Algorithms for Obtaining Cumulative Reaction Probabilities for Chemical Reactions

Xudong T. Wu and Edward F. Hayes

Department of Chemistry, Ohio State University, Columbus, Ohio 43210

Received April 8, 1996; revised August 22, 1996

Miller and coworkers have proposed calculating the cumulative reaction probabilities of a chemical reaction through the computation of the few lowest eigenvalues of a matrix **Z**. In this study, the possibility of computing these eigenvalues using the iterative IRLM approach of Sorensen is investigated. The application to collinear $H + H_2 \rightarrow H + H_2$ reactions shows that it is possible to compute the physical eigenvalues of **Z** iteratively using the IRLM formulation with a Chebychev preconditioned **Z** matrix as the primary matrix. Because the IRLM formulation only requires simple matrix–vector operations, one never needs to assemble and store the whole **Z** matrix. Moreover, the multiplication of **Z** by a vector takes full advantage of the underlying sparseness of the matrix. This study presents several new and successful strategies to improve the convergence speed of IRLM, as well as strategies for obtaining the eigenvalues of **Z** when the eigenvalue span of **Z** is very large (1–10^{10}). © 1997 Academic Press

I. INTRODUCTION

In recent years, considerable progress has been made in developing methods for carrying out exact calculations of the state-to-state reactive cross sections for three-atom systems [1]. While there have been recent reports of exact numerical results for a few four-atom systems [2], there still is a need for more efficient methods that can be used to study reactions involving systems with four or more atoms.

Recently, Miller and coworkers [3] developed a new approach for calculating the cumulative reaction probabilities, $N(E)$ for a chemical reaction. The purpose of this paper is to report an extension to their approach that has the potential to improve significantly the computational performance of the method. This may be significant because the study of the dynamics of simple chemical reactions in the gas phase is a major research area of modern physical chemistry. In particular, the ability to carry out exact numerical reactive-scattering calculations is important for modeling combustion and atmospheric processes.

In the approach of Miller and coworkers [3] the Hamiltonian and the flux operators are computed in a discrete variable representation (DVR) and an accurate representation for the Green's operator, $G(E^+)$, is obtained by imposing absorbing boundary conditions (ABC). The resulting equation for $N(E)$ is

$$N(E) = 4\mathrm{Tr}(\varepsilon_r^{1/2} G \varepsilon_p G^* \varepsilon_r^{1/2}), \tag{1}$$

where ε_r is the absorbing potential in the reactant valley and ε_p is that in the product valley. G is the Green's function operator given by

$$G = (E + i(\varepsilon_r + \varepsilon_p) - H)^{-1}, \tag{2}$$

where H is the Hamiltonian of the reactive system and E is the energy of the reaction.

The impediments to treating four-atom systems without introducing serious approximations in the dynamical calculations are related to the number of active degrees of freedom that must be treated. A three-atom system has three internal dimensions, while a four-atom system has six internal dimensions. This increase in the internal dimensions causes a very significant increase in the dimensionality of the problem. For each internal dimension one must use a sufficient number of DVR points to achieve convergence in the scattering calculations. For instance for a typical three-atom system such as the reaction $Li + HF \rightarrow LiF + H$ the number of DVR basis functions needed to converge the cumulative reaction is about 20^3, whereas for a four-atom reaction such as $Cl + O_3 \rightarrow ClO + O_2$ the number of basis functions would be about 20^6. Miller and coworkers [2c] have determined cumulative reaction probabilities, $N(E)$, as a function of energy for the reaction: $H_2 + OH \rightarrow H + H_2O$. In that study they were able to use a reduced number of DVR points for the four less-coupled degrees of freedom. This resulted in a significantly reduced dimensionality for $N(E)$ (i.e., about 1×10^5 to 2×10^5 for the energy range studied). While such a reduction in grid points is possible for this system, there continues to be a strong interest in improving the underlying computational methods so that a broader range of systems with four and more atoms can be studied.

0021-9991/97 $25.00

The majority of the computational time required to solve Eq. (1) is devoted to obtaining the eigenvalues for the Hermitian operator **Z**,

$$\mathbf{Z} = \tfrac{1}{4}\,\varepsilon_r^{-1/2}(H + i(\varepsilon_r + \varepsilon_p) - E)\varepsilon_p^{-1} \\ (H - i(\varepsilon_r + \varepsilon_p) - E)\varepsilon_r^{-1/2} \tag{3}$$

in an appropriate DVR representation. Once the eigenvalues are known, the cumulative reaction probability, $N(E)$, can be obtained as

$$N(E) = \sum_k \frac{1}{\lambda_k}, \tag{4}$$

where $\{\lambda_k\}$ are the eigenvalues of **Z**.

In the next section, the characteristics of the matrix **Z** will be discussed in order to provide a bases for introducing our extensions to the computational approach. In Section III, we introduce the implicitly restarted Lanczos method (IRLM) of Sorensen and coworkers [4]. Section IV contains the extensions that we are proposing. Finally, in Section V, we present results for the collinear H_3 system.

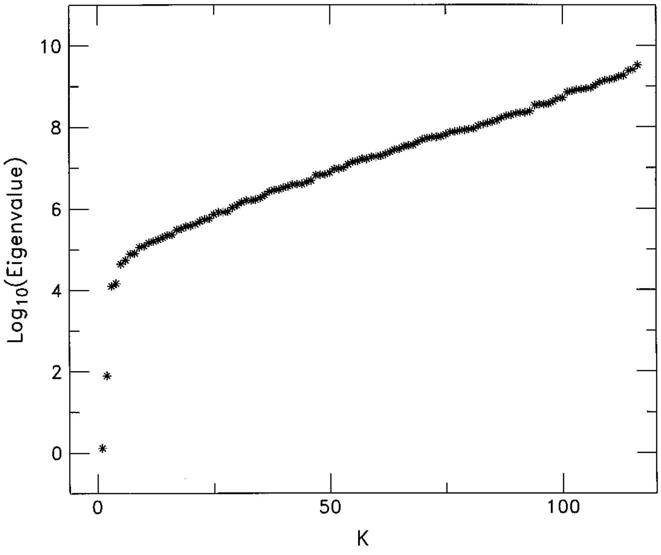

FIG. 1. λ_k vs. k for **Z** matrix, where the eigenvalues are arranged in ascending order. A logarithmic scale is used for the λ_k axis. **Z** is constructed for collinear $H + H_2 \rightarrow H_2 + H$ reaction for a scattering energy of 0.86 eV.

II. THE PROPERTIES OF Z

By definition, the absorbing potential, ε_r, in Eq. (3) becomes vanishingly small at coordinates outside the reactant absorbing region (this is also the case for ε_p outside the product absorbing region). This will cause numerical problems in determining **Z** because of the factors $\varepsilon_r^{-1/2}$ and ε_p^{-1} in this equation. To avoid this problem, we utilize the modification proposed by Manthe and Miller [3d] and add a constant "floor" ε_0 to the absorbing potentials, i.e.,

$$\varepsilon_r \rightarrow \varepsilon_r + \varepsilon_0 \tag{5}$$

$$\varepsilon_p \rightarrow \varepsilon_p + \varepsilon_0. \tag{6}$$

In the final stages of the calculation, it is necessary to check whether ε_0 is small enough to introduce a negligible error in the eigenvalues of **Z**.

Since **Z** is a Hermitian matrix, all of its eigenvalues are real-valued. Manthe and Miller [3d] further proved that all eigenvalues of **Z** are greater than or equal to 1. In this paper, we focus on the collinear $H + H_2 \rightarrow H_2 + H$ reaction system as a simple illustration of the various aspects of our proposed approach to obtaining the eigenvalues, λ_k, of **Z**. Figure 1 contains a plot of λ_k versus k for this reaction for a scattering energy, E, equal to 0.86 eV. The eigenvalues are on a logarithmic scale. From this graph, one can see that for this particular scattering energy there are just two eigenvalues less than 10, while all the other eigenvalues are larger than 10^4. At this particular energy, only two eigenvalues will contribute significantly to the sum for $N(E)$, Eq. (4). The contributions from the other eigenvalues can be ignored when calculating $N(E)$.

Manthe and Miller [3d] have also discussed the relationship between the ABC–DVR formulation and transition state theory. The basic conclusion from this discussion is that eigenvalues, which contribute significantly to $N(E)$, correspond to transition states. Moreover, the very large eigenvalues that make very small contributions are an artifact of the absorbing potentials. Because of the connection between the contributing eigenvalues and the transition states, one expects the following: the higher the total energy, the larger the number of eigenvalues that may contribute significantly to $N(E)$; the higher the dimensionality of the reaction system, the larger the number of eigenvalues that contribute significantly to $N(E)$; and the number of contributing eigenvalues is always much less than the number of DVR basis functions.

Because of the connection between transition states and the contributing eigenvalues, these eigenvalues and their corresponding eigenvectors will be referred to as the *physical* eigenvalues and the *physical* eigenvectors. Since the large eigenvalues of **Z** are an artifact of the absorbing potentials, they and their corresponding eigenvectors are referred to as *nonphysical* eigenvalues and the *nonphysical* eigenvectors.

From Fig. 1, it is apparent that all the nonphysical eigenvalues lie on a straight line in the semilog plot. This means

$$\lambda_k \approx C_1 e^{C_2 k} \tag{7}$$

for all the nonphysical eigenvalues. Therefore, the distance between the adjacent nonphysical eigenvalues is approximately

$$\lambda_{k+1} - \lambda_k \approx C_1 e^{C_2 k}(e^{C_2} - 1). \tag{8}$$

This means that $(\lambda_{k+1} - \lambda_k)$ increases exponentially as k increases for the nonphysical eigenvalues. Thus for the nonphysical eigenvalues, the eigenvalue distribution becomes more and more sparse as the eigenvalues increase.

A characteristics of the eigenvalue spectrum of **Z** that is of interest here is the extremely large span of the eigenvalues—with a few eigenvalues clustered at the lower end of the spectrum and many very large eigenvalues. In Fig. 1, the largest eigenvalue equals 3.34×10^9, while the smallest eigenvalue equals 1.30, which is 2.6×10^9 times smaller than the largest eigenvalue.

A notable feature of Fig. 1 is that there is a large gap between the lowest nonphysical eigenvalue and the highest physical eigenvalue. Because Fig. 1 is a semilog plot, this gap means the ratio $\lambda_{\text{nonphy}}/\lambda_{\text{phy}}$ is very large. For the scattering energy indicated, 0.86 eV, this ratio is over 150.

Another aspect of this approach is that one does not need to determine $N(E)$ to more than two or at most three significant figures in order to generate predictions that will have better error limits than the corresponding experimental results.

III. COMPARISON OF THE LANCZOS METHOD AND IRLM

To determine $N(E)$ one needs to calculate the smallest eigenvalues of the matrix **Z**, Eq. (3). If the order of the matrix **Z**, is small enough one can use standard dense matrix eigenvalue routines to diagonalize **Z**. However, if the order of the matrix is large, then iterative methods may be more efficient for obtaining the small number of needed eigenvalues.

The Lanczos method [5] is one of the most popular iterative methods for obtaining eigenvalues of a large Hermitian matrix. The core of the Lanczos method is the Lanczos factorization. Starting with an initial vector v_1, the k step Lanczos factorization produces a $k \times k$ upper Hessenberg matrix **H**, which satisfies

$$\mathbf{ZU} = \mathbf{UH} + re_k^T \tag{9}$$

where $\mathbf{U} \in \mathbf{C}^{n\times k}$ satisfies

$$\mathbf{U}^\dagger\mathbf{U} = \mathbf{I}_k \tag{10}$$

and $r \in \mathbf{C}^n$ satisfies

$$\mathbf{U}^\dagger r = 0, \tag{11}$$

where the first column of **U** equals v_1 and e_k^T is the transpose of the kth unit vector.

The procedure for determining the Lanczos factorization requires k Lanczos vectors. Beginning with the first Lanczos vector v_1, the other Lanczos vectors are calculated based on the previous Lanczos vectors; specifically,

$$w = \mathbf{Z}v_i - \sum_{j=1}^{i} \langle v_j|\mathbf{Z}v_i\rangle v_j \tag{12}$$

and

$$v_{i+1} = \frac{w}{\|w\|}. \tag{13}$$

After k Lanczos vectors are calculated, **U** can be written as

$$\mathbf{U} = (v_1\, v_2 \cdots v_k) \tag{14}$$

and the elements of **H** are given by

$$H_{ij} = \langle v_i|\mathbf{Z}|v_j\rangle \tag{15}$$

and r is

$$r = \left\| \mathbf{Z}v_k - \sum_{j=1}^{k} \langle v_j|\mathbf{Z}v_k\rangle v_j \right\|. \tag{16}$$

The Lanczos method approximates the eigenvalues of **Z** with the eigenvalues of **H**. To obtain accurate eigenvalues at the upper or the lower ends of the eigenvalue spectrum, one has to increase the size of the Lanczos factorization.

In Section II, we showed that the eigenvalue span of **Z** is very large. In one example, the largest eigenvalue of **Z** is about 2.6×10^9 times as large as the smallest eigenvalue of **Z**. This property makes it possible to determine a few of the largest, nonphysical eigenvalues of **Z** using the Lanczos method but makes it impossible to determine directly the lowest, physical eigenvalues. This is because the largest eigenvalues of **Z** are so much larger than the small ones of interest that the vector space generated by the Krylov procedure—i.e., the vectors generated by successive multiplication of the initial vector by the matrix **Z**—picks out the vector space spanned by the largest eigenvalues of **Z**. This is exactly the opposite of the vector space one is interested in for this application. Therefore only negligible components of the physical eigenvectors are incorporated in the Krylov space until the dimension of the Krylov space approaches the full dimension of **Z**. This property of the Lanczos method is well known and was noted by Miller and coworkers [3d] in their earlier work. In contrast to the Lanczos method, the implicitly restarted Lanczos method

(IRLM) [4] uses a different strategy to calculate a few of the largest or smallest eigenvalues of a matrix. Assuming that one is looking for the k smallest eigenvalues of matrix $\mathbf{Z}$, where k is a fixed prespecified integer of modest size and letting p be another positive integer of modest size, the result of $k + p$ steps of the Lanczos process applied to $\mathbf{Z}$ leads to

$$\mathbf{Z}\mathbf{U}_{k+p} = \mathbf{U}_{k+p}\mathbf{H}_{k+p} + re_{k+p}^{T}; \tag{17}$$

again $\mathbf{U}_{k+p}$ and r satisfy Eqs. (10) and (11) with k now replaced by $k + p$.

Now the eigenvalues of $\mathbf{H}_{k+p}$ may be determined and the p largest eigenvalues are selected to form the set $\{\mu_1, \mu_2, ..., \mu_p\}$. Then for each μ in this set an ''implicit shift'' procedure is applied to the matrices $\mathbf{U}$ and $\mathbf{H}$. The first step of this ''implicit shift'' procedure is to compute the $\mathbf{QR}$-factorization

$$\mathbf{H}_{k+p} - \mu\mathbf{I}_{k+p} = \mathbf{QR}, \tag{18}$$

with $\mathbf{Q}$ being an unitary matrix and $\mathbf{R}$ being an upper triangular matrix. Next $\mathbf{U}_{k+p}$ and H_{k+p} are updated to

$$\mathbf{U}_{k+p} \rightarrow \mathbf{U}_{k+p}\mathbf{Q} \tag{19}$$

$$\mathbf{H}_{k+p} \rightarrow \mathbf{Q}^{\dagger}\mathbf{H}_{k+p}\mathbf{Q}. \tag{20}$$

After p implicit shifts are done, $\mathbf{U}_{k+p}$ and $\mathbf{H}_{k+p}$ are partitioned into

$$\mathbf{U}_{k+p} = (\mathbf{U}_k^{+} \quad \hat{\mathbf{U}}_p) \tag{21}$$

$$\mathbf{H}_{k+p} = \begin{pmatrix} \mathbf{H}_k^{+} & \mathbf{M} \\ \beta_k e_1 e_k^T & \hat{\mathbf{H}}_p \end{pmatrix}. \tag{22}$$

Sorensen [4] showed that $\mathbf{U}_k^{+}$ and $\mathbf{H}_k^{+}$ are a legitimate Lanczos factorization of $\mathbf{Z}$; i.e.,

$$\mathbf{Z}\mathbf{U}_k^{+} = \mathbf{U}_k^{+}\mathbf{H}_k^{+} + r_k^{+}e_k^T \tag{23}$$

and r_k^{+} satisfies Eq. (11).

After the implicit shift series from Eq. (18) to Eq. (23), Eq. (23) is used as the starting point for another Lanczos factorization. The result is a Lanczos factorization similar to that of Eq. (17). At this point the whole process of going from Eq. (17) to Eq. (23) and then back to Eq. (17) starts again. Sorensen [4] also showed that each cycle of this loop replaces the initial vector v_1 by

$$v_1 \rightarrow \frac{1}{\tau}\prod_{j=1}^{p}(\mathbf{Z} - \mu_j\mathbf{I})v_1, \tag{24}$$

where $\mu_1, \mu_2, ..., \mu_p$, are the p shifts and τ is a normalization factor. In this way, the unwanted eigenvectors are cast out from v_1 by the IRLM. This process is continued until the initial vector v_1 is forced into a subspace spanned by the k lowest eigenvectors.

There are several major advantages of the IRLM method over Lanczos methods for this project: (1) The usual Lanczos factorization scheme can generate ''ghost'' eigenvalues when the number of the steps becomes large. IRLM avoids this problem by keeping the number of steps fixed to a moderate value. (2) IRLM is a self-consistent recursive algorithm—numerical errors in the earlier stages are corrected in the later stages. (3) The Lanczos method calculates the k lowest eigenvalues out of a large Lanczos factorization; to keep IRLM process moving, one needs only to calculate the p *largest* eigenvalues of a $k + p$ Lanczos factorization accurately. Numerically the calculation of the *largest* eigenvalues is more favorable because the spacing between the adjacent eigenvalues is larger for larger eigenvalues (see Eq. (8)).

Because of the unique characteristics of the matrix $\mathbf{Z}$ (see Section II for details), the application of the IRLM for this project requires some changes in the standard IRLM. We will describe two of the most important changes needed for this application in the next section.

IV. CHANGES TO THE IRLM

A. *Chebychev Preconditioning*

There are two fundamental computational costs associated with IRLM. One is the cost of the internal numerical operations required to perform the IRLM itself. For each iteration step, this cost is related to the size of the Lanczos factorization $k + p$. The other is the cost associated with performing a matrix–vector product with the matrix $\mathbf{Z}$. Typically in this application, $k + p$ is much smaller than the size of $\mathbf{Z}$ and as a result the matrix–vector operation dominates the computing time.

The second factor that has a significant impact on the total computing time is the number of iterations required to obtain convergence. In general, the IRLM takes fewer iterations when the eigenvalue distribution around the desired eigenvalues is more sparse than the rest of the spectrum. However, from the discussions in Section II, one can see that the eigenvalue distribution of $\mathbf{Z}$ is such that the highest eigenvalues are in the most sparse region of the eigenvalue spectrum—the opposite of the desired distribution.

A standard way to accelerate convergence of a Lanczos algorithm or IRLM is to find the eigenvectors of a polynomial function of the matrix and then to use the eigenvectors to compute the desired eigenvalues of the original matrix via Raleigh quotients; i.e., first compute

$$\phi(\mathbf{Z})q = \mu q \tag{25}$$

for q, then compute λ from

$$\lambda = \frac{q^{\dagger}\mathbf{Z}q}{q^{\dagger}q}. \tag{26}$$

Pendergast, Darakjian, Hayes, and Sorensen [6] showed that a Chebychev polynomial is a good choice for ϕ when obtaining the surface functions for reactive scattering using the Parker–Pack method [7]. The tests carried out as part of this work demonstrate that a Chebychev polynomial can be a very effective preconditioner for this application as well. The Chebychev polynomial φ_m used in this study is defined as

$$\varphi_m(\mathbf{Z}) = \mathbf{I} - \prod_{j=1}^{m}\left(\mathbf{I} - \frac{\mathbf{Z}}{\mu_j}\right), \tag{27}$$

where

$$\mu_j = c + b\cos(\eta_j), \quad \eta_j = \frac{(2j-1)\pi}{2m}, \quad j = 1, 2, ..., m, \tag{28}$$

where c is the center and b is half the length of the interval (a_0, a_1). One can easily verify from Eqs. (27) and (28) that the preconditioned eigenvalue will be within (0, 2), if, as in this application, the original eigenvalues lie in the range $(0, a_1)$. Another desirable property is that, if an eigenvalue λ is in $[a_0, a_1)$, $\varphi_m(\lambda)$ is always greater or equal $\varphi_m(a_0)$; while if an eigenvalue λ is in $(0, a_0)$, $\varphi_m(\lambda)$ is always less than $\varphi_m(a_0)$.

In this application, a_1 is slightly higher than the largest eigenvalue of $\mathbf{Z}$, and a_0 is higher than the largest *physical* eigenvalues of $\mathbf{Z}$. This selection ensures that: (1) after preconditioning, all the eigenvalues will be within (0, 2); (2) the *physical* eigenvalues will still be smaller than *nonphysical* eigenvalues after preconditioning.

The Chebychev preconditioner modifies the eigenvalue distribution in such a way that the resulting eigenvalue distribution (i.e., the Chebychev polynomial evaluated for the eigenvalues of the original matrix) is less dense for the eigenvectors of interest—the physical eigenvectors. This can be seen easily if one examines two eigenvalues λ_1 and λ_2, and assumes that both λ_1 and λ_2 are much less than μ_j, $j = 1, ..., m$, given by Eq. (28). Before the preconditioning, the ratio between $\lambda_1 - \lambda_2$ and the range of the eigenvalues is such that

$$\frac{\lambda_1 - \lambda_2}{\lambda_{\max}} \to e^{-C_2 k_{\max}}. \tag{29}$$

In contrast, the difference between the preconditioned eigenvalues equals

$$\begin{aligned}\lambda_1' - \lambda_2' &= \prod_{j=1}^{m}\left(1 - \frac{\lambda_2}{\mu_j}\right) - \prod_{j=1}^{m}\left(1 - \frac{\lambda_1}{\mu_j}\right) \\ &\approx \sum_{j=1}^{m}\frac{\lambda_1}{\mu_j} - \sum_{j=1}^{m}\frac{\lambda_2}{\mu_j}.\end{aligned} \tag{30}$$

Here we use the approximation that $a_0/\mu_j \approx 0$, $j = 1, ... m$. In the case where $a_0 \ll a_1$, Eq. (30) can be further simplified to

$$\begin{aligned}\lambda_1' - \lambda_2' &\approx (\lambda_1 - \lambda_2)\sum_{j=1}^{m}\frac{1}{\mu_j} \\ &\approx (\lambda_1 - \lambda_2)\frac{2}{a_1}\sum_{j=1}^{m}\frac{1}{1 + \cos\left(\frac{(2j-1)\pi}{2m}\right)} \\ &\approx (\lambda_1 - \lambda_2)\frac{2m^2}{a_1}.\end{aligned} \tag{31}$$

This means that the ratio $(\lambda_1' - \lambda_2')/(\lambda_1 - \lambda_2)$ increases as m^2 for mth order Chebychev preconditioning.

The second positive effect of the Chebychev preconditioning is to shift the nonphysical eigenvalues toward the unwanted end of the resulting spectrum without intermingling them with the eigenvalues associated with the physical eigenvectors. To see this, we define a quantity P as

$$P = -\sum_{j_{\text{nonphysical}}}\left(\ln\frac{\lambda_j - \lambda_{\min}}{\lambda_{\max} - \lambda_{\min}}\right), \tag{32}$$

where $\lambda_{\min}$ is the minimum eigenvalue and $\lambda_{\max}$ is the maximum eigenvalue. It is evident that P is always larger than 0 because $\lambda_j - \lambda_{\min}$ is always less than $\lambda_{\max} - \lambda_{\min}$. Generally speaking, for fixed dimension matrix, the bigger P is, the closer its eigenvalues are to $\lambda_{\min}$.

In one test, before preconditioning, the quantity P for $\mathbf{Z}$ equals 513.8; after preconditioning of order 5, the value of P becomes 278.4; after preconditioning of order 20, the value of P becomes 161.9. This result demonstrates that on the average, $\ln(\lambda_j - \lambda_{\min})/(\lambda_{\max} - \lambda_{\min})$ is reduced to about one-half after Chebychev preconditioning of order 5; $\ln(\lambda_j - \lambda_{\min})/(\lambda_{\max} - \lambda_{\min})$ is reduced to about one-third after Chebychev preconditioning of order 20. In the following sections, it will be evident that our tests show this trend.

B. *Calculation of Multiple Eigenvalues*

In the previous subsection, we discussed utilizing a Chebychev preconditioning of the $\mathbf{Z}$-matrix prior to IRLM iterations. While it might appear that this approach could be used to determine as many physical eigenvectors and eigenvalues as the particular application required, we have not found this to be the case in actual practice. The approach can be used to calculate up to two physical eigenvalues of **Z**. However, when more than two physical eigenvectors are needed we have found that the method frequently encounters numerical problems. In the next paragraphs, we will discuss the reason for this.

After a series of IRLM iterations on the Chebychev preconditioned **Z** matrix, the initial vector v_1 of the Lanczos factorization will *almost* fall in the subspace spanned by the k physical eigenvectors. The reason why v_1 cannot *totally* fall into that subspace is because all of our computations are carried out on finite length floating point numbers. Therefore, instead of

$$v_1 = \sum_{\text{physical}} a_i u_i, \tag{33}$$

one actually gets

$$v_1 = \sum_{\text{physical}} a_i u_i + \sum_{\text{nonphysical}} b_j w_j. \tag{34}$$

Here u_i are physical eigenvectors and w_j are nonphysical eigenvectors. For double precision computations, the $|b_j|$ are typically around 10^{-17}. Although b_j are already small numbers, the impurity within v_1 still makes it difficult or impossible without special provisions to calculate the physical eigenvalues using IRLM alone when the number of physical eigenvalues is larger than 2.

After v_1 is almost in the subspace spanned by the k physical eigenvectors, the IRLM program uses Lanczos factorization to calculate the k physical eigenvectors and eigenvalues. The procedure is as follows: Starting with v_1, the k step Lanczos factorization produces $k - 1$ Lanczos vectors $v_2, ..., v_k$, where v_i is related to $v_1, ..., v_{i-1}$ by

$$\begin{aligned} \eta &= \mathbf{Z} v_{i-1} \\ \hat{v}_i &= \eta - \sum_{j=1}^{i-1} \langle v_j | \eta \rangle v_j \\ v_i &= \frac{\hat{v}_i}{\|\hat{v}_i\|}. \end{aligned} \tag{35}$$

As noted in Section II, the physical eigenvalues of the **Z** matrix generally are greater than one and less than 100, while the largest nonphysical eigenvalue is about 10^{10}. Since the ratio between the largest nonphysical eigenvalue and typical physical eigenvalues is so large, even if $|b_j|$ for the largest nonphysical eigenvalue only equals 10^{-17}, v_3 will already be dominated by the *nonphysical* eigenvectors! Because of this, when k is greater than two, using Lanczos factorization to calculate k physical eigenvalues and eigenvectors simultaneously fails even if v_1 is in the subspace spanned by the k physical eigenvectors within double precision accuracy. This problem cannot be circumvented by calculating the physical eigenvalues one-by-one using IRLM, because the differences between the physical eigenvalues are very small compared to the eigenvalue span of **Z**. As a result, the convergence of IRLM will be very slow if one attempts to compute these eigenvalues separately.

In this study, we have investigated two approaches to calculate the k physical eigenvalues and eigenvectors after v_1 has been projected onto the subspace spanned by the k physical eigenvectors. In the first approach, we choose one vector v_1, then apply the IRLM process to the Chebychev preconditioned matrix to cast out the nonphysical eigenvectors in v_1. After v_1 is mostly forced into a subspace spanned by the k physical eigenvectors, we calculated the physical eigenvectors and eigenvalues using Lanczos factorization with $\mathbf{Z}^{-1}$—the inverse problem. The large ratio between the nonphysical eigenvalues and physical eigenvalues makes the Lanczos factorization on **Z** lose accuracy, but this large ratio will not be a problem for $\mathbf{Z}^{-1}$, because for $\mathbf{Z}^{-1}$ the nonphysical eigenvalues become very small eigenvalues. The main cost of this approach is one IRLM iteration to calculate v_1 succeeded by $k - 1$ solutions of the simultaneous linear equations $\mathbf{Z}x = y$. When the size of **Z** is too large to be stored in core, the solution of $\mathbf{Z}x = y$ can still be obtained through an iterative method, such as the conjugate gradients [8] and generalized minimum residue method (GMRES) [9]. But this can be time consuming because **Z** is a dense matrix.

Manthe and Miller [3d] proposed applying the Lanczos method to the inverse problem $\mathbf{Z}^{-1}$ to calculate the lowest eigenvalues of **Z**. In their approach, the minimum number of solutions of $\mathbf{Z}x = y$ is $k - 1$; usually the number of solutions required will be higher because their initial vector is not in the subspace spanned by the k physical eigenvalues. When the cost of an IRLM iteration is competitive with the solution of $\mathbf{Z}x = y$, our first approach will be favorable because we have limited the number of solutions of $\mathbf{Z}x = y$ to $k - 1$.

In the second approach we calculate multiple physical eigenvalues as follows. Instead of trying to calculate k physical eigenvalues from one "good" initial vector v_1, we calculate them from k "good" initial vectors. First we selected k linear independent vectors t_i, $i = 1, ..., k$; then we apply the IRLM process to the preconditioned matrix for each vector t_i to force t_i into the subspace spanned by the k physical eigenvectors. After most of the nonphysical eigenvector parts are cast out, t_i can be written as

$$t_i = \sum_{j=1}^{k} a_{ij} u_j + \sum_{l=k+1}^{n} b_{il} w_l. \tag{36}$$

Here as in Eq. (34), b_{il} are small numbers—very close to zero. For double precision calculations, $|b_{il}|$ can be as small as 10^{-17}. On the other hand, a_{ij} are generally not close to zero, since physical eigenvector components are not reduced effectively by the IRLM process because these eigenvalues are still very close together, even after preconditioning with a modest order Chebychev polynomial.

Then we construct a linear orthogonal basis from $\{t_i\}$, and name the basis vectors p_i. Next a representation of $\mathbf{Z}$ under basis $\{p_i\}$ is determined; i.e., we calculated a $k \times k$ matrix $\mathbf{Y}$,

$$\mathbf{Y}_{ij} = \langle p_i|\mathbf{Z}|p_j\rangle. \tag{37}$$

Finally, we approximate the physical eigenvalues of $\mathbf{Z}$ by the eigenvalues of $\mathbf{Y}$, and approximate the physical eigenvectors of $\mathbf{Z}$ by

$$q_i = \sum_{j=1}^{k} \gamma_{ij} P_j, \tag{38}$$

where $(\gamma_{i1}, ..., \gamma_{ik})$ is the ith eigenvector of $\mathbf{Y}$.

Unlike the Lanczos factorization, this approach constructs an expansion basis of the physical eigenvectors without obtaining $\mathbf{Z}^n v_1$. This difference is crucial because the operation $\mathbf{Z}^n v_1$ introduces much greater contamination from the nonphysical eigenvector components than the present method produces in v_1. One can see why the approach works by looking at an error estimate for the computed eigenvalues. For each computed approximate eigenvector we may write

$$p_i = \sum_{\text{physical}} \alpha_{ij} u_j + \sum_{\text{nonphysical}} \beta_{il} w_l. \tag{39}$$

As noted previously, the $|\beta_{il}|$ are very close to zero, just like $|b_{il}|$ in Eq. (36), while $|\alpha_{ij}|$ are generally much larger than $|\beta_{il}|$. The numerical errors in $\mathbf{Y}_{ij}$ due to the nonphysical eigenvalue part of p_i are p_j are

$$\Delta Y_{ij} = \sum_{\text{nonphysical}} \beta_{il}^{*} \beta_{jl} \lambda_l. \tag{40}$$

According to Wielandt–Hoffman theory [10], the numerical error of λ_i due to the nonphysical eigenvector components in p_1, ..., p_k is less than

$$\|\Delta\mathbf{Y}\|_F = \sqrt{\sum_{i=1}^{k}\sum_{j=1}^{k} |\Delta Y_{ij}|^2}. \tag{41}$$

Because $|\beta_{il}|$ can be as low as 10^{-17}, while λ_l are at most 10^{10}, from Eq. (41) one finds that the numerical error in λ_i due to the nonphysical eigenvector components can be as low as 10^{-17}! This means that the error can be neglected if the p_k space is close enough to a subspace spanned by the k physical eigenvectors.

Operationally, the first approach needs one IRLM iteration and $k - 1$ solutions of the simultaneously linear equations $\mathbf{Z}x = y$; the second approach, on the other hand, needs k IRLM iterations but no solutions to the linear equations. In the three atom collinear calculations, the first approach is always more efficient than the second approach, because it always takes less time to solve $\mathbf{Z}x = y$ $(k - 1)$ times than to converge $(k - 1)$ additional IRLM iterations. However, regardless of the relative computational time for the collinear system, there are some properties of the second approach that may make it more favorable for the higher dimensional problems that must be solved for full 3D studies of four and five atom chemical reactions. When using the second approach one can always exploit the spareness features of the underlying Hamiltonian needed to determine $\mathbf{Z}$. This is possible because the only requirement for the IRLM process is to be able to compute $\mathbf{Z}$ multiplied into a vector. As a result one does not need to assemble the dense matrix $\mathbf{Z}$. However, in the first approach, which involves the solutions of $\mathbf{Z}x = y$ for x, one must sacrifice the underlying sparseness of the Hamiltonian that is used to determine $\mathbf{Z}$.

When the problem size becomes extremely large, the ability to fully exploit the sparseness of the Hamiltonian becomes a major advantage, since in this case a larger percentage of the required data can be stored in core memory. Another advantage of the second approach is that it has a computational structure with a high level of parallelism—*large-grain* parallelism. Because the k IRLM iterations use unrelated initial vectors, one can perform the IRLM iterations on different workstations or different clusters of processors. Moreover, because each of the IRLM iterations are independent, there is *no* interprocessor communication between the clusters. In addition, each IRLM iteration converges roughly at the same speed; consequently the load balance of this *large-grain* parallelization is very good.

There is another interesting change in the IRLM process for cases where the matrix $\mathbf{Z}$ has more than three physical eigenvalues. In the previous applications of IRLM [4], if one wanted to calculate k_0 lowest eigenvalues of a matrix $\mathbf{A}$, one would set the value of k in Eqs. (17)–(23) equal to k_0. In other words, in every IRLM loop from Eq. (17) to Eq. (23), one first constructs a Lanczos factorization of size $k + p$ and then performs the "implicit shift" p times to purify the vector v_1. In our application, when $k_0 > 3$ and the order of Chebychev preconditioning around 20, the difference between the size of the Lanczos factorization and the number of "shifts" is chosen to be 3 or 2 instead of k_0 (i.e., if the size of Lanczos factorization is nz, we choose $nz - 3$ or $nz - 2$ "shifts" instead of $nz - k_0$

"shifts"). The reason for this is discussed in the following paragraph.

We have shown that even if v_1 is in the subspace spanned by the k physical eigenvectors within double precision accuracy, v_3 of the Lanczos factorization of the $\mathbf{Z}$ matrix is dominated by the *nonphysical* eigenvectors. Following the same line of reasoning, if m is around 20 and the eigenvalue span of $\mathbf{Z}$ is around 10^{10}, with an mth-order Chebychev polynomial of $\mathbf{Z}$ as its primary matrix, the fourth Lanczos vector, v_4, of the Lanczos factorization will also be dominated by the *nonphysical* eigenvectors. After the mth-order Chebychev preconditioning, the *physical* eigenvalues are projected into the interval $(0, t]$, while the *nonphysical* eigenvalues are projected into the interval $(t, 2]$. Since the fourth Lanczos vector is already dominated by the *nonphysical* eigenvectors, the eigenvalues of $\mathbf{H}_{k+p}$ in Eq. (17) contain at most three eigenvalues in the range $(0, t]$. Because ideally one wants the "shifts" to project out all of the eigenvalues of $\mathbf{H}_{k+p}$ in the range $(t, 2]$ and none of the eigenvalues in the range $(0, t]$, it is better to choose $nz - 3$ or $nz - 2$ shifts than to choose only $nz - k_0$.

V. RESULTS AND DISCUSSION

Because a number of modifications of the basic IRLM have been introduced here, we have performed a number of tests to determine the effectiveness of each modification. In this section, we will present our computational results for each test along with analyses of these results.

We begin by comparing the eigenvalue distribution of $\mathbf{Z}$ and that of $\varphi_m(\mathbf{Z})$, where $\varphi_m(\mathbf{Z})$ is a mth-order Chebychev polynomial of $\mathbf{Z}$, which is defined by Eq. (27), for the collinear $H + H_2 \rightarrow H_2 + H$ reaction system. Figure 2 contains a plot of λ_k versus k for $\mathbf{Z}$ for this reaction for a scattering energy, E, equal to 0.86 eV. Figure 3 contains the corresponding plot of λ_k versus k for $\varphi_m(\mathbf{Z})$, where the parameters for the Chebychev polynomial φ_m are $m = 20$, $a_0 = 5 \times 10^4$, and $a_1 = 3.4 \times 10^9$.

From Fig. 2, one can see that the eigenvalue distribution of $\mathbf{Z}$ is very dense in the lower end and is very sparse in the higher end; in fact for $\mathbf{Z}$ the ratio between the number of the eigenvalues in the lower half on the spectrum (i.e., less than $\frac{1}{2}$ of the largest eigenvalue) and the number in the upper half of the eigenvalue spectrum is about 22:1. On the other hand, the eigenvalue distribution of $\varphi_m(\mathbf{Z})$ is shifted significantly towards the higher end of the spectrum. Now, this same ratio of the number of eigenvalues in the upper and lower portions of the spectrum is about 1.8:1.

In Section IV we also introduced another quantity P in Eq. (32) that characterizes the distribution of the eigenvalues. For matrix $\mathbf{Z}$ and $\varphi_m(\mathbf{Z})$, mentioned in the previous paragraphs, the values of P are 513.8 and 161.9, respectively. This means that on the average, $\ln(\lambda_j - \lambda_{\min})/(\lambda_{\max} - \lambda_{\min})$ is reduced to about one third of its original value after Chebychev preconditioning of order 20.

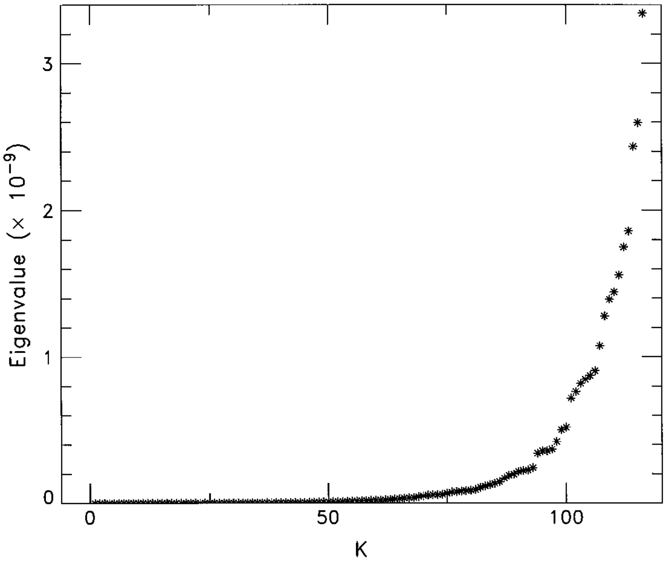

FIG. 2. λ_k vs. k for $\mathbf{Z}$ matrix, where the eigenvalues are arranged in ascending order. $\mathbf{Z}$ is the same as that used in Fig. 1.

Because the eigenvalue distribution shifts toward the higher end of the spectrum after Chebychev preconditioning, the IRLM process converges faster. Table I shows the CPU time and the number of matrix–vector multiplications needed to converge the eigenvectors of $\mathbf{Z}$ for different

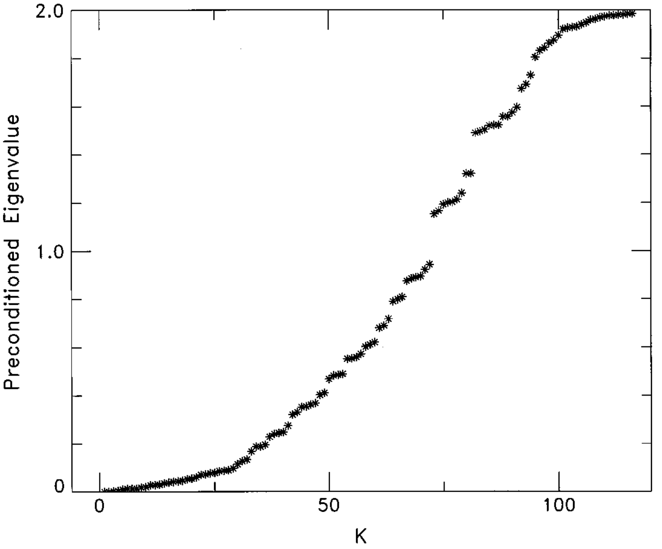

FIG. 3. λ_k vs. k for $\varphi_m(\mathbf{Z})$, where $\varphi_m(\mathbf{Z})$ is a mth-order Chebychev polynomial of $\mathbf{Z}$ given by Eq. (27). The parameters for this plot are: $m = 20$, $a_0 = 5 \times 10^4$, and $a_1 = 3.36 \times 10^9$. In this plot, the eigenvalues of $\varphi_m(\mathbf{Z})$ are arranged in ascending order. $\mathbf{Z}$ is the same as that used in Fig. 1.

TABLE I

CPU Time and the Number of Matrix–Vector Multiplications Needed to Compute the Two Lowest Eigenvalues of **Z** as a Function of the Order of the Chebychev Preconditioner

Order of Chebychev preconditioning	Number of matrix–vector multiplications	CPU time in seconds
2	58056	2116
3	44136	1428
4	29760	897
5	24490	705
6	18876	530
8	16368	442
10	15040	397
12	13536	351
14	11732	301
16	11104	282
18	10224	257
20	8440	212
22	8932	223
24	7968	199
26	7228	179
28	7840	194

Note. **Z** is constructed for an energy of 0.86 eV.

order Chebychev preconditioning polynomials. For this particular example, the *physical* eigenvalues calculated through IRLM differ from those determined using Eispack by less than 10^{-6}. Because about 90% of CPU time is spent on matrix–vector multiplications, the CPU time required is proportional to the number of multiplications calculated. Figure 4 contains a plot of the number of matrix–vector multiplications required to obtain convergence as a function of the Chebychev polynomial order. One can see from this figure that the increase in the order of Chebychev preconditioning polynomial speeds up the rate of convergence. In addition, without Chebychev preconditioning, the IRLM iteration on **Z** does not convergence even after 240,000 iterations. One can also see from this figure that the convergence speed is not improved much as m increases beyond 20. Since higher order Chebychev preconditioning also introduces rounding errors, we use m of moderate value—in this study we use m around 20.

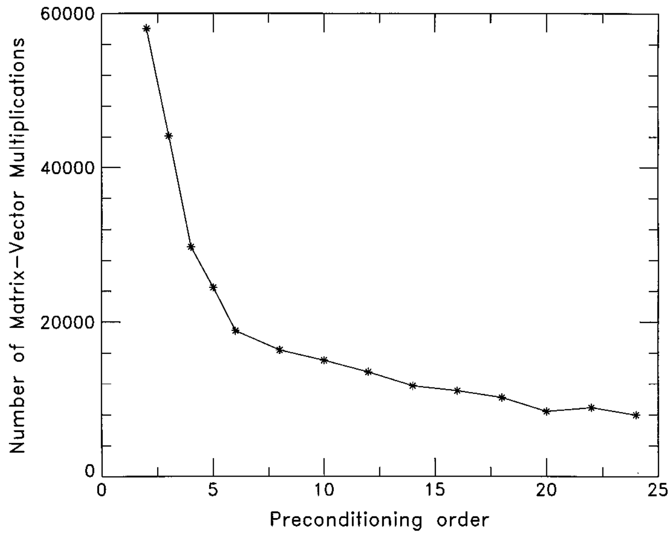

FIG. 4. The number of matrix–vector multiplications required to calculate the two lowest eigenvalues vs. the order of Chebychev polynomial preconditioning. **Z** is the same as that used in Fig. 1.

In order to demonstrate the special treatment for cases with more than two physical eigenvalues, we constructed **Z** for the $H + H_2$ reaction with scattering energy equal to 1.63 eV. The results obtained using the Eispack routine show that in this case there are five physical eigenvalues in **Z**. In the previous section, we asserted that IRLM iterations on the Chebychev preconditioned **Z** matrix can force the initial vector of Lanczos factorization v_1 to fall within the subspace spanned by the physical eigenvectors up to double precision accuracy. To demonstrate this, we computed the overlaps between v_1 and the eigenvectors of **Z** after a certain number of IRLM iteration loops, where the overlap is defined as $|\langle u_i|v_1\rangle|^2$, where u_i are the eigenvectors of **Z** computed using Eispack. Because of the completeness of the eigenvectors of **Z**,

$$\sum_{i=1}^{N} |\langle u_i|v_1\rangle|^2 = 1; \tag{42}$$

therefore $|\langle u_i|v_1\rangle|^2$ measures how v_1 is divided among the eigenvectors of **Z**. In this test, the order of Chebychev preconditioning is 20, while the size of the Lanczos factorization in every IRLM loop is 20 and the number of the "implicit shifts" within each IRLM loop is 17.

Table II shows how the overlaps evolve as the number of IRLM iterations increases. In the first five rows, the overlaps between v_1 and the five physical-eigenvectors are presented. In the last column, the total overlap with the nonphysical eigenvectors are presented, where the last quantity is defined as

$$\sum_{i=6}^{N} |\langle u_i|v_1\rangle|^2. \tag{43}$$

Before the application of IRLM, v_1 is dominated by the nonphysical eigenvectors. In fact, over 95% of v_1 is composed of the nonphysical eigenvectors. But after only 12 IRLM loops, the physical eigenvectors already begin to dominate v_1, after 30 IRLM loops, the nonphysical eigenvectors constitute less than 10^{-22} of v_1! Another observation from Table II is that the relative distribution of v_1 on the five eigenvectors is not significantly affected by IRLM

TABLE II

The Overlaps between v_1 and the Actual Eigenvalues of **Z** versus the Number of IRLM Steps

	Number of IRLM steps			
Overlap norm	0	12	24	30
$\|\langle u_1\|v_1\rangle\|^2$	0.019688	0.470410	0.482841	0.490623
$\|\langle u_2\|v_1\rangle\|^2$	0.000573	0.013687	0.014032	0.014247
$\|\langle u_3\|v_1\rangle\|^2$	0.005967	0.141736	0.144155	0.145624
$\|\langle u_4\|v_1\rangle\|^2$	0.004049	0.095639	0.096417	0.096852
$\|\langle u_5\|v_1\rangle\|^2$	0.012310	0.278526	0.262554	0.252654
Total overlap with nonphysical eigenvectors	0.957	2.0×10^{-6}	1.9×10^{-18}	8.8×10^{-23}

Note. The overlaps between v_1 and the five physical eigenvectors are presented in the first five rows. The total overlap between v_1 and the nonphysical eigenvectors is presented in the last row. **Z** is constructed for an energy of 1.63 eV. The parameters of the Chebychev polynomial used in this table are: $m = 20$, $a_0 = 1.5 \times 10^5$, $a_1 = 2.3 \times 10^{10}$.

iterations, the reason is that the differences among the physical eigenvalues are very small compared with the differences between the physical and nonphysical eigenvalues, the physical eigenvalues appear degenerate to the IRLM routines.

We showed in the previous section that although v_1 is almost in the subspace spanned by the five physical eigenvectors, the impurity within v_1 still makes it impossible to calculate the physical eigenvalues from v_1 alone. However, we also showed that there are two approaches that can be used to overcome the impurity problem. The first approach, namely constructing Lanczos factorization with $\mathbf{Z}^{-1}$ as the primary matrix obviously works; therefore we would like to demonstrate how the second approach works. In the second approach, we first construct k "good" vectors using k different IRLM processes; here "good" means that the vectors are almost in the subspace spanned by the physical eigenvectors. In Table II, we showed that the IRLM process is capable of forcing its initial vector into the subspace spanned by the physical eigenvectors. Therefore, in order to construct k different "good" vectors, we need to perform the IRLM process k times, each time starting with a different initial vector. In order to demonstrate this, we use the same **Z** as in Table II and perform five different IRLM processes. Each IRLM process beginning with a different initial vector and after 24 IRLM loops in each IRLM process we calculated the overlaps of v_1 of each IRLM process with the actual eigenvectors of **Z**. The overlaps of these five different vectors with the physical eigenvectors and the total overlap with the nonphysical eigenvectors are tabulated in Table III. From Table III, it is clear that all five vectors are strongly dominated by the physical eigenvector components. Next we construct a linear orthogonal basis from these five vectors and then calculate the matrix **Y** defined in Eq. (37). Finally, we compute the eigenvalues of **Y**. As shown in the previous section, the resulting eigenvalues of **Y** should be very close to the physical eigenvalues of **Z** (Eq. (41)). The actual results for the H + H_2 case shows that the eigenvalues calculated this way and those calculated using Eispack routines differ by less than 10^{-7}.

In Section IV, we also discussed how many "shifts" should be used after getting Eq. (17) within each IRLM loop when the number of *physical* eigenvalues is larger than or equal to 3. Our conclusion is that the optimal number should be $nz - 3$ or $nz - 2$. In order to illustrate this idea, we ran four IRLM processes. All four processes

TABLE III

The Overlaps between Five Different v_1 with the Eigenvectors of **Z**

	Process number				
Overlap	1	2	3	4	5
$\|\langle u_1\|v_1\rangle\|^2$	0.482842	0.364583	0.199255	0.656706	0.031852
$\|\langle u_2\|v_1\rangle\|^2$	0.014032	0.133521	0.083481	0.033716	0.331736
$\|\langle u_3\|v_1\rangle\|^2$	0.144156	0.057743	0.420015	0.139242	0.198823
$\|\langle u_4\|v_1\rangle\|^2$	0.096417	0.124227	0.148212	0.065702	0.304155
$\|\langle u_5\|v_1\rangle\|^2$	0.262553	0.319926	0.149037	0.104634	0.133434
Total overlap with nonphysical eigenvectors	1.9×10^{-18}	8.7×10^{-18}	7.9×10^{-18}	5.1×10^{-18}	2.6×10^{-17}

Note. Each v_1 is obtained by first choosing a random vector, and then each random vector undergoes a 24-step IRLM "purification." The overlaps between different v_1 and the five physical eigenvectors, u_i, are presented in the first five rows. The total overlap between v_1 and the nonphysical eigenvectors is presented in the last row. **Z** is constructed at the energy of 1.63 eV. The parameters of the Chebychev polynomial used in this table are: $m = 20$, $a_0 = 1.5 \times 10^5$, $a_1 = 2.3 \times 10^{10}$.

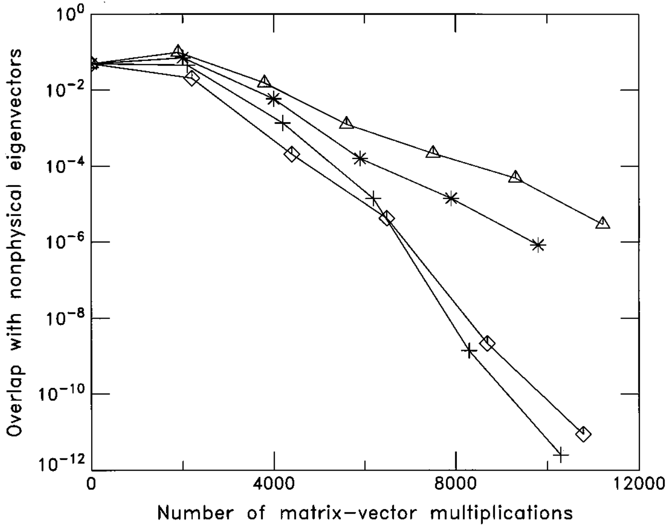

FIG. 5. The total overlap of v_1 with the nonphysical eigenvectors of **Z** (defined by Eq. (43)) vs. the number of matrix–vector multiplications performed. Here the total overlap is plotted on a logarithmic scale. The numbers of "shifts" taken as each IRLM iteration loop are different for the four curves in this plot. For "diamond," "cross," "asterisk," and "triangle" curves, the number of "shifts" are $nz - 2$, $nz - 3$, $nz - 4$, and $nz - 5$, respectively, where nz is the size of the intermediate Lanczos factorizations. **Z** is constructed for collinear $H + H_2 \rightarrow H_2 + H$ reaction for a scattering energy of 1.63 eV.

use a 20th-order Chebychev polynomial of the **Z** matrix; all four processes started with the same initial vectors and, nz, the sizes of the intermediate Lanczos factorizations for the four processes, are also the same; their values are 20. However, the number of "shifts" for the four IRLM processes are different; their values are $nz - 2$, $nz - 3$, $nz - 4$, and $nz - 5$, respectively. In order to monitor the convergence, we computed the total overlap of initial vector v_1 with the nonphysical eigenvectors of **Z** periodically. The definition of the total overlap is given by Eq. (43). We plot the total overlap versus the number of matrix–vector multiplications on a semilogarithm plot in Fig. 5. The reason for choosing the number of matrix–vector multiplications, instead of number of IRLM loops, is that a different number of "shifts" results in a slightly different number of matrix–vector multiplications performed within each IRLM loop. Using the number of matrix–vector multiplications as the variable more accurately reflects the computational cost.

Figure 5 shows that choosing $nz - 2$ shifts is slightly better than choosing $nz - 3$ shifts when impurity in the initial vector v_1 is higher than 10^{-5}. When the impurity in v_1 is lower than 10^{-5}, choosing $nz - 3$ shifts is slightly better than choosing $nz - 2$ shifts, but overall the two choices are about equally good. Choosing $nz - 4$ or $nz - 5$ shifts is definitely poorer than the first two choices. The differences between the first two choices and the last two choices are obvious when the impurity in v_1 is less than 10^{-4}.

In these tests on the collinear $H + H_2 \rightarrow H + H_2$ reaction, the CPU time required to compute the eigenvalues of **Z** is at least 100 times longer for our approaches than that of Eispack routines applied directly to the inverse of the matrix, as implemented by Miller and coworkers [3]. This is especially true for the second approach of Section V that we proposed to calculate multiple physical eigenvalues, because we need to run k IRLM processes to get k "good" vectors. Therefore, the computational cost of the second approach is also proportional to k, the number of physical eigenvectors of **Z**. However, the relative speed can be expected to change when one is dealing with much bigger problems (e.g., three atom 3D reactions and four atom 3D reactions) because in these larger systems, the sparseness features in the Hamiltonian will be much more significant. Since the IRLM algorithm can take full advantage of these sparseness features the method scales more favorably than traditional eigenvalue methods as the length of the vector is increased.

ACKNOWLEDGMENTS

Partial support for this project came from National Science Foundation project (ASC-9408795); additional support was also provided by the National Science Foundation in the form of a postdoctoral fellowship grant (ASC-9504071). Computer time on the Cray-YMP was provided by a grant from the Ohio Supercomputer Center. The authors also thank Danny Sorensen and Zdenko Tomašić for many thoughtful discussions concerning the effective use of the implicitly restarted Lanczos method and the advantages of using Chebychev preconditioning to accelerate the convergence of this method. We thank C-H. Huang, P. Sadayappan, and Gregory Parker for helpful discussions of factors affecting the overall performance of the computation codes. Finally, the authors thank Prakashan Korambath and Phil Pendergast for discussions concerning matrix–vector routines and for assistance in preparing this manuscript.

REFERENCES

1. For example, (a) A. Kuppermann and Y. M. Wu, *Chem. Phys. Lett.* **205,** 577 (1993); (b) J. D. Kress, Z. Bačcić, G. A. Parker, and R. T Pack, *Chem. Phys. Lett.* **170,** 306 (1990); (c) J. D. Kress, R. B. Walker, and E. F. Walker, *J. Chem. Phys.* **93,** 8085 (1990); (d) Z. Darakjian, E. F. Hayes, G. A. Parker, E. A. Butcher, and J. D. Kress, *J. Chem. Phys.* **95,** 2516 (1991); (e) Z. Darakjian, P. Pendergast, and E. F. Hayes, *J. Chem. Phys.* **102,** 4461 (1995); (f) J. Z. H. Zhang and W. H. Miller, *J. Chem. Phys.* **92,** 1811 (1990); (g) D. E. Manolopoulos, M. D'Mello, and R. E. Wyatt, *J. Chem. Phys.* **93,** 403 (1990); (h) N. C. Blais, M. Zhao, D. G. Truhlar, D. W. Schwenke, and D. J. Kouri, *Chem. Phys. Lett.* **166,** 11 (1990).
2. (a) D. Neuhauser, *J. Chem. Phys.* **100,** 9272 (1994); (b) D. H. Zhang and J. Z. H. Zhang, *J. Chem. Phys.* **101,** 1146 (1994); (c) U. Manthe, T. Seideman, and W. H. Miller, *J. Chem. Phys.* **99,** 10078 (1993); (d) E. M. Goldfield, S. K. Gray, and G. C. Schatz, *J. Chem. Phys.* **102,** 8807 (1995).
3. (a) T. Seideman and W. H. Miller, *J. Chem. Phys.* **96,** 4412 (1992);

(b) **97,** 2499 (1992); (c) W. H. Miller, *Acc. Chem. Res.* **26,** 174 (1993); (d) U. Manthe and W. H. Miller, *J. Chem. Phys.* **99,** 3411 (1993).

4. D. C. Sorensen, *SIAM J. Matrix Anal. Appl.* **13,** 357 (1992).
5. C. Lanczos, *J. Res. Nat. Bur. Stand.* **45,** 255 (1950).
6. P. Pendergast, Z. Darakjian, E. F. Hayes, and D. C. Sorensen, *J. Comput. Phys.* **113,** 201 (1994).
7. R. T Pack and G. A. Parker, *J. Chem. Phys.* **87,** 3888 (1987).
8. J. E. Dennis Jr. and K. Turner, *Linear Alg. Appl.* **88,** 187 (1987).
9. (a) Y. Saad and M. Schultz, *SIAM J. Sci. and Stat. Comp.* **7,** 856 (1986); (b) H. F. Walker, *SIAM J. Sci. and Stat. Comp.* **9,** 152 (1988).
10. J. H. Wilkinson, *The Algebraic Eigenvalue Problem* (Claredon Press, Oxford, 1965), p. 104.