Minimum-Error Method for Scattering Problems in Quantum Mechanics: Two Stable and Efficient Implementations

Burcin Temel,[†] Greg Mills,[†] and Horia Metiu*,[‡]

Departments of Chemistry and Biochemistry, University of California at Santa Barbara, Santa Barbara, California 93106-9510

Received: May 15, 2006; In Final Form: July 10, 2006

We examine here, by using a simple example, two implementations of the minimum error method (MEM), a least-squares minimization for scattering problems in quantum mechanics, and show that they provide an efficient, numerically stable alternative to Kohn variational principle. MEM defines an error-functional consisting of the sum of the values of $(H\Psi - E\Psi)^2$ at a set of grid points. The wave function Ψ, is forced to satisfy the scattering boundary conditions and is determined by minimizing the least-squares error. We study two implementations of this idea. In one, we represent the wave function as a linear combination of Chebyshev polynomials and minimize the error by varying the coefficients of the expansion and the **R**-matrix (present in the asymptotic form of Ψ). This leads to a linear equation for the coefficients and the **R**-matrix, which we solve by matrix inversion. In the other implementation, we use a conjugate-gradient procedure to minimize the error with respect to the values of Ψ at the grid points and the **R**-matrix. The use of the Chebyshev polynomials allows an efficient and accurate calculation of the derivative of the wave function, by using Fast Chebyshev Transforms. We find that, unlike KVP, MEM is numerically stable when we use the **R**-matrix asymptotic condition and gives accurate wave functions in the interaction region.

I. Introduction

The Kohn variational principle (KVP) has been widely and successfully[1−14] used to solve problems in physical chemistry. However, some difficulties remain.[15−17] One of them is the discrepancy between calculations using different forms of the asymptotic wave function. One can write the asymptotic form in terms of a phase shift, an **R**-matrix or an **S**-matrix.[18] The form of the variational principle depends on the boundary conditions used: the one using the **S**-matrix differs from the one using the **R**-matrix etc. One test of the adequacy of KVP is that the values of the **R**-matrix, **S**-matrix and the phase shift calculated variationally, must satisfy the exact relationships connecting them. Often this is not the case, and sometimes, these differences can be substantial: for example, the **R**-matrix may give a false resonance at an energy for which the phase shift is correct.[15−17] Such discrepancies signal that the theory has numerical instabilities.

Later work[1−14] has shown that the use of the **S**-matrix boundary condition

$$\Psi(r) = -e^{-ikr} + Se^{ikr} \tag{1}$$

leads to stable and accurate results and this form has been widely used in applications. The **S**-matrix KVP (SKVP) forces us to use complex arithmetic, which roughly doubles the computer time, as compared to the **R**-matrix, which requires real arithmetic. One would like therefore to have a reliable variational principle for the **R**-matrix.

Some doubts about the reliability of SKVP still persist[19−23] on account that KVP assumes that the second-order functional variation term is negligible and this might not be true at certain energies. To cure this deficiency, Rudge[22] proposed an implementation of the variational principle in which the second-order term is forced to be zero. This modification seems to resolve the deficiencies of the old method: there are no false resonances, and the phase shift, **R**-matrix, and **S**-matrix then satisfy (within the numerical errors) the exact equations connecting them. However, the additional condition makes the method less efficient than the traditional Kohn variational principle.

In this article, we examine a method proposed by Bardsley, Gerjuoy and Sukumar,[24] which requires that the wave function satisfies the Schrödinger equation, and the boundary conditions, at a number of grid points, in a least-squares sense. In other words, we find the wave function by minimizing the least-squares error:

$$F \equiv \sum_{i=1}^{N} w_i[(\hat{H} - E)\Psi]_i^*[(\hat{H} - E)\Psi]_i \tag{2}$$

Here, $[(\hat{H} - E)\Psi]_i$ is the value of $\hat{H}\Psi - E\Psi$ at a grid pointr_i; $\hat{H}$ is the Hamiltonian and E is the total energy. The numbers w_i are weights, for which various choices are possible. The simplest, is to take them all equal to 1; another is to take them large for those grid points for which high accuracy is desired, and small for the others. In what follows we use the weights employed in the Gauss integration method.[25] With this choice, the error (eq 2) becomes a discrete approximation to the continuum expression,

$$F = \langle(\hat{H} - E)\Psi|(\hat{H} - E)\Psi\rangle \tag{3}$$

This method involves no intrinsic approximation (such as the neglect of the second order variation term). Furthermore, if the minimization algorithm is numerically stable, then the procedure

---

* Corresponding author. E-mail: metiu@chem.ucsb.edu. Phone: (805) 893-2256. Fax: (805) 893-4120.

[†] Department of Chemistry.

[‡] Department of Biochemistry.

10.1021/jp0629691 CCC: $33.50

should work equally well with all boundary conditions. Using the **R**-matrix boundary condition,

$$\Psi(r) = \sin kr + R \cos kr \tag{4}$$

is then computationally advantageous, because we avoid using complex numbers in the program.

KVP has been designed to calculate the **S**-matrix, or the phase shift, or the **R**-matrix. Any one of these quantities provides a complete description of the measurements performed in a scattering experiment. There are, however, situations for which the wave function in the interaction region is needed. One example is photon-induced association of two colliding atoms. This can take place through photon absorption, by two colliding atoms, to form a bound, electronically excited state of the diatomic molecule, followed by spontaneous emission of a photon to bring the molecule to the bound, electronic ground state. Or, the molecule may be formed due to emission (stimulated or spontaneous) of a photon during the collision. These processes play a role in the formation of molecules in outer space; they can also be performed in the laboratory. To calculate the rates of these processes, we must know the wave function of the colliding atoms when they are sufficiently close to each other to interact. In principle, KVP provides the wave function in the interaction region but we do not know whether this is given accurately. The least-squares method forces the wave function to satisfy the Schrödinger equation everywhere, in the asymptotic as well as the interaction region.

The MEM varies the wave function Ψ to minimize the error F, while making sure that Ψ also satisfies the scattering boundary conditions.[24,26−32] We have experimented with two methods for minimizing the error. In one we take as variables the values of the wave function at the grid points and the **R**-matrix and vary them until F is minimized. To some extent this method resembles the discrete variable representation (DVR) of Light and co-workers.[33−35] A key element in our implementation is the accurate and efficient calculation of the derivatives of the wave function (the kinetic energy operator acting on Ψ) at the grid points. For this purpose, we use the values of the wave function on the grid to perform a Chebyshev interpolation of the wave function. This can be used to calculate the derivative of the wave function very accurately, in roughly $O(N \log_2 N)$ operations.[25,36−38]

In the other procedure, we represent the wave function as a sum of Chebyshev polynomials and vary the coefficients of this expansion and the **R**-matrix, to minimize the error F. The Chebyshev representation allows an efficient and accurate calculation of the derivatives.

We find that, unlike KVP, the minimum error method (MEM), with the boundary condition eq 4, is numerically stable and it gives the wave function more accurately in the interaction region than the **S**-matrix KVP (SKVP). In some of the previous work[1,11,22] the wave function in the interaction region was represented by a polynomial in the interatomic distance r (this corresponds to the Frobenius method for solving differential equations). This representation is accurate when r is small and deteriorates as r increases. We find that increasing the order of the "Frobenius polynomial" leads to numerical instabilities. This prevents one from testing whether the results are converged. The Chebyshev basis set represents the wave function well globally, it is orthonormal and the method of computation is stable with respect to the increase in the size of the basis set.

Here, we explain how this method is implemented and study its numerical stability by solving a very simple problem: elastic, s-wave scattering by an exponentially attractive potential. We have chosen this model because it has an analytic solution for the **S**-matrix and the wave function in the interaction region. This is essential, when problems that are likely to be numerically unstable are studied, because comparison between numerical methods is inconclusive due to the possibility that both methods have errors. In addition, this model has been used extensively to test various versions of the Kohn variational principle,[4,5,11] and we can use much of the previous numerical work for comparison to the results given by the present method.

II. The Model

The Hamiltonian (in atomic units) for the s-wave scattering of an electron by an exponential potential is

$$\hat{H} = -\frac{1}{2}\frac{\mathrm{d}^2}{\mathrm{d}r^2} - \mathrm{e}^{-r} \tag{5}$$

The trial wave function, which will be varied to minimize the error F, is

$$\Psi_t(r) = (1 - f(r))\Psi_\mathrm{a}(r) + f(r)\,\Psi_\mathrm{I}(r) \tag{6}$$

The form of the asymptotic wave function Ψ_a is

$$(1 - f(r))\Psi_\mathrm{a}(r) = \chi_1(r) + c_0\chi_2(r) \tag{7}$$

When we use the **R**-matrix asymptotic wave function, eq 4,

$$\chi_1(r) = (1 - f(r))\sin(kr) \qquad \chi_2(r) = (1 - f(r))\cos(kr) \tag{8}$$

and c_0 is equal to **R**.

When we use the **S**-matrix asymptotic form, eq 1,

$$\chi_1(r) = -(1 - f(r))\exp(-\mathrm{i}kr) \qquad \chi_2(r) = (1 - f(r))\exp(\mathrm{i}kr) \tag{9}$$

and c_0 is the **S**-matrix. k is the wave vector of the incoming electron, whose energy is $E = k^2/2\mathrm{au}$.

We follow ref 15 and use the cutoff function

$$f(r) = \mathrm{e}^{-\alpha r} \tag{10}$$

Its role is to switch Ψ_t smoothly, between Ψ_I andΨ_a, as r increases beyond the interaction region. The cutoff also makes $(1 - f(r))\Psi_\mathrm{a}(r)$ equal to 0 when r is small, allowing us to impose the correct boundary condition for $r = 0$ (see below).

We have tried a variety of forms for the cutoff function and found that, for this particular problem, the best performance was obtained by eq 10 with $\alpha = 1$. With this choice, the cutoff function matches the range of the exponential potential. In calculations on other problems we get better results (i.e., faster convergence during minimization) by using $(1 - f(r))\Psi_\mathrm{a}(r) + \Psi_\mathrm{I}(r)$ with a Fermi function for $f(r)$.

We determine the wave function $\Psi_\mathrm{I}(r)$ in the interaction region and the coefficient c_0 (which is either the **R**- or the **S**-matrix), by minimizing the error F. We do this by two methods:

In one we write $\Psi_\mathrm{I}(r)$ as a linear combination of Chebyshev polynomials with unknown coefficients $\{c_1, ..., c_N\}$. Introducing the resulting expression for Ψ_t in eq 3 makes F a quadratic form in $\{c_0, c_1, ..., c_N\}$. Minimizing F with respect to these variables leads to a linear equation for them, which we solve numerically by matrix inversion. We call this method a least-squares method with a spectral representation and matrix inversion (MEM-SRMI).

The second method uses c_0 and the values of the function Ψ_I at the grid points r_i as unknowns. We express F in terms of these quantities and vary them (and c_0) to minimize it. The minimization is performed with a preconditioned conjugate-gradient method. We call this the minimum error method with a pseudospectral representation and conjugate gradient (MEM-PSRCG).

III. Minimum Error Method with Spectral Representation and Matrix Inversion (MEM-SRMI)

The trial wave function in this method is

$$\Psi_t(r) = (1 - f(r)(\chi_1(r) + c_0\chi_2(r)) + f(r)\sum_{\alpha=1}^{N} c_\alpha\varphi_\alpha(r) \qquad (11)$$

where

$$\varphi_\alpha(r) = \left(T_\alpha\left(\frac{2r}{a} - 1\right) - T_\alpha(-1)\right)$$
$$\chi_1(r) = \sin kr$$
$$\chi_2(r) = \cos kr \qquad (12)$$

The Chebyshev polynomials $T_\alpha(x)$ are only defined for $x \in [-1, 1]$. Because of this, the argument in $T_\alpha(-1 + 2r/a)$is chosen so that when r varies from 0 to a, $(-1 + 2r/a)$ varies between -1 and $+1$. Here a is the point on the r axis where the potential becomes zero; we have taken it to be $a = 16$ au. The term $T_\alpha(-1)$ ensures that the interaction wave function Ψ_I goes to 0 like r, when r approaches the origin, as required by the boundary conditions for s-wave scattering.

We determine the unknown coefficients $c \equiv \{c_0, c_1, ..., c_N\}$ by introducing eq 11 for Ψ_t in the expression eq 2 and minimizing it with respect to the c_i's. The expression obtained by inserting Ψ_t in eq 11 is a quadratic form in the coefficients c_i:

$$F = \sum_{\alpha=0}^{N}\sum_{\beta=0}^{N} c_\alpha^* c_\beta M_{\alpha\beta} + \left(\sum_{\alpha=0}^{N} c_\alpha^* N_\alpha + \text{complex conjugate}\right) + D \qquad (13)$$

When the **R**-matrix asymptotic form is used, all quantities in eq 13 are real numbers.

The mathematical structure of these equations is very similar to that obtained in ref 15. This is not accidental: the least-squares error functional minimized here is quadratic in the wave function, as is the Kohn functional for the **S**-matrix. This makes much of the mathematics of the two procedures very similar. However, we use the **R**-matrix asymptotic form, avoiding the need for complex quantities—required in previous work—and thus reducing the computational burden.

The minimization of F leads to a linear equation for the c_i's, whose matrix is ill-conditioned; we invert it my using singular value decomposition.

Previous work has found that KVP with the **R**-matrix boundary condition is numerically unstable, producing large errors in certain energy ranges. We find no such instability for MEM-SRMI. Because KVP is stable when the **S**-matrix boundary conditions are used, we have also performed MEM-SRMI calculations with **S**-matrix boundary conditions and the basis set

$$\varphi_\alpha(r) = r^\alpha \qquad (14)$$

used by Miller et al.[11] Our purpose was to test whether the **S**-matrix version of MEM-SRMI has any advantages over the version using the **R**-matrix. We found that within MEM, the numerical stability of the **R**-matrix procedure is as good as that for the **S**-matrix formulation. The **R**-matrix form is preferable because it avoids using complex numbers.

The MEM-SRMI with **S**-matrix boundary condition approach has a major weakness, shared with KVP. The resulting expressions are rather complicated, there are many integrals to perform and these require a large number of operations. This is why we have decided to develop an alternative method (see section IV). This takes advantage of the properties of the Chebyshev polynomials to calculate the derivatives at the grid points and perform various matrix manipulations in $N \log_2 N$ operations. In addition, the direct matrix inversion is less efficient than the minimization of F by an iterative method, such as the conjugated-gradient method.

IV. Pseudospectral Representation of the Wave Function and the Conjugate-Gradient Minimization of *F* (MEM-PSRCG)

The unknown quantity in this procedure is the vector

$$x = \{\Psi_2, ..., \Psi_N, c_0\} \qquad (15)$$

where Ψ_i, $i = 2, ..., N$, are the values of $\Psi_I(r)$ at $N - 1$ grid points r_i. The boundary conditions give

$$\Psi_1 = \Psi_{N+1} = 0 \qquad (16)$$

Note that using the wave function at the grid points as unknown quantities allows an easy implementation of the boundary conditions. These reduce the number of unknown quantities; for higher dimensionality problems, this provides a substantial advantage over the method that uses the expansion coefficients $c_1, ..., c_{N+1}$ as unknowns.

We determine the value of x by minimizing $F(x)$. For this we use a conjugate-gradient procedure,[37] which requires us to calculate efficiently $F(x)$ and its derivatives with respect to Ψ_1, ..., Ψ_{N+1}, c_0. The major difficulty in this calculation is the accurate evaluation of the kinetic energy, in $(\hat{H} - E)\Psi_I$, when we only know the values of Ψ_I at a set of grid points.

The procedure by which we evaluate $F(\Psi_I(2), ...,\Psi_I(N), c_0)$ and the gradient of F

$$\nabla F(x^k) = \left\{0, \frac{\partial F}{\partial \Psi_I(2)}, ..., \frac{\partial F}{\partial \Psi_I(N)}, 0, \frac{\partial F}{\partial c_0}\right\} \qquad (17)$$

is described in Appendix C.

Normally this is all that is needed for minimizing F with respect to $\Psi_I(2)$, ..., $\Psi_I(N)$, c_0. Unfortunately, the matrix giving F as a quadratic form in these variables is ill conditioned; its largest eigenvalue exceeds the smallest one by 8 orders of magnitude. This means that without preconditioning the convergence of the conjugate-gradient search will be slow. Preconditioning requires that we find a similarity transformation that turns the matrix into one whose eigenvalues are of comparable magnitude to each other. Unfortunately, there is no systematic and logical prescription for doing this; preconditioning is an art form, with various recipes offered for each specific situation.[39] The manner in which we precondition the steepest descent direction is explained in Appendix D.

TABLE 1: Dependence of the R-Matrix (c_0) on the Order N of the Polynomial Used To Represent the Wave Function in the Interaction Region[a]

N	MEM-SRMI R	SKVP Re(R)	SKVP Im(R)
8	−1.72472	−1.74494	$7.2080\ 10^{-6}$
12	−1.74517	−1.74494	$1.0129\ 10^{-4}$
18	−1.74493	−1.74493	$9.5500\ 10^{-3}$
19	−1.74494	−1.74490	$1.3202\ 10^{-2}$
20	−1.74494	−1.74467	$3.5728\ 10^{-2}$
25	−1.74494	−0.68491	$82.667\ 10^{-1}$
30	−1.74494	−1.60068	$4.6443\ 10^{-1}$

[a] The SKVP (**S**-matrix Kohn variational principle) results were obtained by using the method of ref 15 to calculate the **S**-matrix; the **R**-matrix was calculated with the exact relationship connecting **S** to **R**. MEM-SRMI is the minimum error method with a spectral representation and matrix inversion. The wave vector of the incident particle is $k = 0.15$ (au) and the exact value of R is $R = -1.744\,94$.

The minimization scheme iterates the vector $x = \{\Psi_2, ..., \Psi_N, c_0\}$ to generate a sequence $x^{(0)}, x^{(1)}, ..., x^{(k)}, ...$ that converges to the value that minimizes $F(x)$. Here $x^{(0)}$ is a guess that starts the iteration. The scheme generating the $x^{(k+1)}$ value out of $x^{(k)}$ is explained below. First, we calculate a new conjugate-gradient direction $d^{(k)}$ by using

$$d^{(k)} = -\nabla F^{(k)} + \beta^{(k)} d^{(k-1)} \tag{18}$$

$-\nabla F^{(k)}$ is the steepest descent direction at the point $x^{(k)}$, and $\beta^{(k)}$ is calculated with the Fletcher−Reeves formula[37]

$$\beta^{(k)} = \frac{\nabla F^{(k)} \cdot G^{(k)}}{\nabla F^{(k-1)} \cdot G^{(k-1)}} \tag{19}$$

$G^{(k)}$ is the preconditioned gradient at the point $x^{(k)}$, whose evaluation is explained in Appendix D. This is where the conjugate-gradient procedure used in the present work differs from the standard one.[37] Next, we set

$$x^{(k+1)} = x^{(k)} + \alpha^{(k)} d^{(k)} \tag{20}$$

and calculate $\alpha^{(k)}$ by minimizing, with Brent's method,[37] $F(x^{(k)} + \alpha^{(k)} d^{(k)})$ with respect to $\alpha^{(k)}$.

The iteration scheme is initialized, by guessing the values of $\Psi_t(2)$, ..., $\Psi_t(N)$, c_0. We do this by assuming that Ψ_t is equal toΨ_a, and by giving c_0 a physically reasonable value. The efficiency of the scheme is not very sensitive to the initial choice.

The preconditioned conjugated gradient scheme is particularly efficient for minimizing quadratic functions of many variables.[39]

V. Results

A. Convergence with Respect to the Basis Set. In MEM-SRMI we represent the wave function in the interaction region as a sum of Chebyshev polynomials. Previous calculations,[11] using SKVP, wrote it as a sum of powers of r. In Tables 1 and 2 we show how the two methods converge with the maximum number of polynomials used to represent the wave function, for two different wave vectors. In both cases, the MEM-SRMI requires a sum of 18 Chebyshev polynomials, whereas SKVP obtains results of the same accuracy with an ordinary polynomial of order 8 (note, however, that the accuracy involved is much higher than that needed in applications). This is puzzling, because a sum of Chebyshev polynomials represents more accurately a function than an ordinary polynomial of the same order.[40] Our method requires a larger basis set because it is more demanding: it demands that the wave function is well represented everywhere. The SKVP, on the other hand, calculates directly the **S**-matrix and has no specific requirements for the accuracy of the wave function in the interaction region. As one can see below (Figures 2a,b), an SKVP calculation that provides a numerically exact value for the **S**-matrix produces an inaccurate wave function, whereas the MEM wave function is very accurate.

TABLE 2: Dependence of the R-Matrix (c_0) on the Order N of the Polynomial Used To Represent the Wave Function in the Interaction Region[a]

N	MEM-SRMI R	SKVP Re(R)	SKVP Im(R)
8	2.19872	2.20038	$2.8418\ 10^{-5}$
12	2.20041	2.20038	$1.0349\ 10^{-4}$
18	2.20039	2.20038	$4.1457\ 10^{-4}$
19	2.20038	2.20038	$1.1331\ 10^{-3}$
20	2.20038	2.20020	$1.7192\ 10^{-2}$
25	2.20038	2.15436	$6.8327\ 10^{-1}$
30	2.20038	0.23585	$10.557\ 10^{-1}$

[a] Same as Table 1, but $k = 0.55$ (au) and the exact value of R is $R = 2.200\,38$.

TABLE 3: R-Matrix Values for the Incident Wave Vectors, in Which RKVP Gives False Resonances[a]

k (au)	0.28100	0.28101	0.28102	0.28103
R-exact	−18.9064	−18.9159	−18.9255	−18.9350
R-KVP	14.3845	74.4587	−161.069	−60.1516
R-MEM-SRMI	−18.9065	−18.9157	−18.9257	−18.9353
R-MEM-PSRCG	−18.8880	−18.8086	−18.8821	−18.7409
R−SKVP	−18.9066	−18.9162	−18.9257	−18.9353

[a] RKVP values are the results of calculations with the Kohn variational principle for the **R**-matrix[15] (c_0). R-SKVP values were obtained by using SKVP to calculate the **S**-matrix, which was then used to obtain **R**-matrix values. MEM-SRMI and MEM-PSRCG are defined in the article.

Because we know the exact wave function Ψ we can fit it, and various terms of $\hat{H}\Psi$, to find out why so many Chebyshev polynomials are required for a good fit. Three Chebyshev polynomials give an excellent fit of the exact wave function. The most demanding part in the calculation of $\hat{H}\Psi$ is the kinetic energy $d^2\Psi/dr^2$. An accurate fit of F requires 18 Chebyshev polynomials, mostly because the square of $d^2\Psi/dr^2$ is present in it and we need to represent it correctly at all grid points.

The "ordinary" power series expansion used sometimes in SKVP becomes very inaccurate as the highest exponent in the expansion increases (see Table 1). This happens because the term $f(r)r^n$ does not go to zero fast enough with r, when n is large. Because of this, the interacting wave function spills into the asymptotic region and this causes errors. The Chebyshev polynomials do not have this problem because they take values between −1 and +1.

B. Numerical Stability of MEM with the R-Matrix Asymptotic Wave Function. Several studies[15−17] have shown that KVP with the **R**-matrix asymptotic wave function is numerically unstable at certain incident energies. The use of KVP became possible only after it was discovered[4] that the **S**-matrix version is numerically stable. To determine whether MEM-SRMI and MEM-PSRCG are stable, when used with the **R**-matrix asymptotic wave function, we have calculated **R** with these methods, at incident energies for which RKVP has difficulties.[15] The results of these calculations are presented in Table 3. A plot of **R** versus k is shown in Figure 1. Both MEM methods with the **R**-matrix asymptotic condition have excellent stability. This is gratifying because we no longer need to use complex numbers in the calculation, thus cutting the computer time by half.

C. Accuracy of the Wave Function. As we have explained in the Introduction, in some cases one is interested in obtaining

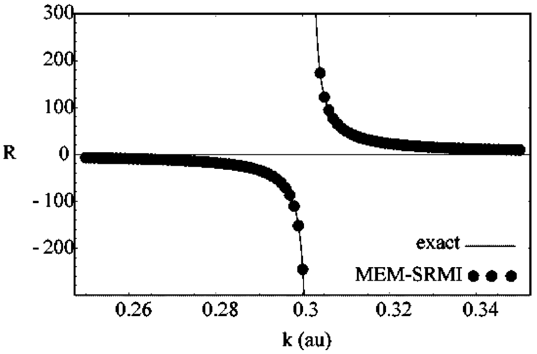

Figure 1. Dependence of the **R**-matrix on the wave vector of the incident electron, around a resonance.

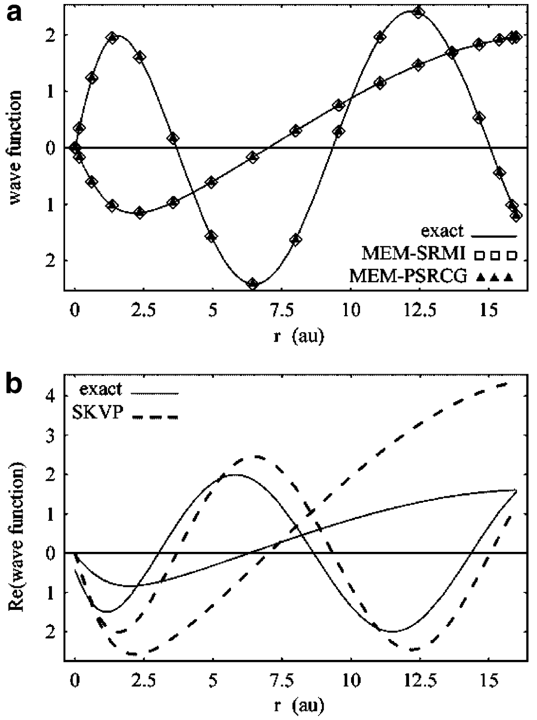

Figure 2. Wave functions in the interaction region. (a) Exact wave function (full line) and the ones obtained with MEM-SRMI (boxes) and MEM-PSRCG (full triangles), for two incident wave vectors ($k = 0.15$ au and $k = 0.55$ au). The high energy wave functions have more oscillations. (b) Real part of the wave function in the interaction region obtained by SKVP for the same incident wave vectors as in (a). The full line is the exact wave function, and the dashed line is the one calculated by SKVP. The high energy wave functions have more oscillations ($k = 0.15$ au and $k = 0.55$ au).

not only the **R**- or **S**-matrix but also the wave function. Both KVP and MEM allow the calculation of the wave function. Here we examine how accurate these wave functions are, when obtained in calculations that give the **R**- or the **S**-matrix very accurately.

The exact wave functions, obtained by using the **R**-matrix asymptotic condition, are plotted in Figure 2a, along with the wave functions generated by MEM-SRMI and MEM-PSRCG. The agreement with the exact wave functions is excellent. The real parts of the wave functions produced by SKVP are shown in Figure 2b. The errors of the wave functions in the interaction region are quite large. These "numerical experiments" indicate that SKVP can give erroneous wave functions in calculations that give excellent results for the **S**-matrix.

D. Convergence of the Conjugate-Gradient Method. The matrix inversion method is applicable only when the functional F is a bilinear function of the unknown parameters in the trial wave function Ψ_t. Often using a nonlinear dependence on parameters is advantageous, because it provides a more flexible representation of Ψ_t. In such cases it is best to determine the parameters by a method that minimizes F directly. In addition, even when F is bilinear and the matrix inversion method is applicable, the minimization by conjugate gradient is more efficient than matrix inversion. As a rule, if matrix inversion requires N^3 operations, the conjugate gradient requires N operations.[37]

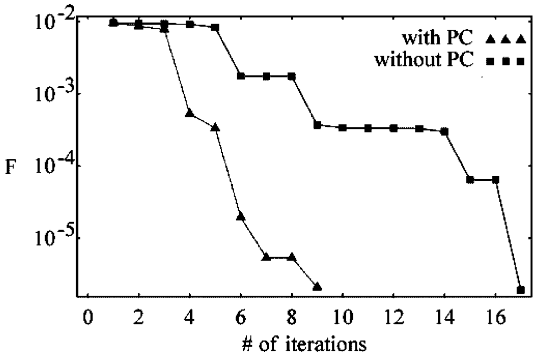

Figure 3. Evolution of the functional F (in au) as a function of the number of iterations in the conjugate-gradient minimization. The connected line with boxes is without preconditioning and the line with triangles is with preconditioning.

For these reasons, we have explored the possibility of minimizing F by the conjugate-gradient method. Given the history of numerical instabilities of KVP with an **R**-matrix asymptotic wave function, we were worried that MEM with conjugate gradient may also turn out to be unstable. We found that this is not the case: the method converges efficiently even for those energies for which KVP gives large errors. This indicates that preconditioning is not obligatory, but our calculations show that it can speed up the convergence. In Figure 3 we show a typical history of the conjugate-gradient convergence with the number of iterations. The iteration was started by assuming that the asymptotic wave function is valid in the interaction region and by taking **R** (c_0 in our notation) to be equal to 5 (the exact value at the incident wave vector $k = 0.15$ au is $-1.744\,94$). These are as unreasonable starting points as one is likely to get in any other system. We required the iterative scheme to stop when the difference between the last two iterated values of F became equal to 10^{-5}. Figure 3 shows that both searches converge and the preconditioned method converges faster. Although in this simple problem the advantage of preconditioning is not great, it turns out to be essential in more complicated problems, where it reduced the number of iterations by a factor between 7 and 20 (depending on the energy), and the magnitude of the reduction increases with the number of dimensions.

VI. Conclusion

We have shown that the least-squares method (with either matrix inversion or conjugate-gradient minimization), used with the Chebyshev polynomials as the basis set, enables one to obtain accurate phase shifts and wave functions for elastic s-wave scattering. Compared to the SKVP method, the MEM provides a better representation of the wave function in the interaction region. Because it uses real algebra, it is faster and more efficient in terms of the computational time and storage than SKVP. Even though we use an **R**-matrix asymptotic condition, we find no numerical instabilities.

Appendix A. Equations Defining $M_{\alpha\beta}$, M_{00}, $M_{0\alpha}$, N_0, N_α and D

To obtain eq 13, we insert the wave function given by eqs 11 and 12 in eq 2 for F. This gives the following equations for $M_{\alpha}\beta$, M_{00}, $M_{0\alpha}$, N_0, N_α and D.

$$M_{\alpha\beta} = \sum_{i=1}^{N} w_i[(H-E)\varphi_\alpha]_i^*[(H-E)\varphi_\beta]_i \tag{A1}$$

$$M_{00} = \sum_{i=1}^{N} w_i[(H-E)\chi_2]_i^*[(H-E)\chi_2]_i \tag{A2}$$

$$M_{0\alpha} = \sum_{i=1}^{N} w_i[(H-E)\chi_2]_i^*[(H-E)\varphi_\alpha]_i \tag{A3}$$

$$N_0 = \sum_{i=1}^{N} w_i[(H-E)\chi_1]_i^*[(H-E)\chi_2]_i \tag{A4}$$

$$N_\alpha = \sum_{i=1}^{N} w_i[(H-E)\chi_1]_i^*[(H-E)\varphi_\alpha]_i \tag{A5}$$

$$D = \sum_{i=1}^{N} w_i[(H-E)\chi_1]_i^*[(H-E)\chi_1]_i \tag{A6}$$

The expressions $[(\hat{H} - E)\varphi]_i$ mean the value of $(\hat{H} - E)\varphi$ at the grid point r_i. Because the sums above are the Gaussian-quadrature approximations for various matrix elements, we use the grid points and the weights appropriate for this procedure. These are provided by the Mathematica function GaussianQuadratureWeights.[41]

To evaluate the expressions being summed, we need to calculate the kinetic energy operator acting on the test function Ψ_t. We have performed all the derivatives analytically using the equation

$$\frac{dT_n(x)}{dx} = \frac{n[T_{n-1}(x) - T_{n+1}(x)]}{2(1-x^2)} \tag{A7}$$

For more demanding examples one should use the method explained in Appendix B.

Appendix B. Calculation of F and the Derivatives in It

Trial wave function is

$$\Psi_t(r) = f(r)\,\Psi_I(r) + (1 - f(r))\Psi_a(r) \tag{B1}$$

We are only interested in its values $\{\Psi_I(1), ..., \Psi_I(N+1)\}$ at the grid points $r_1, ..., r_{N+1}$.

The boundary conditions require

$$\Psi_I(1) = \Psi_I(N+1) = 0 \tag{B2}$$

Therefore we only need to minimize F with respect to

$$x \equiv \{\Psi_I(2), ..., \Psi_I(N)\} \quad \text{and} \quad c_0 \tag{B3}$$

The minimization algorithm starts with a guess for x and c_0 and generates better values iteratively. At each iteration we start with known (but approximate) values of x and c_0 and need to evaluate F and ∇F. Having these allows us to generate a better approximation for x and c_0.

Here we explain how we calculate F and ∇F from x and c_0: The more delicate part in this calculation is the evaluation of the derivative of Ψ_t from its values at the grid points. The finite difference method of low order is inaccurate and high order calculations are inefficient. A better method is explained below.

We expand Ψ_t as a sum of Chebyshev polynomials T_α. Because the Chebyshev polynomials $T_\alpha(y)$ are defined only for $y \in [-1, 1]$, we perform the change of variable:

$$r = \frac{a}{2}(y+1) \tag{B4}$$

The function $\Phi(y)$ defined by

$$\Psi_t(r) = \Psi_t\left(\frac{a}{2}(y+1)\right) \equiv \Phi(y) \tag{B5}$$

is expanded as a sum of Chebyshev polynomials

$$\Phi(y_i) = \frac{c_1}{2} + \sum_{\alpha=2}^{N} c_\alpha T_\alpha(y_i) + \frac{c_{N+1}}{2}T_{N+1}(y_i) \tag{B6}$$

The coefficients c_α are obtained in $N \log_2 N$ operations, from knowledge of $\Phi(y_i)$ at the grid points, by using a fast Chebyshev transform. The availability of a fast transform is the main reason for preferring the Chebyshev polynomials over other orthogonal polynomials.

The grid points y_i are given by (the extrema grid)

$$y_i = \cos\left(\frac{(i-1)\pi}{N}\right) \qquad i = 1, ..., N+1 \tag{B7}$$

We use this extrema grid (as opposed to the "root grid") of Chebyshev polynomials because it allows us to impose the boundary conditions at $r_1 = a$ and $r_{N+1} = 0$ easily.

Having chosen this grid, we must use the corresponding weights (in the expression for F)

$$w_i = \frac{a}{2}\frac{2}{Np_i}\sum_{m=1}^{N+1}\frac{1}{p_{m-1}}T_{m-1}(y_{i-1})\left[\frac{-1+\cos(m\pi)}{m(m-2)}\right] \tag{B8}$$

where p_i is 2 for the first and the last terms and is otherwise 1. These weights are computed once at the beginning of the calculation.

To evaluate the derivatives of Ψ_t with respect to r (needed for the kinetic energy), we use

$$\left[\frac{d\Psi_t(r)}{dr}\right]_i = \frac{2}{a}\left[\frac{d\Phi(y)}{dy}\right]_i \tag{B9}$$

and calculate $d\Phi(y)/dy$ from the expression

$$\left[\frac{d\Phi}{dy}\right]_i = \frac{d_1}{2} + \sum_{\alpha=2}^{N-1} d_\alpha T_\alpha(y_i) + \frac{d_N}{2}T_N(y_i) \tag{B10}$$

with d_α obtained from the recursion formula:

$$\begin{aligned} &d_{N+1} = 0 && d_N = Nc_{N+1} \\ &d_\alpha = d_{\alpha+2} + (2\alpha c_{\alpha+1}) && \alpha = N-1, ..., 1 \end{aligned} \tag{B11}$$

This process involves only N operations and it is very accurate. When we apply the recursion twice, we obtain the coefficients of the second derivatives. The evaluation of the sum in eq B10 is performed with an inverse fast Chebyshev transform.

Appendix C. Functional Derivatives

The functional F is given by

$$F = \sum_{i=1}^{N+1} w_i[(\hat{H} - E)\Psi_{\mathrm{t}}(r)]_i^2 \qquad \text{(C1)}$$

The derivative with respect to interaction wave function is

$$\frac{\partial F}{\partial \Psi_{\mathrm{I}}(m)} = 2\sum_{i=1}^{N+1}\left[\frac{\partial}{\partial \Psi_{\mathrm{I}}(m)}((\hat{H} - E)\Psi_{\mathrm{t}})_i\right] w_i[(\hat{H} - E)\Psi_{\mathrm{t}}]_i \qquad \text{(C2)}$$

The only difficulty in the above equation is the calculation of the terms involving kinetic energy. Formally, we can write

$$\left[\frac{\mathrm{d}\Psi_{\mathrm{t}}}{\mathrm{d}r}\right]_i = \left(\frac{2}{a}\right)\sum_{j=1}^{N+1} D_{ij}\Psi_{\mathrm{t}}(y_j) \qquad \text{(C3)}$$

$$\left[\frac{\mathrm{d}^2\Psi_{\mathrm{t}}}{\mathrm{d}r^2}\right]_i = \left(\frac{2}{a}\right)^2 \sum_{j=1}^{N+1}\sum_{n=1}^{N+1} D_{ij}D_{jn}\Psi_{\mathrm{t}}(y_n) \qquad \text{(C4)}$$

This matrix notation is useful for the derivation that follows. However, the whole point of the derivation is to avoid matrix multiplication and find an order N procedure of the evaluation.

We denote

$$A(i) = w_i[(\hat{H} - E)\Psi_{\mathrm{t}}(y)]_i \qquad \text{(C5)}$$

and rewrite the functional derivative formula as

$$\begin{aligned}\frac{\partial F}{\partial \Psi_{\mathrm{I}}(m)} &= 2\sum_{i=1}^{N+1}\left[\frac{\partial}{\partial \Psi_{\mathrm{I}}(m)}((\hat{H} - E)\Psi_{\mathrm{t}}(i))\right]A(i) \\ &= 2\left(\frac{2}{a}\right)^2 \sum_{i=1}^{N+1}\left[\frac{\partial}{\partial \Psi_{\mathrm{I}}(m)}\left(\sum_{j=1}^{N+1} D_{ij}D_{jn}\Psi_{\mathrm{t}}(n) + V(i)\,\Psi_{\mathrm{t}}(i) - E\Psi_{\mathrm{t}}(i)\right)\right]A(i) \\ &= 2\left(\frac{2}{a}\right)^2 \sum_{i=1}^{N+1}\left[\left(\sum_{j=1}^{N+1} D_{ij}D_{jn}\frac{\partial \Psi_{\mathrm{t}}(n)}{\partial \Psi_{\mathrm{I}}(m)}\right)A(i)\right] + \\ &\qquad \left[2\sum_{i=1}^{N+1}\left(V(i)\frac{\partial \Psi_{\mathrm{t}}(i)}{\partial \Psi_{\mathrm{I}}(m)} - E\frac{\partial \Psi_{\mathrm{t}}(i)}{\partial \Psi_{\mathrm{I}}(m)}\right)A(i)\right] \qquad \text{(C6)}\end{aligned}$$

Next we define

$$D_{in}^{(2)} = \sum_{j=1}^{N+1} D_{ij}D_{jn} \qquad \text{(C7)}$$

and make use of

$$\frac{\partial \Psi_{\mathrm{t}}(n)}{\partial \Psi_{\mathrm{I}}(m)} = \delta_{nm} \qquad \text{(C8)}$$

The functional derivative becomes

$$\frac{\partial F}{\partial \Psi_{\mathrm{I}}(m)} = \left(2\left(\frac{2}{a}\right)^2 f(r_m)\sum_{i=1}^{N+1} D_{im}^{(2)}A(i)\right) + 2f(r_m)\,V(m)\,A(m) - 2f(r_m)\,EA(m) \qquad \text{(C9)}$$

Because we do not want to perform matrix operations, we rewrite this equation as

$$\frac{\partial F}{\partial \Psi_{\mathrm{I}}(m)} = \left(2\left(\frac{2}{a}\right)^2 f(r_m)\sum_{i=1}^{N+1} D_{mi}^{(2)T}A(i)\right) + 2f(r_m)\,V(m)\,A(m) - 2f(r_m)\,EA(m) \qquad \text{(C10)}$$

To evaluate the sum, we use the following algorithm: fast Chebyshev transform $A(i)$ to calculate the coefficients a_α in

$$A(i) = a_1 + \sum_{\alpha=2}^{N} a_\alpha T_\alpha(y_i) + a_{N+1}T_{N+1}(y_i) \qquad \text{(C11)}$$

Then apply the recursion

$$\begin{aligned} b_1 &= 0 \qquad b_2 = a_1 \\ b_\alpha + 1 &= b_{\alpha-1} + 2a_\alpha \qquad \alpha = 2, ..., N \end{aligned} \qquad \text{(C12)}$$

to calculate b_α, $\alpha = 2, ..., N$.

Then perform the multiplication

$$\bar{b}_\alpha = b_\alpha(\alpha - 1) \qquad \alpha = 2, ..., N + 1 \qquad \text{(C13)}$$

to calculate $\bar{b}_\alpha$. Then use the recursion

$$\begin{aligned} c_1 &= 0 \qquad c_2 = b_1 \\ c_{\alpha+1} &= c_{\alpha-1} + 2b_\alpha \qquad \alpha = 2, ..., N \end{aligned} \qquad \text{(C14)}$$

and the multiplication

$$\bar{c}_\alpha = c_\alpha(\alpha - 1) \qquad \alpha = 2, ..., N + 1 \qquad \text{(C15)}$$

These coefficients are used to calculate the second derivative from

$$(D^{(2)T}A)_m = \frac{\bar{c}_0}{4} + \sum_{\alpha=1}^{N} \bar{c}_\alpha T_\alpha(y_m) + \frac{\bar{c}_{N+1}}{4}T_{N+1}(y_m) \qquad \text{(C16)}$$

The sum is performed with an inverse fast Chebyshev transform. Note that the first and the last elements of the summations in eqs C11 and C16 are multiplied by constants different from those of the other summations we perform throughout the calculation. The difference comes from the application of $D^{(2)T}$ operator.

The derivative of F with c_0 is

$$\frac{\partial F}{\partial c_0} = 2\sum_{i=1}^{N+1} A(i)(\hat{H} - E)[(1 - f(r_i))\cos(kr_i)] \qquad \text{(C17)}$$

Appendix D. Preconditioning in the Conjugate-Gradient Method

Experimenting with various possibilities, we have found that we only want to precondition the gradient components $\partial F/\partial \Psi_1$, ..., $\partial F/\partial \Psi_{N+1}$ and not $\partial F/\partial c_0$. For this purpose, we will drop the $\partial F/\partial c_0$ term in $\nabla F = \{0, \partial F/\partial \Psi_2, ..., \partial F/\partial \Psi_N, 0, \partial F/\partial c_0\}$ while preconditioning the gradient. After the selected components of the gradient have been preconditioned, we add back the $\partial F/\partial c_0$ component.

Preconditioning is performed through a sequence of operations, each involving either N or $N \log_2 N$ operations.

First, we evaluate the vector

$$P_1 = \left\{0, e^{r_2/8} w_2 \frac{\partial F}{\partial \Psi_2}, ..., e^{r_N/8} w_N \frac{\partial F}{\partial \Psi_N}, 0\right\} \tag{D1}$$

This creates an N+1-dimensional vector in the coordinate representation.

The points r_i are given by

$$r_i = \frac{a}{2}\left[\cos\left(\frac{(i-1)\pi}{N}\right) + 1\right] \qquad i = 1, ..., N + 1 \tag{D2}$$

These multiplications tend to diminish the values of the gradient components toward the edge of the grid. These choices, and the ones that follow, are the results of numerical experimentation. Although we had some qualitative arguments that have guided us in making them, they are not worth mentioning here.

After obtaining P_1, we perform a Chebyshev transform on it and generate an N+1-dimensional vector, in the Chebyshev space, called P_2. This is used to evaluate

$$P_3 = \left\{P_2(1), \frac{1}{1 + \frac{(2-1)^2}{N}} P_2(2), ..., \frac{1}{1 + \frac{(N-1)^2}{N}} P_2(N), \frac{P_2(N+1)}{N+1}\right\} \tag{D3}$$

We perform on this array an inverse fast Chebyshev transform, which generates the N+1-dimensional vector P_4, in the coordinate space. Its components are denoted by $\{P_4(1), P_4(2), ..., P_4(N), P_4(N+1)\}$. They are used to evaluate

$$P_5 = \{0, w_2 P_4(2), ..., w_N P_4(N), 0\} \tag{D4}$$

A fast Chebyshev transform of P_5 generates the array P_6, which is used to calculate

$$P_7 = \left\{P_6(1), \frac{1}{1 + \frac{(2-1)^2}{N}} P_6(2), ..., \frac{1}{1 + \frac{(N-1)^2}{N}} P_6(N), \frac{P_6(N+1)}{N+1}\right\} \tag{D5}$$

The inverse fast Chebyshev transform of this array generates the vector P_8 in the coordinate space. Next we calculate the preconditioned gradient

$$G = \left\{0, e^{r_2/8} w_2 P_8(2), ..., e^{r_N/8} w_N P_8(N), 0, \frac{\partial F}{\partial c_0}\right\} \tag{D6}$$

One can make a qualitative argument that the division of the Chebyshev coefficients with $[1 + (\alpha - 1)^2/N]$, $\alpha = 1, ..., N + 1$, decreases the high eigenvalues of the kinetic energy operator and therefore makes the matrix (in the quadratic form, F) less ill conditioned. The preconditioning does reduce the number of iterations in the conjugated gradient search. However, this reduction is not sufficiently dramatic to prompt us to experiment further with better preconditioners. Applications to high dimensionality, more complex problems may require a search for better preconditioners.

References and Notes

(1) Duneczky, C.; Wyatt, R. E. *J. Chem. Phys.* **1988**, *89*, 1448.

(2) Heidinger, L. E. *Surf. Sci.* **1999**, *430*, 89.

(3) Manolopoulos, D. E.; D'Mello, M.; Wyatt, R. E. *J. Chem. Phys.* **1989**, *91*, 6096.

(4) Miller, W. H.; op de Haar, B. M. D. D. *J. Chem. Phys.* **1987**, *86*, 6213.

(5) Peet, A. C.; Miller, W. H. *Chem. Phys. Lett.* **1988**, *149*, 257.

(6) Saalfrank, P.; Miller, W. H. *J. Chem. Phys.* **1993**, *98*, 9040.

(7) Schwenke, D. W.; Haug, K.; Truhlar, D. G.; Sun, Y.; Zhang, J. Z. H.; Kouri, D. J. *J. Phys. Chem.* **1987**, *91*, 6080.

(8) Staszewska, G.; Truhlar, D. G. *Chem. Phys. Lett.* **1986**, *130*, 341.

(9) Tawa, G. J.; Mielke, S. L.; Truhlar, D. G.; Schwenke, D. W. *J. Chem. Phys.* **1994**, *100*, 5751.

(10) Yang, W.; Peet, A. C.; Miller, W. H. *J. Chem. Phys.* **1989**, *91*, 7537.

(11) Zhang, J. Z. H.; Chu, S. I.; Miller, W. H. *J. Chem. Phys.* **1988**, *88*, 6233.

(12) Zhang, J. Z. H.; Miller, W. H. *Chem. Phys. Lett.* **1987**, *140*, 329.

(13) Zhang, J. Z. H.; Miller, W. H. *Chem. Phys. Lett.* **1988**, *153*, 465.

(14) Zhang, J. Z. H.; Miller, W. H. *J. Chem. Phys.* **1988**, *89*, 4454.

(15) Nesbet, R. K. *Phys. Rev.* **1968**, *175*, 134.

(16) Nesbet, R. K. *Variational Methods in Electron-Atom Scattering Theory*; Plenum: New York, 1980.

(17) Schwartz, C. *Ann. Phys.* **1961**, *16*, 36.

(18) Mott, N. F.; Massey, H. S. *The Theory of Atomic Collisons*, 3rd ed.; Clarendon Press: Oxford, U.K., 1965.

(19) Rudge, M. R. H. *J. Phys. B: At. Mol. Phys.* **1976**, *9*, 2357.

(20) Rudge, M. R. H. *J. Phys. B: At. Mol. Phys.* **1980**, *13*, 3717.

(21) Rudge, M. R. H. *J. Phys. B: At. Mol. Opt. Phys.* **1990**, *23*, 4275.

(22) Rudge, M. R. H. *J. Phys. B: At. Mol. Opt. Phys.* **2000**, *33*, 4505.

(23) Rudge, M. R. H. *J. Phys. A: Math. Gen.* **2003**, *36*, 1485.

(24) Bardsley, J. N.; Gerjuoy, E.; Sukumar, C. V. *Phys. Rev. A* **1972**, *6*, 1813.

(25) Boyd, J. P. *Chebyshev and Fourier Spectral Methods*, 2nd ed.; Dover: New York, 2001.

(26) Abdelraouf, M. A. *J. Phys. B—At. Mol. Opt. Phys.* **1979**, *12*, 3349.

(27) Abdelraouf, M. A. *Phys. Rep.—Rev. Sect. Phys. Lett.* **1982**, *84*, 163.

(28) Apagyi, B.; Ladanyi, K. *Phys. Rev. A* **1986**, *33*, 182.

(29) Levine, R. D.; Bernstei. Rb; Labudde, R. A.; Bottner, R.; Ross, U.; Toennies, J. P.; King, D. L.; Loesch, H. J.; Herschba, Dr.; Ding, A. M. G.; Polanyi, J. C.; Kendall, G. M.; Toennies, J. P.; Larsen, R. A.; Loesch, H. J.; Krenos, J. R.; Herschba, Dr.; Aquilant, V. *Faraday Discuss.* **1973**, 221.

(30) McCurdy, C. W.; Rescigno, T. N. *Phys. Rev. A* **1989**, *40*, 1297.

(31) Merts, A. L.; Collins, L. A. *J. Phys. B—At. Mol. Opt. Phys.* **1985**, *18*, L29.

(32) Read, F. H.; Soto-Montiel, J. R. *J. Phys. B—At. Mol. Opt. Phys.* **1973**, *6*, L15.

(33) Heather, R. W.; Light, J. C. *J. Chem. Phys.* **1983**, *79*, 147.

(34) Light, J. C.; Hamilton, I. P.; Lill, J. V. *J. Chem. Phys.* **1985**, *82*, 1400.

(35) Lill, J. V.; Parker, G. A.; Light, J. C. *J. Chem. Phys.* **1986**, *85*, 900.

(36) Mason, J. C.; Handscomb, D. C. *Chebyshev Polynomials*; CRC Press: Boca Raton, FL, 2003.

(37) Press: W. H.; Teukolsky, S. A.; Vetterling, W. T.; Flannery, B. P. *Numerical Recipes in Fortran 77*, 2nd ed.; Cambridge University Press: Cambridge, U.K., 2001; Vol. 1.

(38) Rivlin, T. J. *An Introduction to the Approximation of Functions*, 2nd ed.; Dover: New York, 1981.

(39) Canuto, C.; Hussaini, M.; Quarteroni, A.; Zang, T. *Spectral Methods in Fluid Dynamics*, 3rd ed.; Springer: Berlin, 1991.

(40) Hamming, R. W. *Numerical Methods for Scientists and Engineers*, 2nd ed.; Dover: New York, 1987.

(41) Wolfram. *Mathematica 3.0 Standard Add-on Packages*; Cambridge University Press: Cambridge, U.K., 1996.