JOURNAL OF COMPUTATIONAL PHYSICS 100, 179–187 (1992)

Low-Order Polynomial Approximation of Propagators for the Time-Dependent Schrödinger Equation

HILLEL TAL-EZER

Department of Applied Mathematics, Tel Aviv University, Tel Aviv 69978, Israel

RONNIE KOSLOFF

Department of Physical Chemistry and the Fritz Haber Institute, Jerusalem 91904, Israel

AND

CHARLES CERJAN

Lawrence Livermore National Laboratory, Livermore, California 94550

Received January 10, 1990; revised May 8, 1991

Polynomial approximations for propagating the time-dependent Schrödinger equation are studied. These methods are motivated by the numerical demands of systems with time-dependent Hamiltonian operators. First-order and second-order Magnus expansions are tested for approximating the time ordering operator. The polynomials considered are based on a reduced basis space which is obtained by iterating the Hamiltonian operator on an initial wavefunction. The approximate polynomials are obtained by minimizing the error in the propagation. One such approach which minimizes the residuum outside the reduced space is proved to be equivalent to the short time iterative Lanczos procedure. A second approach which uses a different optimization scheme (the residuum method) is found to be somewhat superior to the Lanczos procedure. Numerical examples are used to illustrate the convergence of these methods. The second-order Magnus approximation is found to be a significant improvement over the first-order approximation regardless of method.

1. INTRODUCTION

Simulations of molecular and atomic phenomena based on a time-dependent approach lend substantial insight into these complex problems. These simulations are particularly useful when the problem has explicit time dependence—as in photodissociation or photoionization in a time varying field. Efficient means of constructing such simulations have a direct bearing on the scope of phenomena that can be studied. This study deals with the problem of approximating the time propagation operator. Much of the previous work in this vein was summarized in [1], which was mainly concerned with comparing the different approaches. This paper concentrates on the theory and application of low-order polynomial approximations for the evolution operator.

A time-dependent quantum mechanical approach to molecular dynamics [2] has many advantages over time-independent alternatives, such as the ease of physical interpretation, but for phenomena which have explicit time dependence the method becomes overwhelmingly superior. Explicit time dependence arises in many physical problems. A useful description of photochemical problems in strong fields involves a time-dependent driving potential or a time-dependent vector field [3],

$$\hat{\mathbf{H}}(x,t)=\frac{\hbar^2}{2M}\left(\hat{\mathbf{P}}-\frac{e}{c}\hat{\mathbf{A}}(t)\right)^2+\hat{\mathbf{V}}(x). \tag{1.1}$$

A typical situation in strong fields is that the energy stored in the vector field $(\hbar^2e^2/c^2)\,\hat{\mathbf{A}}^2(t)/2M$ is two orders of magnitude larger than the ionization energy. This requires a very small time step which is much larger than a typical driving frequency.

Another important class of problems arises from the use of the time-dependent self-consistent field (TDSCF) [4] approximation which transforms the original linear Schrödinger equation to coupled nonlinear differential equations. The TDSCF framework allows the possibility of mixing different types of solutions for different degrees of freedom. For example, for helium atom scattering from a crystalline surface the transitional degrees of freedom can be treated quantum mechanically and the surface motion classically [3]. All TDSCF approximations lead to an effective time-dependent Hamiltonian for each degree of freedom.

581/100/1-13

0021-9991/92 $5.00

Another class of problems involves systems which inherently have more than one characteristic time scale. This is common for molecules which have a large range of coupling frequencies or when electronic and vibrational degrees of freedom are considered simultaneously. For these systems one would want to avoid propagating with a time step corresponding to the fastest motion. A common feature of these problems with a time step corresponding to the fastest motionA common feature of these problems is that they require repetition of a relatively short propagation many times. Motivated by these problems a search for an accurate and efficient propagator was initiated.

This study is based on the time dependent Schrödinger equation

$$i\hbar \frac{\partial \psi(t)}{\partial t} = \hat{\mathbf{H}}\psi(t) \tag{1.2}$$

which is a member of a more general class of equations

$$\frac{\partial v}{\partial t} = \hat{\mathbf{A}}v, \tag{1.3}$$

where $\hat{\mathbf{A}}$ is a linear operator. Similar equations include the Liouville–von Neumann equation [6], the paraxial equation [7], and the heat or diffusion equation [8]. The Schrödinger equation has the formal solution

$$u(t) = \hat{\mathbf{U}}(t)\, u(0) = e^{-(i/\hbar)\hat{\mathbf{H}}t} u(0), \tag{1.4}$$

where $\hat{\mathbf{U}}$ is the evolution operator. If $\hat{\mathbf{H}}$ has explicit time dependence then

$$\hat{\mathbf{U}}(t) = \mathbf{T} e^{-(i/\hbar)\int_0^t \hat{\mathbf{H}}(t')\, dt'}, \tag{1.5}$$

where $\mathbf{T}$ is the time ordering operator. The lowest order solution to this problem is to split the evolution operator into a product of short time propagators for which the Hamiltonian operator $\hat{\mathbf{H}}(t)$ is almost constant,

$$\hat{\mathbf{U}}(t, 0) = \prod_n \hat{\mathbf{U}}(t_{n+1}, t_n), \tag{1.6}$$

where $t_n = n\,\Delta t$. For each step, the evolution operator is approximated by an exponential form

$$\hat{\mathbf{U}}(t_{n+1}, t_n) \approx e^{\hat{\mathbf{O}}_i}. \tag{1.7}$$

The operator $\hat{\mathbf{O}}_i$ can be the first Magnus approximation [9]

$$\hat{\mathbf{O}}_1 = -\frac{i}{\hbar}\int_{t_n}^{t_{n+1}} \hat{\mathbf{H}}(t')\, dt' \tag{1.8}$$

or the second Magnus approximation [9]

$$\hat{\mathbf{O}}_2 = \hat{\mathbf{O}}_1 - \frac{1}{2\hbar^2}\int_{t_n}^{t_{n+1}} dt' \int_{t_n}^{t'} [\hat{\mathbf{H}}(t'), \hat{\mathbf{H}}(t'')]\, dt''. \tag{1.9}$$

The approximation of the commutator inherent in Eqs (1.8) and (1.9) limits the maximum size of the time step Two distinct approximations are thus made in the application of the evolution operator: the numerical representation of the operators on a grid and a truncation of the commutator series in the formal expansion of the time-ordered operator.

These considerations lead to a common procedure in which the evolution operator becomes a product of exponentioned operators. In this paper the merits of a low order polynomial approximation to this exponentiation are considered,

$$e^{\hat{\mathbf{O}}_i} \approx \sum_{n=0}^{m} a_n(t)\, P_n(\hat{\mathbf{O}}_i). \tag{1.10}$$

The problem addressed is to find the optimum polynomials P_n and their expansion coefficients a_n.

2. BACKGROUND

Let $\hat{\mathbf{O}}$ be a Hermitian linear operator and $F(z)$ a function analytic in a domain D in the complex plane which includes all the eigenvalues of $\hat{\mathbf{O}}$. Let w be the vector which results from operating with $F(\hat{\mathbf{O}})$ on a vector v,

$$w = [F(\hat{\mathbf{O}})]\, v. \tag{2.1}$$

If $P_m(z)$ is a polynomial approximation of order m of $F(z)$, w can be approximated by w_m,

$$w_m = [P_m(\hat{\mathbf{O}})]\, v. \tag{2.2}$$

The error vector is

$$r_m = w - w_m = [F(\hat{\mathbf{O}}) - P_m(\hat{\mathbf{O}})]\, v. \tag{2.3}$$

Assume now that $\hat{\mathbf{O}}$ is finite-dimensional and has a complete set of eigenvectors $v_1, ..., v_N$ with corresponding eigenvalues, $\lambda_1, ..., \lambda_N$. Then v can be expanded in this set,

$$v = \sum_{k=1}^{N} a_k v_k, \tag{2.4}$$

where a_k are the expansion coefficients and the error vector becomes

$$r_m = \sum_{k=1}^{N} a_k [F(\lambda_k) - P_m(\lambda_k)]\, v_k. \tag{2.5}$$

Observing (2.5), it is obvious that by choosing $P_m(z)$ which interpolates $F(z)$ at the eigenvalue points λ_k, $k=1, ..., N$, the error will vanish. This is a manifestation of the Cayley–Hamilton theorem, since every N-dimensional linear operator satisfies its Nth order characteristic polynomial equation. This is a theoretical and impractical observation, since it requires the knowledge of all the eigenvalues, which is equivalent to solving the original problem. Moreover, if N is large (what we actually have in practice), it results in a very high degree interpolating polynomial which will lead to a time-consuming algorithm. A more practical approach is to settle for $P_m(z)$ which interpolates $F(z)$ in $m+1$ ($m \ll N$) points such that $\max_{z \in D} |F(z)-P_m(z)|$ is small. This approach was taken in [10]. It requires an a priori estimate of the domain D. (The algorithms given there require the vertices of a polygon in the complex plane which includes all the eigenvalues of $\hat{\mathbf{O}}$.) These techniques would be adequate for time-independent propagation schemes and have been demonstrated for the Chebyshev algorithm [1].

When this information is not at our disposal, we have to resort to other techniques. An algorithm which utilizes polynomial interpolation and is free of the requirement to know D is given in [11]. It is based on looking for $P_m(z)$ which interpolates $F(z)$ at $m+1$ points such that r_m is minimized in some sense. Since the minimization process is done on the error vector itself, we obtain a set of interpolating points which depend on the vector v on which we operate. Using this approach, one does not have to know D and, sometimes, the results are more accurate than the results produced by the previous method [10]. This phenomenon is explained by the following observation. In [10], the error function $E(z)=F(z)-P_m(z)$ has a uniform behavior in D. Thus, the method does not take into account a possible nonuniformity in the expansion coefficients a_k's (2.5). The new algorithm is constructed within the vector subspace generated by the action of a linear operator on an initial vector $v=\psi(0)$. This subspace, called the Krylov subspace of order $(m+1)$, is generated by the linear operator $\hat{\mathbf{O}}$,

$$u_j=\hat{\mathbf{O}}^j\psi(0), \tag{2.6}$$

where the u_j span a projected or reduced subspace of the original infinite-dimensional space. The new algorithm has a "focusing nature." It puts interpolating points in areas in the domain of eigenvalues D where they are "most needed". Obviously, the error function $E(z)$ is nonuniform. In Section 3a description of the algorithm is presented, and in Section 4 it will be shown that one variant of this algorithm is exactly the Lanczos method [1, 12].

3. RESIDUUM MINIMIZATION

The approach is to minimize the error or the residuum produced by the polynomial approximation. The differences in error estimation lead to two slightly different propagation methods.

Method I. Let $z_0, ..., z_m$ be $m+1$ distinct interpolating points; then the interpolating polynomial in Newton form is

$$P_m(z)=\sum_{k=0}^{m} a_k R_k(z), \tag{3.1}$$

where a_k are the divided differences [14]

$$a_k=F[z_0, ..., z_k] \tag{3.2}$$

and

$$R_0(z)=1 \tag{3.3}$$

$$R_k(z)=\prod_{i=0}^{k-1}(z-z_i), \qquad k=1, ..., m+1. \tag{3.4}$$

The error function $E(z)=F(z)-P_m(z)$ is given by

$$E(z)=a_{m+1}(z)\,R_{m+1}(z), \tag{3.5}$$

where

$$a_{m+1}(z)=F[z_0, ..., z_m, z]. \tag{3.6}$$

Using (3.5), the error vector r_m (2.3) can be written as

$$r_m=E(\hat{\mathbf{O}})v=a_{m+1}(\hat{\mathbf{O}})\,R_{m+1}(\hat{\mathbf{O}})v=a_{m+1}(\hat{\mathbf{O}})\,\tilde{r}_m, \tag{3.7}$$

where

$$\tilde{r}_m=R_{m+1}(\hat{\mathbf{O}})v. \tag{3.8}$$

Full minimization of the l_2-norm of r_m, $\|r_m\|$ can be very complicated due to the dependence of a_{m+1} on $\hat{\mathbf{O}}$. Thus the values z_k are chosen so that R_{m+1} yields a minimization of $\|\tilde{r}_m\|$. $R_{m+1}(z)$ can be written as

$$R_{m+1}(z)=\sum_{k=0}^{m+1}\gamma_{k+1}z^k, \tag{3.9}$$

where

$$\gamma_{m+2}=1. \tag{3.10}$$

Using (3.8) and (3.9) one obtains

$$\|\tilde{r}_m\|^2=\langle\tilde{r}_m|\tilde{r}_m\rangle=h(\gamma_1, ..., \gamma_{m+1}). \tag{3.11}$$

Thus, in order to minimize $\|\tilde{r}_m\|$ one has to solve the following set of $m+1$ linear equations

$$\frac{\partial h}{\partial \gamma_i}=0, \qquad i=1, ..., m+1. \tag{3.12}$$

The matrix representation of (3.12) is

$$H\gamma=\beta, \tag{3.13}$$

where H is a Hermitian matrix satisfying

$$H_{ij}=\langle \hat{\mathbf{O}}^i v | \hat{\mathbf{O}}^j v\rangle, \qquad 1\leqslant i,\ \ j\leqslant m+1, \tag{3.14}$$

and

$$\beta_i=-\langle \hat{\mathbf{O}}^{i-1} v | \hat{\mathbf{O}}^{m+1} v\rangle, \qquad 1\leqslant i\leqslant m+1. \tag{3.15}$$

(Notice that the matrix H_{ij} is the Hamiltonian matrix representation in the reduced basis set of the Krylov space). Once the solution to the linear system (3.13) is known one proceeds by computing the roots of $R_{m+1}(z)$ which are the desired interpolating points. Then the divided differences (3.2) are calculated and the interpolating polynomial $P_m(z)$ is obtained. The approximated vector $w_m=P_m(\hat{\mathbf{O}})v$ is computed by the following algorithm.

ALGORITHM 3.1.

$$r=v$$
$$w=a_0 r$$

For $i=1, ..., m$ do

$$r=(\hat{\mathbf{O}}-z_{i-1}\hat{\mathbf{I}})r$$
$$w=w+a_i r.$$

end do

$$w_m=w.$$

Method II. For the particular function $F(z)=1/z$ it can be verified that

$$a_{m+1}(z)=\frac{(-1)^{m+1}}{zz_0\cdots z_m}. \tag{3.16}$$

Thus

$$zE(z)=\left(1-\frac{z}{z_0}\right)\cdots\left(1-\frac{z}{z_m}\right)=Q_{m+1}(z), \tag{3.17}$$

where

$$Q_{m+1}(z)=\sum_{k=0}^{m+1}\delta_k z^k, \tag{3.18}$$

$$\delta_0=1. \tag{3.19}$$

Hence, for the function $F(\hat{\mathbf{O}})=\hat{\mathbf{O}}^{-1}$ a full minimization o the vector $\hat{\mathbf{O}}E(\hat{\mathbf{O}})v=\hat{\mathbf{O}}r_m$ is possible. In this case one has tc solve the set of equations

$$\hat{H}\delta=\hat{\beta}, \tag{3.20}$$

where $\hat{H}$ is a Hermitian matrix satisfying

$$\hat{H}_{ij}=\langle \hat{\mathbf{O}}^i v | \hat{\mathbf{O}}^j v\rangle, \qquad 1\leqslant i,\ \ j\leqslant m+1 \tag{3.21}$$

and

$$\hat{\beta}_i=-\langle \hat{\mathbf{O}}^i v | v\rangle \qquad 1\leqslant i\leqslant m+1. \tag{3.22}$$

In practice, as will be seen in Section 5, it was found that, using (3.20) instead of (3.13), gave equivalent or better results for approximating the evolution operator. Once the solution to the linear system (3.20) is known the algorithm proceeds as in Method I.

4. LANCZOS METHOD

In Ref. [1] as well as in Ref. [12], an algorithm for solving the Schrödinger equation based on Lanczos method [13] was described. The method can be presented as follows. Let S be the Krylov subspace

$$S=\{v, \hat{\mathbf{O}}v, ..., \hat{\mathbf{O}}^m v\}, \tag{4.1}$$

where $\hat{\mathbf{O}}_{m+1}$ is the $(m+1)\times(m+1)$ matrix representation of $\hat{\mathbf{P}}\hat{\mathbf{O}}$ and $\hat{\mathbf{P}}$ is an l_2-orthogonal projection operator on S. Then

$$(\hat{\mathbf{O}}_{m+1})_{ij}=\begin{cases}1 & i=j+1, \quad 1\leqslant j\leqslant m\\ -a_{i-1} & 1\leqslant i\leqslant m+1, \quad j=m+1\\ 0 & \text{otherwise,}\end{cases}$$

where

$$-(a_0 v+\cdots+a_m\hat{\mathbf{O}}^m v)=\hat{\mathbf{P}}(\hat{\mathbf{O}}^{m+1}v). \tag{4.2}$$

Observe that if one defines

$$a=[a_0, ..., a_m]^T \tag{4.3}$$

then $a=\gamma$, where γ is the solution of (3.13).

Now, if

$$w = F(\hat{\mathbf{O}})\, v, \tag{4.4}$$

define

$$w_S = F(\hat{\mathbf{O}}_{m+1})\, v_S, \tag{4.5}$$

where

$$v_S = [1, 0, ..., 0]^T, \tag{4.6}$$

so that

$$w_S = F(\hat{\mathbf{O}}_{m+1})_1. \tag{4.7}$$

The subscript S denotes a representation of a vector in the space S, defined in Eq. (4.1); the subscript 1 specifies the first column. The polynomial approximation of $F(\hat{\mathbf{O}})$ based on the Lanczos method is

$$P_m^{\mathrm{Lan}}(\hat{\mathbf{O}})\, v = \sum_{i=0}^{m} (w_S)_i\, \hat{\mathbf{O}}^i v. \tag{4.8}$$

The vector w_S is computed as follows. If T is the matrix which diagonalizes $\hat{\mathbf{O}}_{m+1}$,

$$T^{-1} \hat{\mathbf{O}}_{m+1}\, T = \Lambda, \tag{4.9}$$

then

$$F(\hat{\mathbf{O}}_{m+1}) = TF(\Lambda)\, T^{-1}. \tag{4.10}$$

Hence

$$w_S = [F(\hat{\mathbf{O}}_{m+1})]_1 = TF(\Lambda)(T^{-1})_1. \tag{4.11}$$

The next section will show the equivalence of Method I described in Section 2 and the Lanczos method. To this end the following two lemmas are needed.

LEMMA 4.1. *The characteristic polynomial of* $\hat{\mathbf{O}}_{m+1}$ *is*

$$C_m(\lambda) = \sum_{j=0}^{m+1} a_j \lambda^j \qquad (a_{m+1} = 1). \tag{4.12}$$

Proof. One has

$$|\lambda \hat{\mathbf{I}} - \hat{\mathbf{O}}_{m+1}| = \lambda\, |\lambda \hat{\mathbf{I}} - \tilde{\hat{\mathbf{O}}}_m| + (-1)^m a_0 (-1)^m, \tag{4.13}$$

where the last column of $\tilde{\hat{\mathbf{O}}}_m$ is $[-a_1, ..., -a_m]^T$. Using induction one obtains the desired result.

Since $a = \gamma$, the interpolating points used in Method I are the roots of the characteristic polynomial of $\hat{\mathbf{O}}_{m+1}$. Assume now that the roots are distinct and designated $z_0, ..., z_m$. The next lemma gives the elements of T^{-1}.

LEMMA 4.2. T^{-1} *is Vandermonde's matrix whose elements are*

$$T_{ij}^{-1} = z_{i-1}^{j-1}, \qquad 1 \leqslant i, \quad j \leqslant m+1 \tag{4.14}$$

(*the superscript above z indicates power*).

Proof. Define the matrix Q,

$$Q_{ij} = z_{i-1}^{j-1}. \tag{4.15}$$

One has

$$(Q \hat{\mathbf{O}}_{m+1})_{ij} = \sum_{n=1}^{m+1} Q_{in} (\hat{\mathbf{O}}_{m+1})_{nj} = \sum_{n=1}^{m+1} z_{i-1}^{n-1} (\hat{\mathbf{O}}_{m+1})_{nj}. \tag{4.16}$$

Hence

$$(Q \hat{\mathbf{O}}_{m+1})_{ij} = \begin{cases} z_{i-1}^{j}, & 1 \leqslant j \leqslant m \\ -\sum_{n=1}^{m+1} a_{n-1} z_{i-1}^{n-1}, & j = m+1. \end{cases}$$

Since the z_i are the roots of the characteristic polynomial one obtains

$$(Q \hat{\mathbf{O}}_{m+1})_{ij} = \begin{cases} z_{i-1}^{j}, & 1 \leqslant j \leqslant m \\ z_{i-1}^{m+1}, & j = m+1 \end{cases}$$

or

$$(Q \hat{\mathbf{O}}_{m+1})_{ij} = z_{i-1}^{j}, \qquad 1 \leqslant i, \quad j \leqslant m+1. \tag{4.17}$$

Since

$$\Lambda_{ii} = z_{i-1}, \qquad 1 \leqslant i \leqslant m+1, \tag{4.18}$$

then

$$Q \hat{\mathbf{O}}_{m+1} = \Lambda Q. \tag{4.19}$$

Thus

$$T^{-1} = Q \tag{4.20}$$

and the proof is concluded.

Based on these two lemmas the following theorem holds

THEOREM 4.1.

$$P_m^{\mathrm{Lan}}(\hat{\mathbf{O}}) = P_m(\hat{\mathbf{O}}), \tag{4.21}$$

where the roots $z_0, ..., z_m$ *are distinct,* $P_m(\hat{\mathbf{O}})$ *is the polynomial approximation of* $F(\hat{\mathbf{O}})$ *which results from using Method I, and* $P_m^{\mathrm{Lan}}(\hat{\mathbf{O}})$ *is the polynomial approximation based on the Lanczos method.*

Proof. Let

$$P_m(z) = \sum_{i=0}^{m} a_i z^i. \tag{4.22}$$

One has

$$a_0 + a_1 z_j + \cdots + a_m z_j^m = F(z_j), \qquad 0 \leqslant j \leqslant m. \tag{4.23}$$

Writing (4.23) in matrix form results in

$$T^{-1}a = F(\Lambda)e = F(\Lambda)(T^{-1})_1, \tag{4.24}$$

where

$$a = [a_0, ..., a_m]^T, \qquad e = [1, ..., 1]^T. \tag{4.25}$$

Hence

$$a = TF(\Lambda)\, T^{-1} = w_S \tag{4.26}$$

and the proof is concluded.

Examining the differences between the two algorithms one finds that in the Lanczos algorithm the procedure minimizes the projection of the operator outside the Krylov subspace. The Krylov subspace is then used in order to construct the approximation for the function $F(\hat{\mathbf{O}})$ regardless of the function. Method II includes some information about the form of the function, because of the specific choice made; nevertheless the approximation is not tailored for the exponential operator. Its utility must be verified by examining representative problems.

5. RESULTS

There are two sources of error in the time propagation: the second-order Magnus approximation and the error induced by the truncated polynomial representation. The accuracy in the second-order Magnus expansion, as compared to the first order, was first investigated.

In order to check an explicitly time-dependent problem a linearly driven harmonic oscillator was chosen. The Hamiltonian has the form

$$\hat{\mathbf{H}} = \frac{\hat{\mathbf{p}}^2}{2M} + \frac{1}{2}k\hat{\mathbf{q}}^2 + \hat{\mathbf{q}}\varepsilon \sin t. \tag{5.1}$$

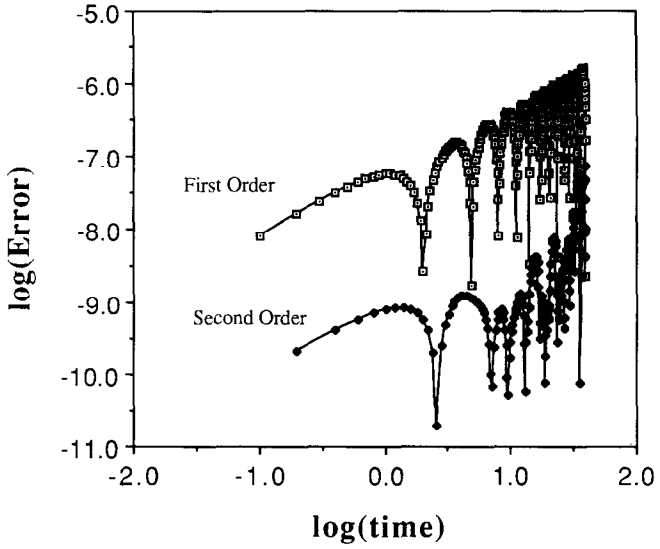

FIG. 1. A comparison of the first- and second-order Magnus accumulated logarithmic amplitude error for a fifth-order interpolating polynomial using the residuum procedure. The error is defined to be the absolute value of the difference between the exact and numerical result. In this figure, a driven harmonic oscillator was evaluated.

A resonant driving field was chosen by using $M = 1$ and $k = 1$. In this case an analytic solution for the dynamics can be obtained. The wavefunction-averaged amplitude $\langle \hat{\mathbf{q}} \rangle$ was chosen for comparison,

$$\langle \hat{\mathbf{q}}(t) \rangle = \tfrac{1}{2}\varepsilon(t \cos t - \sin t) \tag{5.2}$$

where $\langle \hat{\mathbf{q}} \rangle(0) = 0$ and the oscillator starts from the ground

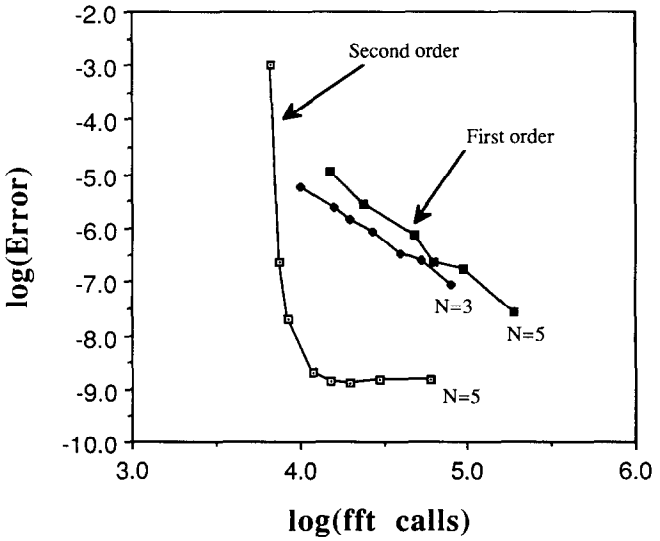

FIG. 2. The logarithm of the maximum error as a function of numerical effort which is given by the number of calls to the FFT routines. The errors were estimated by applying a linear fit to the maximum eror in each oscillator cycle. The first-order results have an approximate quadratic functionality.

state. A grid of 128 points was used with a grid spacing of 0.2 Bohr. The spectral range of the grid becomes 205.3 hartrees, which means that after 16 cycles with a power of $\varepsilon = 0.25$, the maximum possible average position represented on this grid—12.8 Bohr—is reached.

For a particular time-dependent problem one has to consider when to use the higher-order Magnus approximation. The numerical effort involved in the second Magnus approximation is twice that of the first-order approximation. (This numerical effort can be reduced to a factor of 1.5 by storing the first and second derivatives of the potential). The high-order expansion is worthwhile if the time step can at least be doubled. Figure 1 shows the error as a function of time for the first- and second-order Magnus approximation. The time steps were set such that the numerical effort is approximately the same for both orders. The error shows an oscillatory behavior. For the first-order results this error is superimposed on a linear relationcaused by the accumulation of errors. Examining the corresponding case in Fig. 2, which displays the logarithm of the error as a function of the logarithm of the numerical effort, a quadratic scaling relation is found for the error with respect to Δt. This means that the source of error is the Magnus series truncation, and that the fifth-order polynomial is already converged. The second-order Magnus runs show a different behavior. For about 13 cycles the error shows no dependence on time because of the saturation of the error at the level of 10^{-9}. The error is two orders of magnitude less than the first-order run for the same numerical effort. After about 13 cycles the error due to overflowing the spectral range of

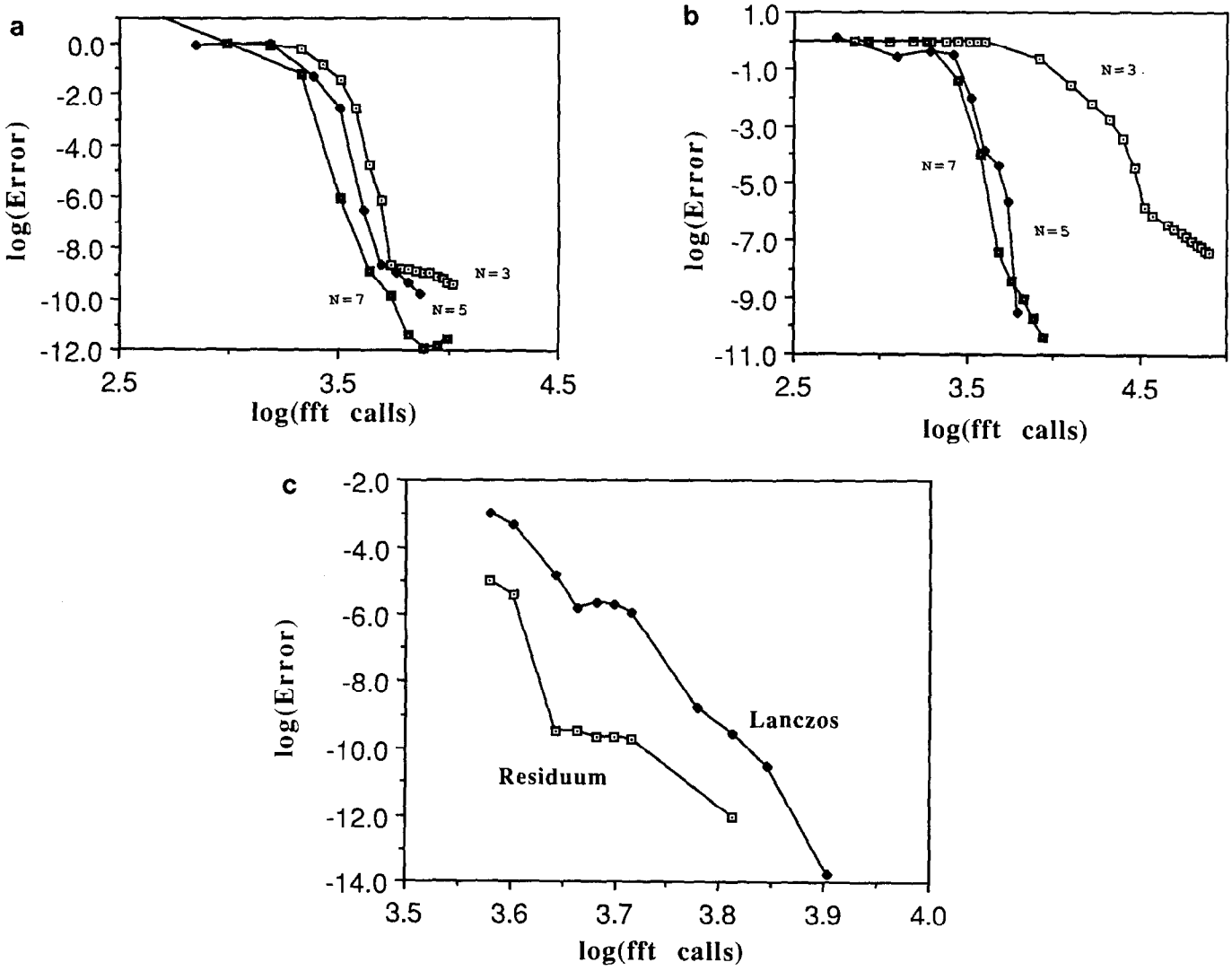

FIG. 3. (a) Convergence of the residuum algorithm as a function of polynomial order for the logarithm of the phase error for a bound state in a Morse oscillator-propagated 2400 atomic time units. The parameters are given in Ref. [1]. (b) Convergence of the Lanczos algorithm as a function of polynomial order for the logarithmic phase error as in Fig. 3a. (c) Comparison between the residuum and Lanczos algorithms for the logarithmic phase error as in Fig. 3a for polynomial order 7.

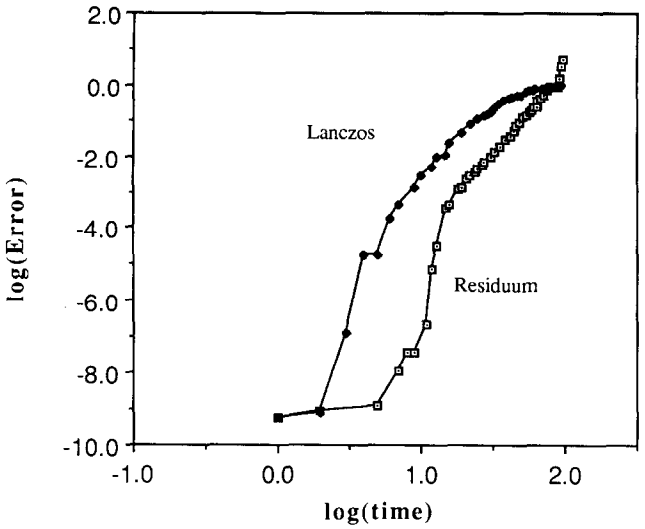

FIG. 4. The logarithmic position error for the second-order Magnus approximation to the forced oscillator problem are compared for the residuum and Lanczos methods as a function of time. The parameters for the calculation are the same as those used in Figs. 1 and 2.

the grid starts to dominate so the total error increases. Increasing the number of grid points which increases the spectral range will postpone this error. This behavior can be seen in Fig. 2: the results show very rapid convergence which is controlled by the polynomial approximation.

The second set of runs compared the two optimization methods. A Morse oscillator was used for this comparison with the same parameters as those used in Ref. [1]. Figures 3a and b show the convergence of the different polynomial approximations with respect to the numerical effort involved for the residuum and Lanczos algorithms. The low order ($N=3$) residuum polynomial is approximately 10 times more efficient for the same accuracy than the Lanczos algorithm. Similar results are found for orders 5 and 7 with an even greater disparity in the accuracy afforded for the same numerical effort. Figure 3c contains a comparison of the converged, order 7, results. The difference in accuracy is at least several orders of magnitude over the range calculated.

The driven oscillator example was used to compare the behavior of the residuum and Lanczos methods for an explicitly time-dependent problem. The second-order Magnus approximation was calculated in a fifth-order scheme for both methods. The results of this computation are plotted in Fig. 4.

6. DISCUSSION

Three sources of error have been mentioned: grid size, time propagation (exponentiation), and time ordering. Efficient calculations should balance all the errors. The Fourier representation is known to be a highly convergent method, thus an efficient time propagation scheme must complement its use. Otherwise the calculation is unbalanced in its error treatment and thus inefficient. It has been shown here and in Ref. [1], that a short optimized polynomial approximation has extremely rapid convergence. Therefore the combined use of the Fourier representation has extremely rapid convergence. Therefore the combined use of the Fourier representation and these algorithms leads to a balanced calculation. For explicitly time-dependent problems, it might happen that truncation of the time ordering error can dominate the error. This behavior is demonstrated in Section 5 where the first-order Magnus approximation is used. The first-order calculation shows quadratic convergence, whereas the second-order Magnus approximation resulted in a balanced, highly accurate numerical result.

The behavior of the residuum and Lanczos methods are compared for the driven oscillator problem in Fig. 4. In this figure, the second-order Magnus approximations was used to evaluate the time evolution as in Fig. 1. The error overwhelms the spatial grid representation by the time 10 atomic time units are reached. For this problem, the residuum method is significantly superior to the Lanczos method before the error saturates. Thus, the time-independent and time-dependent model problems reveal the same trends: the residuum algorithm is more efficient than the Lanczos algorithm. The underlaying reason for the greater efficiency of the residuum algorithm is not obvious; indeed, it might not always be the case, though it has been observed in other driven oscillator systems [16]. The generality of this conclusion is under investigation.

A further comment concerning the differences between the two methods should be made. The Lanczos algorithm is explicitly unitary—it conserves both normalization and average energy. The residuum method is not unitary. Though inherent stability is a desirable feature of any propagation algorithm, practically speaking it does not matter if the accuracy of the method is sufficiently great. Furthermore, departure from unitarity can be used as a measure of numerical error. Also, both methods can be used in a variable time stepping scheme which can offer significant advantages for problems with differing timescales. Extensions to higher dimensions are straightforward for either algorithm and have been applied to multi-dimensional systems [15].

ACKNOWLEDGMENTS

The authors thank Dr. A. Szoke (LLNL) for his comments. Work at the Fritz Haber Institute was supported by the Minerva Gesellschaft and the U.S.-Israel Binational Science Foundation. Work at the Lawrence Livermore National Laboratory was performed under the auspices of the U.S. Department of Energy, administered by the University of California under Contract W-7405-Eng-48.

REFERENCES

1. C. Leforestier, R. Bisseling, C. Cerjan, M. Feit, R. Friesner, A. Guldberg, A. Hammerich, G. Jolicard, W. Karrlein, H. D. Meyer, N. Lipkin, O. Roncero, and R. Kosloff, *J. Comput. Phys.*, to appear.
2. R. Kosloff, *J. Phys. Chem.* **92**, 2087 (1988).
3. C. Cerjan and R. Kosloff, *J. Phys. B* **20**, 4441 (1987).
4. R. H. Bisseling, R. Kosloff, R. B. Gerber, M. A. Ratner, L. Gibson, and C. Cerjan, *J. Chem. Phys.* **87**, 2760 (1987); A. Nitzan, Z. Kotler, and R. Kosloff, *Chem. Phys. Lett.* **153**, 483 (1988).
5. R. Kosloff and C. Cerjan, *J. Chem. Phys.* **93**, 7556 (1989).
6. M. Berman, H. Tal-Ezer, and R. Kosloff, unpublished, 1989.
7. M. D. Feit and J. M. Fleck, Jr., *Appl. Opt.* **17**, 3990 (1978).
8. N. Agmon and R. Kosloff, *J. Phys. Chem.* **91**, 1988 (1987).
9. R. D. Levine, *Quantum Mechanics of Molecular Rate Processes* (Oxford Univ. Press, Oxford, 1969).
10. H. Tal-Ezer, *Polynomial Approximation of Functions of Matrices and its Application to the Solution of a General System of Linear Equations*, ICASE Report No. 87–63, NASA Langley Research Center, Hampton, Virginia, 1987.
11. H. Tal-Ezer, Polynomial approximation of functions of matrices by residuum minimization, to appear.
12. T. J. Park and J. C. Light, *J. Chem. Phys.* **85**, 5870 (1986).
13. C. Lanczos, *J. Res. Nat. Bur. Stand.* **45**, 255 (1950).
14. For example, S. M. Pizer, *To Compute Numerically* (Little Brown, Boston, 1983).
15. R. Kosloff, S. A. Rice, P. Gaspard, S. Tersigni, and D. J. Tannor, *Chem. Phys.*, to appear.
16. R. Kosloff and C. Cerjan, unpublished results.