small-step method were used to integrate the problem over the entire interval.

The generalized methods solve (2.2a) approximately, stepping over many periods at a time. This is another source of errors, which we will investigate in this section. In doing so, we will extend the concepts of order, stability and convergence to the generalized methods.

Throughout this section, we will assume that the period T is a constant, or has been transformed to a constant by a change of variables, and that all implicit equations are solved exactly. For simplicity, we will consider only problems involving a single

differential equation, though the results are easily extended to systems of equations. Additionally, the stepsize H is assumed to be an integral multiple of the period.¹

We can now proceed to investigate the errors due to the generalized methods. To begin, let $H = NT$ (N an integer). The solution $y(t)$ of the original differential equation and the choice of T define a set of functions $\{g_{k/N}(z)\}$ at multiples of the period, $t_{k/N} = kT = (k/N)H$, where

$$(4.1) \quad g_{k/N}(z) = g(z, t_{k/N}) = \frac{1}{T} \Delta z(t_{k/N}) = \frac{1}{T} (z(t_{(k+1)/N}) - z(t_{k/N})).$$

Note that the symbol Δ is used here to denote the forward difference over an interval of length T . We will assume that the functions $g_{k/N}(z)$ satisfy

$$(4.2) \quad \left| \frac{\partial^j \Delta^i g_{k/N}(z)}{\partial z^j} \right| \leq C_{ij} (CT^i), \quad \begin{cases} j = 0, i = 0, 1, \dots, N, \\ j = 1, i = 0, \end{cases}$$

and $C_{ij} \leq k$, where

$$\Delta g_{k/N}(z) = g_{(k+1)/N}(z) - g_{k/N}(z),$$

and the higher order differences are defined similarly.

These conditions are analogous to conditions which bound derivatives for ordinary differential equations. For $j = 0$, (4.2) requires for each problem that the secant lines of length T at multiples of T change slowly along the solution and along integral curves near the solution. For $j = 1$ it requires that changes in $g_{k/N}(z)$ are not rapid with respect to changes in z . Thus, the solution over one period must depend slowly on the value of z at the beginning of the period. This condition has the effect of constraining the multiplier of the error at each (large) step to be small. If integral curves near the solution are nearly periodic (with period T), then it is likely that these conditions are satisfied.

It appears to be a difficult problem to determine whether a problem satisfies the conditions (4.2) from the original differential equation alone. Using some qualitative information about the behavior of the solution and curves near the solution we can at least decide whether it is plausible that the conditions are satisfied.

With these assumptions we can compute bounds on the error due to the generalized methods without having to bound derivatives of g and z with respect to t . If $g(z, t)$ is nearly periodic with period T , then $\{g_{k/N}(z)\}$ is a slowly changing sequence of functions.

To gain some insight into how errors for these methods might be analyzed, we will first take a closer look at the simplest method, the generalized forward Euler method.

Suppose that z_n is the solution computed by the generalized forward Euler method. Then we have, for $t_n = nH$,

$$(4.3) \quad z_{n+1} = z_n + Hg(z_n, t_n).$$

In addition, it is easy to see that

$$(4.4) \quad z(t_{n+1}) = z(t_n + NT) = \sum_{i=0}^N \binom{N}{i} \Delta^i z(t_n).$$

¹ The case when H is not constrained to be a multiple of the period is discussed in Petzold [16]. It is possible, for many oscillating problems, to define a sequence of functions $\{z_l(t)\}$ which approximate z at the points $\{kT\}$, k an integer, to order T^l . Then z can be defined on all of $[0, L]$ as a smooth, continuous function, for small T in terms of the z_l . Bounds can be found on the distance between the numerical solution and z_l for some l . In this way, the difference between the numerical solution and the true solution can be bounded. While the practice of not constraining H to be a multiple of the period is not, in general, recommended, it is important to consider this case because when $T(t)$ is varying, phase errors introduced in the solution of (2.11) may not be negligible.

Letting $e_n = z(t_n) - z_n$, and subtracting (4.3) from (4.4), we have

$$(4.5) \quad e_{n+1} = e_n + NT[g(z(t_n), t_n) - g(z_n, t_n)] + \frac{N(N-1)}{2} T \Delta g(z(t_n), t_n) + \dots + T \Delta^{N-1} g(z(t_n), t_n).$$

Then, using (4.2), we get

$$(4.6) \quad |e_{n+1}| \leq |e_n|(1 + NTC_{01}) + D, \quad e_0 = 0,$$

where

$$D = \sum_{i=2}^N \binom{N}{i} (CT)^{i-1} C_{i-1,0} T.$$

Note that e_n is identically zero if $N = 1$, as we would expect. In fact, this is true for all of the generalized Adams formulae, except the generalized backward Euler method.

We will investigate convergence by means of a sequence $\{P_q\}$ of problems where the period T_q becomes arbitrarily small as $q \rightarrow \infty$ without changing the true solution much.

We can define functions $\{g_{k/N_q, q}\}, \{z_q\}$ for the q th problem just as in the case of a single problem. We consider only sequences where $\{g_{k/N_q, q}\}$ and $\{z_q\}$ satisfy condition (4.2). That is,

$$(4.7) \quad \left| \frac{\partial^j (\Delta^i g_{k/N_q, q}(z_q))}{\partial z_q^{(j)}} \right| \leq C_{ij} (CT_q)^i, \quad \begin{cases} j = 0, & i = 0, \dots, N_q, \\ j = 1, & i = 0, \end{cases}$$

and $C_{ij} \leq K$, where

$$(4.8) \quad H = N_q T_q$$

and C_{ij} is independent of the problem q .

One might visualize the sequence of problems as having the same “envelope” (although, it is a little less restrictive than that), where the oscillations “inside” become more and more dense as q becomes larger.

If we were solving an equation which, in the conventional sense, had for its solution the “envelope”, we would choose the stepsize H to compute the solution to the desired error tolerance. In investigating convergence, we would let $H \rightarrow 0$ and see whether the computed solution tends to the true solution. Here, for one problem, $H \geq T_q$, so we consider solving all of the problems satisfying (4.7) for which $H \geq T_q$, and for q large we look at how close the computed solution is to the true quasi-envelope.

We will now proceed with the discussion of the generalized forward Euler method. Recall that (4.6) bounded $|e_{n+1}|$ in terms of $|e_n|$ and $D_{n,q}$. We can bound the solutions of that recurrence by the usual methods to obtain for the q th problem,

$$(4.9) \quad |e_{n,q}| \leq \frac{D_{n,q}(e^{C_{0,1}L} - 1)}{N_q T_q C_{0,1}},$$

for $0 \leq nH \leq L$. (Note the notation. The letter q will always refer to the q th problem. The letter n will always refer to the time $t_n = nH$. When the method is applied to a single

problem, the subscript q will not appear.) Now

$$D_{n,q} = \sum_{i=2}^{N_q} \binom{N_q}{i} (CT_q)^{i-1} C_{i-1,0} T_q \leq \frac{K}{C} [(1 + CT_q)^{N_q} - (N_q CT_q + 1)].$$

If H is a constant, and $N_q T_q = H$, then we have

$$D_{n,q} \leq \frac{K}{C} \left[\left(1 + \frac{CH}{N_q} \right)^{N_q} - (CH + 1) \right] \leq \frac{K}{C} (e^{CH} - (CH + 1)).$$

Thus, for H fixed and for all q such that $H \geq T_q$,

$$(4.10) \quad |e_{n,q}| \leq \frac{K(e^{CH} - (CH + 1))(e^{C_0,1L} - 1)}{CHC_{0,1}}.$$

Since we can make H arbitrarily small, with $H \geq T_q$, by taking q large enough, (4.10) implies that $|e_{n,q}|$ is $O(H)$ in the sense that for all q such that $H \geq T_q$, $|e_{n,q}| \leq KH$, $0 < nH \leq L$. The constant K does not depend on H or q .

In general, for any method, the statement “ $|e_{n,q}|$ is $O(H^p)$ ” will mean that for any sequence of problems satisfying (4.7), there are positive constants r_0 and H_0 such that for each $H \in (0, H_0]$, if $T_q/H \leq r_0$ and $0 < nH \leq L$, then $|e_{n,q}| \leq \hat{K}H^p$. The constant \hat{K} is independent of H and q , and r_0 depends only on the method.

In this section, a generalized l -step method will have the form

$$(4.11) \quad \sum_{i=0}^l \left(\alpha_i \left(\frac{1}{N} \right) z_{n+i} + H\beta_i \left(\frac{1}{N} \right) g(z_{n+i}, t_{n+i}) \right) = 0,$$

where $H = NT$, z_n is the solution computed at time t_n , and $t_n = nH$. The formula (2.7) truncated provides examples of some of these methods. For purposes of discussing local truncation error, we will assume that (4.11) is normalized so that $\sum_{i=0}^l \beta_i (1/N) = 1$.

$e_{n,q}^{(H)}$ is the (global) error of the method, applied to the q th problem at time t_n with stepsize H . That is, it is the difference between the solution computed by the method and the true quasi-envelope. Analogously to ordinary differential equations, we have the following definition:

DEFINITION 4.1. A generalized l -step method is *convergent* if there is a constant $r_0 > 0$ such that for any sequence of problems satisfying (4.7), we have

$$\lim_{H \rightarrow 0} \sup_{\{q | T_q/H \leq r_0\}} e_{n,q}^{(H)} = 0,$$

uniformly for $0 \leq nH \leq L$.

Thus, a method is convergent if $|e_{n,q}|$ is $O(H^p)$ with $p > 0$.

To investigate the convergence of generalized l -step methods we will first examine several other properties of the methods, which are analogous to similar properties of methods for ordinary differential equations.

A linear multistep method is stable if for any differential equation satisfying a Lipschitz condition on an interval $[0, L]$, a small perturbation in the initial values causes a bounded change in the numerical solution as $H \rightarrow 0$, with $nH = L$, where n is the number of steps taken. One way to guarantee this is to require the polynomial

$$\rho(\xi) = \sum_{i=0}^l \alpha_i \xi^i$$

to satisfy the root condition. That is, the roots of $\rho(\xi) = 0$ must be inside the unit circle, or on the unit circle and simple.

Analogously, for the generalized linear l -step method (4.12) we have the following definition.

DEFINITION 4.2. A method of the form (4.11) is *uniformly stable* if the polynomials

$$\rho_r(\xi) = \sum_{i=0}^l \alpha_i(r)\xi^i, \quad r = \frac{1}{N}$$

satisfy the root condition for all r in some interval $[0, r_0]$, and $\alpha_i(r)$ and $\beta_i(r)$ are continuous functions of r on $[0, r_0]$, ($r_0 > 0$).

Another concept from ordinary differential equations that will be useful here is that of order.

DEFINITION 4.3. A method (4.11) is of *order* p if

$$\sum_{i=0}^l \left[\alpha_i\left(\frac{1}{N}\right)z(t_{n+i}) + \frac{H}{T}\beta_i\left(\frac{1}{N}\right)\Delta z(t_{n+i}) \right] = 0$$

whenever z is a polynomial of degree p , for all r ($r = 1/N$) such that $0 \leq r \leq r_0$. (As usual, Δz is the forward difference of z over an interval of length T .) A method of order p , with $p \geq 1$, will be called *consistent*.

The following theorem tells us that a uniformly stable and consistent generalized l -step method is convergent.

THEOREM 4.1. For a sequence of problems satisfying conditions (4.7), the error $e_{n,q}$ of a uniformly stable generalized l -step method of order p , with H sufficiently small, is $O(H^p)$.

Proof. First, we will find a bound for $D_{n,q}$, the local error in solving the q th problem with a stepsize H . Let $H = T_q N_q$. Then

$$(4.12) \quad D_{n,q} = \sum_{i=0}^l \left[\alpha_i\left(\frac{1}{N_q}\right)z_q(t_{n+i}) + N_q\beta_i\left(\frac{1}{N_q}\right)\Delta z_q(t_{n+i}) \right]$$

$$(4.13) \quad = \sum_{i=0}^l \sum_{j=0}^{iN_q} \left[\alpha_i\left(\frac{1}{N_q}\right)\Delta^j z_q(t_n) + N_q\beta_i\left(\frac{1}{N_q}\right)\Delta^{i+1} z_q(t_n) \right] \binom{iN_q}{j}$$

$$(4.14) \quad = \sum_{j=0}^{iN_q+1} [Q_{j,q}]\Delta^j z_q(t_n),$$

where

$$Q_{j,q} = \sum_{i=0}^l \left[\alpha_i\left(\frac{1}{N_q}\right)\binom{iN_q}{j} + N_q\beta_i\left(\frac{1}{N_q}\right)\binom{iN_q}{j-1} \right].$$

By the definition of order p , $Q_{j,q} = 0$ for $j = 0, 1, \dots, p$. The first nonzero term, $Q_{p+1,q}\Delta^{p+1} z_q(t)$, is the principal part of the local truncation error. Deleting the vanishing terms we have

$$(4.15) \quad |D_{n,q}| \leq \left| \sum_{i=0}^l \left\{ \sum_{j=p+1}^{iN_q} \alpha_i\left(\frac{1}{N_q}\right)\Delta^j z_q(t_n)\binom{iN_q}{j} + \sum_{j=p}^{iN_q} N_q\beta_i\left(\frac{1}{N_q}\right)\Delta^{i+1} z_q(t_n)\binom{iN_q}{j} \right\} \right|.$$

Since $\Delta z_q(t_k) = T_q g(z_q(t_k), t_k)$, we have from (4.7) that

$$|\Delta^i z_q(t_n)| \leq T_q K (CT_q)^{i-1}.$$

Substituting this into (4.15) and using the triangle inequality, we obtain

$$(4.16) \quad |D_{n,q}| \leq \sum_{i=0}^l \left\{ \left| \alpha_i \left(\frac{1}{N_q} \right) \right| \frac{K}{C} \sum_{j=p+1}^{iN_q} (CT_q)^j \binom{iN_q}{j} + N_q T_q K \left| \beta_i \left(\frac{1}{N_q} \right) \right| \sum_{j=p}^{iN_q} (CT_q)^j \binom{iN_q}{j} \right\}.$$

Now, since $\binom{iN}{j} \leq (iN)^j / j!$, we have that

$$(4.17) \quad \sum_{j=p}^{iN_q} (CT_q)^j \binom{iN_q}{j} \leq \sum_{j=p}^{iN_q} \frac{(iN_q CT_q)^j}{j!} \leq e^{iN_q CT_q} \frac{(iN_q CT_q)^p}{p!}.$$

Now (4.16) and (4.17) together imply that

$$(4.18) \quad |D_{n,q}| \leq \sum_{i=0}^l \left\{ \frac{K}{C} \left| \alpha_i \left(\frac{1}{N_q} \right) \right| e^{iCH} \frac{(iCH)^{p+1}}{(p+1)!} + HK \left| \beta_i \left(\frac{1}{N_q} \right) \right| e^{iCH} \frac{(iCH)^p}{p!} \right\}.$$

For q sufficiently large this bound can be made independent of q by continuity of $\alpha_i(r)$ and $\beta_i(r)$ on $[0, r_0]$.

Now that we have a bound for $D_{n,q}$, the next steps are almost the same as the standard proofs in ODEs. (See, for example, Henrici [8, Theorem 5.11].) The basic idea is to find a difference equation defining the global error, and then use the root condition and the bound on $D_{n,q}$ to bound the solutions of the difference equation by a bound which is a constant times H^p . The principal tools that are generally used, and that we will use here, are variants of Lemmas 5.5 and 5.6 of Henrici [8, pp. 242–244]. It is important to notice that many of the constants in these lemmas can be bounded independently of r (where $r = T/H$). In particular, Lemma 5.5 says that if $\rho(\xi)$ satisfies the root condition, and $\gamma_l, l = 0, 1, 2, \dots$, are defined by

$$\frac{1}{\alpha_k + \alpha_{k-1}\xi + \dots + \alpha_0\xi^k} = \gamma_0 + \gamma_1\xi + \gamma_2\xi^2 + \dots,$$

then $\Gamma = \sup_{l=0,1,\dots} |\gamma_l| < \infty$. Examining the proof it is easy to see that it will hold for ρ, r sufficiently small, and that $\Gamma(r)$ is a continuous function of r . It follows by compactness that $\Gamma(r)$ is bounded in $[0, r_0]$. Lemma 5.6 concerns the growth of solutions of

$$(4.19) \quad \begin{aligned} &\alpha_k \omega_{m+k} + \alpha_{k-1} \omega_{m+k-1} + \dots + \alpha_0 \omega_m \\ &= h \{ \beta_{k,m} \omega_{m+k} + \beta_{k-1,m} \omega_{m+k-1} + \dots + \beta_{0,m} \omega_m \} + \lambda_m. \end{aligned}$$

If ρ satisfies the root condition, and B^*, β , and Λ are nonnegative constants such that

$$|\beta_{k,n}| + |\beta_{k-1,n}| + \dots + |\beta_{0,n}| \leq B^*, \quad |\beta_{k,n}| \leq \beta, \quad |\lambda_n| \leq \Lambda, \quad n = 0, 1, 2, \dots, N,$$

and $0 \leq h < |\alpha_k| \beta^{-1}$, then every solution of (4.19) for which $|\omega_n| \leq z, \mu = 0, 1, \dots, k-1$ satisfies $|\omega_n| \leq K^* e^{nhL^*}, n = 0, 1, \dots, N$, where $L^* = \Gamma^* B^*, K^* = \Gamma^*(N\Lambda + AZk)$,

$$A = |\alpha_k| + |\alpha_{k-1}| + \dots + |\alpha_0|, \quad \Gamma^* = \frac{\Gamma}{1 - h|\alpha_k|^{-1}\beta}.$$

By uniform stability and compactness, β and A can be bounded independently of r on $[0, r_0]$, and we can bound the solutions of (4.19) independently of q and r for $T_q/H \leq r_0$. Proceeding with this we define the global error $e_{n,q}$ by

$$(4.20) \quad e_{n,q} = z_q(t_n) - z_{n,q},$$

where $z_{n,q}$ is the solution to the q th problem at time t_n by the method (4.11). Then $e_{n,q}$

satisfies

$$\sum_{i=0}^l \left[\alpha_i \left(\frac{1}{N_q} \right) e_{n+i,q} + N_q T_q \beta_i \left(\frac{1}{N_q} \right) \left(g_{n+i,q}(z_q(t_{n+i})) - g_{n+i,q}(z_{n+i,q}) \right) \right] = D_{n,q}.$$

Using the mean value theorem, we have

$$g_{n,q}(z_q(t_n)) - g_{n,q}(z_{n,q}) = \frac{\partial g_{n,q}(z_q)}{\partial z_q} \Big|_{(\xi_n, t_n)} e_{n,q}.$$

So this gives

$$\sum_{i=0}^l \left[\alpha_i \left(\frac{1}{N_q} \right) e_{n+i,q} + H \beta_i \left(\frac{1}{N_q} \right) \frac{\partial g_{n+i,q}}{\partial z_q} \Big|_{(\xi_{n+i}, t_{n+i})} e_{n+i,q} \right] = D_{n,q}.$$

Now it is a simple (but tedious) matter to use Lemma 5.6 in Henrici [8], plus uniform stability, the bound (4.16) and the conditions (4.7) to show that for

$$0 \leq H < \frac{\alpha \beta^{-1}}{K}, \quad \text{where } \left| \alpha_l \left(\frac{1}{N_q} \right) \right| \geq \alpha \text{ and } \left| \beta_l \left(\frac{1}{N_q} \right) \right| \leq \beta,$$

we have that $|e_{n,q}|$ is $O(H^p)$. \square

It is easily seen that the generalized Adams formulae are uniformly stable, because $\rho_{1/N}(\xi)$ is independent of N , so that it is the same as $\rho(\xi)$ for conventional Adams formulae, which satisfies the root condition. In addition, the coefficients $\{\beta_i(1/N)\}$ are polynomials in $1/N$, and hence are uniformly bounded for $0 \leq 1/N \leq r_0$.

It is also easily found that the generalized Adams formula which is of order p when $T = 0$ is of order p for $T \neq 0$ (it is exact for polynomials of degree $\leq p$). Thus, we have trivially from Theorem 4.1:

COROLLARY 4.1. *The generalized Adams-Moulton and Adams-Bashforth formulae of order p are convergent, and $|e_{n,q}|$ is $O(H^p)$.*

In summary, the generalized methods are limited in accuracy by the small-step method used to compute the solution over one period. By choosing the period, and stepsize and order of the generalized method appropriately to follow the smooth solution, computation of the solution can be speeded up a great deal, with little sacrifice in accuracy.

More precisely, if z^c is the computed solution and \hat{z} is the quasi-envelope of the numerical solution computed by the small-step method, we have

$$(4.21) \quad \|z^c(nNT) - y(nNT)\| \leq \|z^c(nNT) - \hat{z}(nNT)\| + \|\hat{z}(nNT) - y(nNT)\|.$$

For nNT near L , the first term on the right-hand side represents the global error due to the large-step discretization, analogous to e_n . The second term represents the global error of the small-step method over $[0, nNT]$. In practice, we choose the parameters of the small-step method as if it alone were solving the problem, and the parameters of the generalized method are chosen to compute the smooth quasi-envelope $z(t)$ to the desired error tolerance.

5. Implementation considerations. The development of an automatic program implementing the generalized Adams methods will be considered in this section.

Because of the similarities between the generalized Adams formulae and conventional Adams formulae, it is relatively straightforward to implement the generalized Adams methods as a predictor-corrector process. In fact, this can be accomplished by making a few changes in a code which implements the conventional Adams methods.

A data organization scheme which is similar to Nordsieck's form of the Adams method will be described. This scheme enables us to change stepsize and order easily.

Using the change of variables described in § 2, append t onto the end of z . Then the equations to be solved can be written as

$$(5.1) \quad \mathbf{z}(\hat{t} + \tau) = \mathbf{z}(\hat{t}) + \tau g(\mathbf{z}(\hat{t})),$$

where

$$g(\mathbf{z}(\hat{t})) = [\hat{g}_1, \hat{g}_2, \dots, \hat{g}_{s+1}]^T$$

and $\hat{g}_{s+1} = T(t(\hat{t}))/\tau$. This change of variables will be assumed. It is useful to notice that the $(s+1)$ st component of g is the estimate for the period $T(t)$ (divided by τ) at each step. Thus, computation of $T(t)$ is treated like a function evaluation. The predictor equation is used to form an initial estimate for the period at each step (except the first step, where the user-supplied estimate is used), which is then corrected by the Newton iteration of § 3.

The procedures described here can be applied to the vector \mathbf{z} simply by applying them individually to each component of \mathbf{z} at every time step, so we consider only the scalar difference equation

$$(5.2) \quad z(t+T) = z(t) + Tg(z, t).$$

In this section the generalized l -step explicit method will be written as

$$(5.3) \quad z_n = \sum_{i=1}^l (\alpha_i(r)z_{n-i} + H\beta_i(r)g_{n-i}),$$

where $r = T/H$. The generalized l -step implicit method is given by

$$(5.4) \quad z_n = \sum_{i=1}^l (\alpha_i^*(r)z_{n-i} + H\beta_i^*(r)g_{n-i}) + H\beta_0^*(r)g(z_n).$$

Using the formulae in (5.3) and (5.4) as predictor and corrector, we have

$$(5.5) \quad \begin{aligned} z_{n,(0)} &= \sum_{i=1}^l (\alpha_i(r)z_{n-i} + H\beta_i(r)g_{n-i}), \\ z_{n,(m+1)} &= \sum_{i=1}^l (\alpha_i^*(r)z_{n-i} + H\beta_i^*(r)g_{n-i}) + H\beta_0^*(r)g(z_{n,(m)}). \end{aligned}$$

We will use the generalized Adams-Bashforth formula of order k as the predictor ($k = l$), and the generalized Adams-Moulton formula of order k as the corrector. These formulae require values of z and g , which can be stored in the vector \mathbf{w}_n ,

$$(5.6) \quad \mathbf{w}_n = [z_n, Hg_n, \dots, Hg_{n-l+1}]^T.$$

It is easier to compare different organizations of the computation if (5.5) is rewritten in terms of the vector \mathbf{w}_n . Then the prediction step is given by

$$(5.7) \quad \mathbf{w}_{n,(0)} = B\mathbf{w}_{n-1},$$

where, for the generalized Adams methods,

$$(5.8) \quad B = \begin{bmatrix} 1 & \beta_1(r) & \beta_2(r) & \cdots & \beta_k(r) \\ 0 & \gamma_1(r) & \gamma_2(r) & \cdots & \gamma_k(r) \\ & 1 & & & \\ & & 1 & & \\ & & & 1 & 0 \end{bmatrix}.$$

Now, $\beta_i(r)$, $i = 1, \dots, k$ are the coefficients of the generalized Adams-Bashforth method of order k . $\gamma_i(r)$, $i = 1, \dots, k$ are given by

$$\gamma_i(r) = \frac{\beta_i(r) - \beta_i^*(r)}{\beta_0^*(r)},$$

where $\beta_i^*(r)$, $i = 0, 1, \dots, k$ are the coefficients of the generalized Adams-Moulton formula of order k .

The corrector iteration in (5.5) will then be rewritten as

$$(5.9) \quad \mathbf{w}_{n,(m+1)} = \mathbf{w}_{n,(m)} + \mathbf{c}G(\mathbf{w}_{n,(m)}),$$

for $m = 0, 1, \dots, M$. $G(\mathbf{w}_n)$ is defined by

$$(5.10) \quad G(\mathbf{w}_n) = -Hg_n + Hg(z_n).$$

Here, Hg_n is the second component of \mathbf{w}_n , and z_n is the first component of \mathbf{w}_n . $G(\mathbf{w}_n)$ is the amount by which \mathbf{w}_n fails to satisfy (5.2). The vector \mathbf{c} is chosen so that (5.7) followed by (5.9) is equivalent to the iteration defined by (5.5). For the generalized Adams methods

$$(5.11) \quad \mathbf{c} = [H\beta_0^*, 1, 0, \dots, 0]^T.$$

Table 5.1 lists expressions for $\beta_0^*(r)$ for generalized Adams-Moulton formulae.

TABLE 5.1.
Adams-Moulton coefficients— $\beta_0^*(r)$.

k (Order)	$\beta_{0,k}^*(r)$ for Order k
1	1
2	$\frac{1}{2}(1-r)$
3	$\frac{5}{12} - \frac{6}{12}r + \frac{1}{12}r^2$
4	$(9 - 12r + 3r^2)/24$
5	$(251 - 360r + 110r^2 - r^4)/720$
6	$(475 - 720r + 250r^2 - 5r^4)/1440$
7	$(19,087 - 30,240r + 11,508r^2 - 357r^4 + 2r^6)/60,480$
8	$(36,799 - 60,480r + 24,696r^2 - 1029r^4 + 14r^6)/120,960$
9	$(1,070,017 - 1,814,400r + 784,080r^2 - 40,614r^4 + 425r^6 - 3r^8)/3,628,800$
10	$(2,082,753 - 3,628,800r + 1,643,760r^2 - 100,926r^4 + 2250r^6 - 27r^8)/7,257,600$
11	$(134,211,265 - 239,500,800r + 112,923,360r^2 - 7,960,480r^4 + 266,090r^6 - 4785r^8 + 10r^{10})/479,001,600$

Changing the stepsize from H_{n-1} to H_n is inconvenient when using the representation (5.6) of the past values of z and g . Changing the stepsize by a ratio $R_n = H_n/H_{n-1}$ corresponds to premultiplying \mathbf{w}_{n-1} by a matrix $C(R_n)$. $C(R_n)$ computes the values of g at the points $(t_n, t_n - H_n, \dots, t_n - (n - k + 1)H_n)$ from the values of g at the points $(t_n, t_n - H_{n-1}, \dots, t_n - (n - k + 1)H_{n-1})$ by interpolation. The step-changing operation is represented by

$$(5.12) \quad \tilde{\mathbf{w}}_{n-1} = C(R_n)\mathbf{w}_{n-1}.$$

The step-changing matrix $C(R_n)$ for the vector \mathbf{w}_n defined by (5.6) is nontrivial. To find a more convenient representation for the past values of z and g , note that the vector

w_n uniquely determines a polynomial $w_n(t)$ of degree k which satisfies

$$(5.13) \quad \begin{aligned} w_n(t_n) &= z_n, \\ (H/T)[w_n(t_n + T) - w_n(t_n)] &= Hg_n, \\ (H/T)[w_n(t_{n-k+1} + T) - w_n(t_{n-k+1})] &= Hg_{n-k+1}. \end{aligned}$$

Instead of representing $w_n(t)$ in terms of $(z_n, Hg_n, \dots, Hg_{n-k+1})$, we can equivalently store

$$(5.14) \quad \mathbf{a}_n = \begin{bmatrix} w_n(t_n) \\ (H/T)\Delta w_n(t_n) \\ (H^2/2T)\Delta w'_n(t_n) \\ \vdots \\ (H^k/k!T)\Delta w_n^{(k-1)}(t_n) \end{bmatrix},$$

where $\Delta w_n(t) = w_n(t + T) - w_n(t)$. This is similar to Nordsieck's form of Adams method, except that one of the derivatives has been replaced by Δ .

We can write (5.7) and (5.9) in terms of the representation (5.14). Since \mathbf{a}_n and \mathbf{w}_n represent the same polynomials, they are related by the linear transformation

$$(5.15) \quad \mathbf{a}_n = S\mathbf{w}_n.$$

For example, when $k = 2$,

$$\begin{bmatrix} w_n(t_n) \\ (H/T)\Delta w_n(t_n) \\ (H^2/2T)\Delta w'_n(t_n) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \frac{1}{2} & -\frac{1}{2} \end{bmatrix} \begin{bmatrix} w_n(t_n) \\ (H/T)\Delta w_n(t_n) \\ (H/T)\Delta w_n(t_{n-1}) \end{bmatrix}.$$

S is independent of T/H for all k .

With the past information stored in \mathbf{a}_n , the predictor becomes

$$(5.16) \quad \mathbf{a}_{n,(0)} = A\mathbf{a}_{n-1},$$

where $A = SBS^{-1}$. A can also be determined by using the fact that the predictor is of order k . That is, if $w(t)$ is any k th order polynomial, and

$$\mathbf{w}(t_n) = [w(t_n), (H/T)\Delta w(t_n), H^2/(2T)\Delta w'(t_n), \dots, H^k/(k!T)\Delta w^{(k-1)}(t_n)]^T,$$

then

$$(5.17) \quad \mathbf{w}(t_n) = A\mathbf{w}(t_{n-1}).$$

Since rows 2 through $(k + 1)$ of A extrapolate Δw in terms of its past derivatives, these coefficients do not depend upon T/H . They are the same as the corresponding rows of the Pascal Triangle Matrix.

$$(5.18) \quad A = \begin{bmatrix} 1 & 1 & a_{13}(r) & a_{14}(r) & \cdots \\ & 1 & 2 & 3 & \\ & & 1 & 3 & \\ & & & 1 & \ddots \\ & 0 & & & \ddots \end{bmatrix}.$$

The coefficients $a_{1,i}(r)$ are given in Table 5.2. If $T/H = 0$, then A is the Pascal triangle matrix.

TABLE 5.2.
Perturbed coefficients for Nordsieck prediction.

i	$\alpha_{1,i}(r)$
3	$1 - r$
4	$1 - \frac{3}{2}r + \frac{1}{2}r^2$
5	$1 - 2r + r^2$
6	$1 - \frac{1}{6}(15r - 10r^2 + r^4)$
7	$1 - 3r + \frac{5}{2}r^2 - \frac{1}{2}r^4$
8	$1 - \frac{7}{2}r + \frac{7}{2}r^2 - \frac{7}{6}r^4 + \frac{1}{6}r^6$
9	$1 - 4r + \frac{14}{3}r^2 - \frac{7}{3}r^4 + \frac{2}{3}r^6$
10	$1 - \frac{1}{10}(45r - 60r^2 + 42r^4 - 20r^6 + 3r^8)$
11	$1 - 5r + \frac{15}{2}r^2 - 7r^4 + 5r^6 - \frac{3}{2}r^8$

The corrector iteration is rewritten in terms of the vector \mathbf{a}_n as

$$(5.19) \quad \mathbf{a}_{n,(m+1)} = \mathbf{a}_{n,(m)} + \mathbf{l}[Hg_{n,(m)} - Hg(z_{n,(m)})],$$

where $Hg_{n,(m)}$ is the second component of $\mathbf{a}_{n,(m)}$, $z_{n,(m)}$ is the first component of $\mathbf{a}_{n,(m)}$ and

$$(5.20) \quad \mathbf{l} = \mathbf{S}\mathbf{c}.$$

For example, when $k = 2$ we have

$$\mathbf{l} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \frac{1}{2} & -\frac{1}{2} \end{bmatrix} \begin{bmatrix} \beta_0^* \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} \beta_0^* \\ 1 \\ \frac{1}{2} \end{bmatrix}.$$

The first column of \mathbf{S} is always $[1, 0, \dots, 0]^T$, and \mathbf{c} has the form $[\beta_0^*(r), 1, \dots, 0]^T$. Thus the components of \mathbf{l} will be the same as \mathbf{l} for conventional Adams methods in Nordsieck form, except β_0^* now depends upon r . (This reduces to Nordsieck's form of Adams method as $r \rightarrow 0$.)

With this form, the stepsize is changed easily by interpolation. Changing stepsize by a ratio $R_n = H_n/H_{n-1}$ corresponds to premultiplying \mathbf{a}_{n-1} by the matrix $C(R_n)$, where $C(R_n) = \text{diag}[1, R_n, R_n^2, \dots, R_n^k]$. Thus the predictor-corrector technique that we use is given by

$$(5.21) \quad \begin{aligned} \mathbf{a}_{n,(0)} &= AC(R_n)\mathbf{a}_{n-1}, \\ \mathbf{a}_{n,(m+1)} &= \mathbf{a}_{n,(m)} + \mathbf{l}H[g_{n,(m)} - g(z_{n,(m)})]. \end{aligned}$$

A code using the Nordsieck formulation of the generalized Adams methods was written (see Petzold [16] for a more detailed description and FORTRAN code). We will describe a few of the more important features here.

Using the Nordsieck representation to store past data as described above, it is only a matter of changing a few lines of DIFSUB (see Gear [5]) to allow it to solve difference equations of the type that are treated here.

Errors are estimated and the stepsize and order are changed in exactly the same way as in DIFSUB. The single-step truncation error is estimated to be a multiple of the difference between the predictor and the corrector, and is controlled to be less than EPS. For a k th order corrector the first term of the local truncation error is (this is the

same as $Q_{k+1}\Delta^{k+1}z$ from (4.14))

$$(5.22) \quad C_{k+1}^* \left(\frac{H}{T}\right)^{k+1} \Delta^{k+1}z.$$

The order is chosen (from the current order, one higher, and one lower) which gives the largest H when the local truncation error is set equal to EPS. However, here the error coefficient C_{k+1}^* depends on T/H (in conventional Adams methods, C_{k+1}^* is a constant). Setting (5.22) equal to EPS and solving for H thus amounts to solving a nonlinear equation. Since the error coefficients are very nearly the same as the (conventional) Adams-Moulton coefficients for T/H small, and T/H will usually be small for the type of problems that would be solved with this routine, we avoid the trouble of solving the nonlinear equations by assuming the coefficients are the same as for the corresponding (conventional) Adams method. This is a poor approximation only when H is near T (generally the first few steps at low order). The error coefficients for conventional Adams formulae are always larger than the coefficients for the generalized Adams formulae of the same order. The error coefficients C_{k+1}^* for the generalized Adams-Moulton formulae are listed in Table 5.3, where it is easy to verify that they tend to the error coefficients of conventional Adams formulae when T/H tends to 0, and that they tend to 0 when H tends to T (except for the generalized backward Euler method).

TABLE 5.3.
Error constants for generalized Adams-Moulton formulae.

k (Order)	$k!C_{k+1}^*$
1	$-\frac{1}{2} + \frac{1}{2}r$
2	$-\frac{1}{6} + \frac{1}{6}r^2$
3	$-\frac{1}{4} + \frac{1}{4}r^2$
4	$-\frac{19}{30} + \frac{2}{3}r^2 - \frac{1}{30}r^4$
5	$-\frac{9}{4} + \frac{5}{2}r^2 - \frac{1}{4}r^4$
6	$-\frac{863}{84} + 12r^2 - \frac{7}{4}r^4 + \frac{1}{42}r^6$
7	$-\frac{1375}{24} + 70r^2 - \frac{105}{8}r^4 + \frac{5}{12}r^6$
8	$-\frac{33953}{90} + 480r^2 - \frac{1624}{15}r^4 + \frac{50}{9}r^6$
9	$-\frac{57281}{20} + 3780r^2 - \frac{9849}{10}r^4 + 70r^6 - \frac{21}{20}r^8$
10	$-\frac{3250433}{132} + 33600r^2 - \frac{29531}{3}r^4 + \frac{5345}{6}r^6 - \frac{91}{4}r^8 + \frac{5}{66}r^{10}$
11	$-\frac{1891755}{8} + 332640r^2 - \frac{214995}{2}r^4 + \frac{47025}{4}r^6 - \frac{3465}{8}r^8 + 15r^{10}$

Note: C_{k+1}^* reduces to C_{k+1} ([5, p. 156]) when $r = 0$.

In finding the period, the set P (see § 3) was chosen to be empty in this implementation.

Almost any “black box” can be used to do the small-step integrations over one period, although it is good to use a code that can interpolate between steps efficiently, because of the quadrature involved in finding the period. DIFSUB was used here to perform the small-step integrations over one period.

The routines the user must supply are exactly the ones needed by DIFSUB. Also, the user must supply a good (accurate to within about 5 or 10 percent) starting guess TN for the period of the oscillation to the driving program. This may be deduced from the nature of the physical problem, from the largest eigenvalue, if the problem is nearly linear, or, if necessary, by just integrating with DIFSUB for a short time and observing where the solution begins to repeat itself.

6. Computational results. This section describes results of applying the code described in § 5 to several test problems. All computations were done on an

IBM 360/75 in double precision. The first example is

$$(6.1) \quad \begin{aligned} y'' + \lambda^2 y &= a \sin(\lambda t), \\ y(0) &= 1, \\ y'(0) &= -a/2\lambda. \end{aligned}$$

Converting to a system of first order equations makes this

$$(6.2) \quad \begin{aligned} \begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix}' &= \begin{bmatrix} 0 & \lambda \\ -\lambda & 0 \end{bmatrix} \begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ a/\lambda \sin(\lambda t) \end{bmatrix}, \\ \begin{bmatrix} y_1(0) \\ y_2(0) \end{bmatrix} &= \begin{bmatrix} 1 \\ -a/2\lambda^2 \end{bmatrix}. \end{aligned}$$

This has the solution

$$(6.3) \quad \begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} = \begin{bmatrix} (1 - (a/2\lambda)t) \cos(\lambda t) \\ -(1 - (a/2\lambda)t) \sin(\lambda t) - a/2\lambda^2 \cos(\lambda t) \end{bmatrix}.$$

The solutions obtained by the program will be compared to (6.3) at intervals of time which were chosen by the program. (These intervals were, of course, chosen for purposes of efficiency. However, it is easy to obtain approximations to $z(t)$ wherever you want them, by using the predictor formula and the vectors of stored values at these points to interpolate to other points.)

This problem was solved with the parameters $\lambda = 1000$, $a = 100$. Because of the high frequency forcing function $a \sin(\lambda, t)$, it is important to retain the phase information in this problem, so the stepsize H of the outer method was constrained to be a multiple of the period.

The parameters for the routine are listed in Table 6.1. The parameters have the same meanings as parameters of DIFSUB. (Actually, the parameters for the inner integrations are the parameters of DIFSUB.) For example, $EPS = .1D-3$ for the outer integration means that the routine will try to keep the single step local truncation error of the generalized methods alone to be less than $.1D-3$.

TABLE 6.1.
Problem 1—parameters.

Problem parameters	
λ	= 1000
a	= 100
Parameters for inner integration	
H	= .1D-5
EPS	= .1D-6
$HMIN$	= .1D-13
$HMAX$	= 1.D0
MF	= 0
$MAXDER$	= 10
Parameters for outer integration	
H	= .2512D-1
EPS	= .1D-3
$HMIN$	= .314D-1
$HMAX$	= 5.D0
MF	= 0
$MAXDER$	= 10
Parameters for period-finder	
TN	= .628D-2 = initial estimate of the length of a period

The problem was solved with DIFSUB alone (with parameters identical to the parameters for the inner integration) for comparison. Results are shown in Table 6.2 and the errors of the two solutions are compared, along with the number of function evaluations (of the original function $f(y, t)$ from the equation $y' = f(y, t)$), in Table 6.3. The generalized methods were much faster than DIFSUB alone, both in terms of function evaluations and execution time.

TABLE 6.2.
Problem 1—results.

Time	True solution	Solution by gen. methods	Solution by DIFSUB
.2513274D-1	.9987434	.9987451	.9987458
	-.5D-4	-.5032188D-4	-.4766485D-4
.5026548D-1	.9974867	.9974902	.9974922
	-.5D-4	-.5063552D-4	-.4513941D-4
2.664071	.8667964	.8669955	.8670935
	-.5D-4	-.6509385D-4	-.2940519D-3
5.277876	.7361062	.7363785	.7366541
	-.5D-4	-.1620934D-3	-.6477888D-4
8.846726	.5576634	.5581064	.5584882
	-.5D-4	-.2100309D-3	-.3004953D-3
12.41558	.3792145	.3798945	.3802331
	-.5D-4	-.1672133D-3	-.1978808D-2
13.29522	.3352390	.3359656	.3362903
	-.5D-4	-.1543100D-3	.2296590D-3
14.17487	.2912542	.2920207	*
	-.5D-4	-.1529315D-3	
15.05451	.2472740	.2480030	*
	-.5D-4	-.2412899D-3	
Period = .006283186			

* DIFSUB was stopped here after executing for 5 minutes, 45 seconds. Execution time for the generalized Adams program was approximately $\frac{1}{2}$ minute.

TABLE 6.3.
Problem 1—errors and efficiency.

Time	Error (gen. methods)	Error (DIFSUB)	Function evaluations (gen. methods)	Function evaluations (DIFSUB)
.2513274D-1	-.1737D-5	-.2400D-5	568	429
	.3218D-6	-.2335D-5		
.5026548D-1	-.3474D-5	-.5500D-5	809	810
	.6355D-6	-.4860D-5		
2.664071	-.1991D-3	-.2971D-3	1291	40,635
	.1509D-4	.2440D-3		
5.277876	-.2723D-3	-.5479D-3	1990	80,457
	.1120D-3	.1477D-4		
8.846726	-.4430D-3	-.8248D-3	2678	134,829
	.1600D-3	.2504D-3		
12.41558	-.6799D-3	-.1018D-2	3335	182,645
	.1172D-3	.1928D-2		
13.29522	-.7266D-3	-.1051D-2	4213	192,881
	.1043D-3	-.2796D-3		
14.17487	-.7664D-3	—	4633	—
	.1029D-3			
15.05451	-.7289D-3	—	5251	—
	.1912D-3			

Problem 2 describes the nonlinear oscillations of a slightly damped pendulum. The angle x of deviation from the origin for a pendulum of mass 1, length l , and damping coefficient μ is described by

$$(6.4) \quad x'' + \mu x' + \left(\frac{g}{l}\right) \sin(x) = 0.$$

Starting from an angle of 1 radian and releasing the pendulum, the initial conditions are

$$x(0) = 1, \quad x'(0) = 0.$$

This initial angle is large enough so that it is not possible to obtain an accurate solution to (6.4) from the linearized equation.

This problem differs from Problem 1 in several significant ways. First, it is an autonomous problem, and thus we expect not to see the synchronization difficulties (where H must be constrained to be a multiple of T) which were present in Problem 1 because of the high frequency forcing function. Secondly, the period of the oscillation is changing slowly. Because of this, it is more difficult for us to retain the phase information of the oscillation, although the amplitude information is quite easily obtained. Values of the parameters for the problem and the program are given in Table 6.4. For the program, Eq. (6.4) was split into a system of first order equations

$$(6.5) \quad \begin{aligned} x'_1 &= \frac{g}{l} x_2, & (x_1 = x), \\ x'_2 &= -\mu x_2 - \frac{g}{l} \sin(x_1). \end{aligned}$$

A change of variables was made so that the units of time are thousands of seconds.

TABLE 6.4.
Problem 2—parameters.

Problem parameters
$g = 9.8 \times 10^6 \text{ m}/(1000 \text{ sec})^2$
$l = 2 \text{ m}$
$\mu = .1/(1000 \text{ sec})$
Parameters for inner integration
H = .1D-5
EPS = .1D-6
HMIN = .1D-13
HMAX = 1.D0
MF = 1
MAXDER = 6
Parameters for outer integration
H = .1204D-1
EPS = .1D-2
HMIN = .1505D-1
HMAX = 5.D0
MF = 0
MAXDER = 10
Parameters for period-finder
TN = .301D-2

The numerical solution obtained by the generalized methods is compared to the amplitude of the first asymptotic approximation of Bogoliubov and Mitropolsky [3, pp. 141–143], since an exact analytic solution is not available. The first approximation is

given by

$$x(t) = a(t) \cos(\psi(t)),$$

where a and ψ are given by

$$(6.6) \quad \begin{aligned} a(t) &= e^{-(1/20)t}, \\ \psi(t) &= \frac{g}{l} \left[t + \frac{(e^{-.1t} - 1)}{.0016} \right]. \end{aligned}$$

The smooth solution $z(t)$ computed by the generalized methods is compared to $a(t)$, the amplitude of the first approximation. Results are shown in Table 6.5. The solutions obtained by the generalized methods for this problem were obtained with the program with H constrained to be a multiple of T . For this problem, in contrast to Problem 1, skipping over nonintegral numbers of periods does not impair the accuracy or efficiency of the program. In fact, the program is slightly faster in that case.

TABLE 6.5.
Problem 2—results.

Time	$z(t)$ = solution by generalized methods	Period	$a(t)$ = first asymptotic approximation to amplitude	$a(t) - z(t)$
.1210568D-1	.9993580	.3026420D-2	.9993949	.3689918D-4
.2421034D-1	.9987188	.3026164D-2	.9987902	.7141545D-4
.3804384	.9802374	.3018882D-2	.9811579	.9204561D-3
.7358455	.9622003	.3011924D-2	.9638763	.1676038D-2
1.060807	.9458811	.3005764D-2	.9483417	.2460649D-2
1.385118	.9299027	.2999881D-2	.9330879	.3185176D-2
1.708803	.9142388	.2994190D-2	.9181081	.3869295D-2
3.574200	.8292937	.2965452D-2	.8363484	.7054714D-2
4.036335	.8096557	.2959278D-2	.8172447	.7588984D-2
4.497534	.7905759	.2953405D-2	.7986147	.8038793D-2
4.957845	.7719919	.2947911D-2	.7804440	.8452143D-2
7.283115	.6847854	.2923921D-2	.6947830	.9997589D-2
9.591594	.6083878	.2905526D-2	.6190435	.1065574D-1
11.88713	.5410051	.2891251D-2	.5519176	.1091253D-1
15.72542	.4452374	.2874016D-2	.4555404	.1030296D-1
19.54533	.3670435	.2862533D-2	.3763384	.9294937D-2
23.35267	.3027923	.2854800D-2	.3111023	.8310016D-2

The numerical solution for this problem (with these parameters) is bounded above by the first approximation and below by the second approximation at each step, though we cannot expect that to happen in general. It should also be noted that the differences between the numerical solution and the first and second approximations are much smaller if a smaller initial angle (amplitude) is used (say, $\frac{1}{2}$ radian instead of 1 radian). This is probably due to the fact that the asymptotic approximations are much more accurate for small amplitudes, as they approximate $\sin(x)$ by the first two (or three) terms of its Taylor series. The amplitude information obtained by the generalized methods is very near to the amplitude of the solution obtained by DIFSUB alone. These observations would seem to indicate that the amplitude information in the numerical solution is quite accurate.

The solution by the generalized methods is compared to the solution by DIFSUB in Table 6.6. Because the phase information has been lost by the program (in the sense that the solution by DIFSUB at times which DIFF takes as integral multiples of the period is not at the top of a cycle), it is not possible to compare the numerical solution to the solution obtained by DIFSUB directly. Instead, we compare the energy of the two solutions to see how much they differ. (More precisely, a multiple of the energy is compared.) The total energy (potential plus kinetic) of the pendulum is given by

$$E(\mathbf{x}) = mgl[-\cos(x_1) + \frac{1}{2}x_2^2],$$

which is proportional to

$$(6.7) \quad E^*(\mathbf{x}) = [-\cos(x_1) + \frac{1}{2}x_2^2].$$

Thus, Table 6.6 compares E^* of the solution by the generalized methods to E^* of the solution obtained by DIFSUB. Again, it is apparent that the generalized methods are much more efficient than DIFSUB alone.

TABLE 6.6.
Problem 2—comparison with DIFSUB.

Time	E^* (solution by generalized methods)	E^* (solution by DIFSUB)	Function evaluations (generalized methods)	Function evaluations (DIFSUB)
.1210548D-1	-.5408417	-.5408409	794	592
.2421034D-1	-.5413800	-.5413782	1131	1,099
.3804384	-.5568254	-.5569341	2032	15,996
.7358455	-.5717161	-.5719238	2891	30,608
1.060807	-.5850285	-.5852117	4767	44,148
1.385118	-.5979120	-.5980755	5224	58,061
1.708803	-.6103976	-.6105285	5680	71,953
3.574200	-.6753966	-.6753252	6944	151,458
4.036335	-.6897475	-.6896766	8675	171,280*

* DIFSUB was stopped here after executing for 7 minutes, 46 seconds. The generalized Adams program completed the solution (to $t = 20$) in less than 1 minute.

If it is absolutely necessary to find the phase information, too, this can be accomplished, though with some sacrifice in efficiency. Until now, stepsizes in the outer integration routine were selected to control the l_2 -norm of the single-step local errors (absolute error is controlled if the solution is less than one in magnitude, and relative error otherwise). In Problem 2 the component of the solution which computes time (because the period is changing) was not influencing the choice of stepsize and order significantly. However, the solution (including the phase information) is much more sensitive to time than to the other components of \mathbf{z} (by a factor of the frequency). Thus if we choose the stepsizes to control a weighted l_2 -norm, where the last equation (determining time) is weighted by a multiple of the frequency, we would expect the phase information to be fairly accurate. This does happen, as indicated by the results in Table 6.7, where the last equation was weighted by $2/TN$ (TN is the initial estimate of the period). Since the phase information depends on that of the underlying small-step method alone, results are compared to the solution obtained by DIFSUB. Here the phase information is deteriorating slowly. Some efficiency has been sacrificed (it takes about twice as long to obtain the phase information as it did to obtain only the amplitude).

TABLE 6.7.
Problem 2—retaining phase information.

Time	First component of solution by generalized methods	First component of solution by DIFSUB	Function evaluations (generalized Adams methods)	Function evaluations (DIFSUB)
.1210548D-1	.9993588	.9993579	1127	592
.2421034D-1	.9987188	.9987160	1801	1,099
.1693964	.9910700	.9910877	2924	7,178
.3144379	.9834891	.9835288	3708	13,253
.4593375	.9759835	.9760352	4490	19,252
1.073636	.9448545	.9446012	5722	44,687
1.685614	.9149374	.9133819	6968	70,961
1.835275	.9077903	.9057128	8745	77,372
1.984810	.9007157	.8980791	9079	83,758
2.134221	.8937066	.8904342	9813	90,115
2.283510	.8877701	.8827193	10,547	96,437
2.754491	.8652782	.8587107	11,646	116,375
3.224348	.8443805	.8349327	12,717	136,373

From the problems described here it is apparent that with a proper choice of parameters the generalized methods can achieve large gains in efficiency over the underlying small-step method, at little expense in accuracy, for some oscillating problems. One class of oscillating problems that at present cannot be solved using these methods are stiff oscillating problems (see [16]). (These are problems whose solutions tend very rapidly to an oscillating solution.)

7. Summary—additional comments. A class of methods has been described which can efficiently solve highly oscillatory ordinary differential equations. These methods are based on the observation that if the solution is sampled at multiples of the period of the oscillation, then the resulting sequence of points can be used to define a slowly-varying function $z(t)$ which can be followed with large steps, and from which the solution to the original system can be recovered. Problems with slowly varying periods of oscillation can be solved with these methods by means of a change of variables.

The 'period' of a nearly periodic function has been defined here as the minimum over all possible values of the period, of a norm of the difference between the function and the function displaced by the period. An algorithm which is based on Newton's method has been introduced to find this period.

Formulae which are generalizations of Adams formulae have been derived. These methods can solve reasonably well-behaved nonstiff oscillating problems with an error of order H^k (in the sense described in § 4), taking steps that skip over several (possibly many) cycles of the oscillation. The error bounds for these methods depend upon the function $g(z, t)$, which is indirectly related to the differential equation.

A program was described which demonstrated the generality and efficiency of these methods for solving oscillating problems. Many decisions are involved in writing such a program, and in some cases it does not seem at all clear which choices are best. It seems evident that future efforts should lead to significantly faster codes, especially if the problems have a special form (for example, almost linear problems).

Acknowledgments. This work was performed as part of my doctoral dissertation research in the Department of Computer Science at the University of Illinois, Urbana-Champaign. I am most grateful to my advisor, Professor C. W. Gear, for his guidance and assistance, and especially for many illuminating and enjoyable discussions.

REFERENCES

- [1] V. AMDURSKY AND A. ZIV, *The Numerical Treatment of Linear Highly-oscillatory ODE Systems by Reduction to Nonoscillatory Type*, IBM Israel Scientific Center Report No. 39, March 1976.
- [2] L. AUSLANDER AND W. L. MIRANKER, *Algebraic Methods in the Study of Stiff Differential Equations*, IBM Report RC 7183 (# 30894), 1978.
- [3] N. N. BOGOLIUBOV AND Y. A. MITROPOLSKY, *Asymptotic Methods in the Theory of Nonlinear Oscillations*, Gordon and Breach Science Publishers, New York, 1961.
- [4] W. GAUTSCHI, *Numerical integration of ordinary differential equations based on trigonometric polynomials*, Numer. Math. 3 (1961), pp. 381–397.
- [5] C. W. GEAR, *Numerical Initial Value Problems in Ordinary Differential Equations*, Prentice-Hall, Englewood Cliffs, NJ, 1971.
- [6] O. F. GRAFF, *Methods of Orbit Computation with Multirevolution Steps*, Applied Mechanics Research Laboratory Report AMRL 1063, University of Texas, Austin, Texas, 1973.
- [7] O. F. GRAFF AND D. G. BETTIS, *Modified multirevolution integration methods for satellite orbit computation*, Celestial Mechanics 11 (1975), pp. 443–448.
- [8] P. HENRICI, *Discrete Variable Methods in Ordinary Differential Equations*, John Wiley, New York, 1963.
- [9] H. O. KREISS, *Problems With Different Time Scales For Ordinary Differential Equations*, Department of Computer Science Report No. 68, Uppsala University, Uppsala, Sturegaten 4, Sweden, October 1977.
- [10] J. D. LAMBERT AND I. A. WATSON, *Symmetric Multistep Methods for Periodic Initial Value Problems*, Department of Mathematics Report No. 12, University of Dundee, Dundee, Scotland, September 1975.
- [11] D. MACE AND L. H. THOMAS, *An extrapolation method for stepping the calculations of the orbit of an artificial satellite several revolutions ahead at a time*, Astronomical Journal, 65 (1960), # 1280.
- [12] N. MINORSKY, *Nonlinear Oscillations*, Robert E. Krieger Publishing Company, Huntington, NY, 1974, pp. 390–415.
- [13] W. L. MIRANKER AND F. HOPPENSTEADT, *Numerical Methods for Stiff Systems of Differential Equations Related with Transistors, Tunnel Diodes, etc.*, Lecture Notes in Computer Science 10, Springer-Verlag, New York, 1973.
- [14] W. L. MIRANKER AND M. VAN VELDHUIZEN, *The Method of Envelopes*, IBM Research Report RC 6391, #27537, February 1977.
- [15] W. L. MIRANKER AND G. WAHBA, *An averaging method for the stiff highly oscillatory problem*, Math. Comp., 30 (1976), pp. 383–399.
- [16] L. R. PETZOLD, *An Efficient Numerical Method for Highly Oscillatory Ordinary Differential Equations*, University of Illinois Dept. of Computer Science Report UIUCDCS-R-78-933, August 1978.
- [17] G. P. TARATYNOVA, *Numerical solution of equations of finite differences and their application to the calculation of orbits of artificial earth satellites*, AES Journal supplement, translated from Artificial Earth Satellites, 4 (1960), pp. 56–85.
- [18] C. E. VELEZ, *Numerical Integration of Orbits in Multirevolution Steps*, NASA Technical Note D-5915, Goddard Space Flight Center, Greenbelt, Maryland.