Computer Assisted Mechanics and Engineering Sciences, 5: 85–92, 1998.

A fast solver for the complex symmetric eigenproblem[1]

Ilan Bar-On
Department of Chemistry, Technion Institute of Technology, Haifa 32000, Israel

Marcin Paprzycki
Department of Computer Science and Statistics, University of Southern Mississippi, Hattiesburg, MS 39406, USA

(Received October 9, 1996)

Many numerical methods for studying chemical reaction problems require the computation of the eigenvalues of very large complex symmetric matrices. Recently, a new algorithm for this problem has been proposed by Bar-On and Ryaboy [3]. This algorithm is similar in concept and complexity to the Hermitian eigensolver and is based on application of complex orthogonal transformations to preserve symmetry and recovery transformations to preserve stability. We demonstrate the performance of the proposed algorithm on several high performance computers from Digital, SGI, and Cray. The results show that the new algorithm is much faster than the general eigensolver, the present method used for solving these problems.

1. INTRODUCTION

Recent advances in studying chemical reaction problems require the calculation of the eigenvalues of large complex symmetric matrices. The complex rotation [2, 7, 9, 12, 15, 17, 19, 20] and the optical potential [8] methods are typical examples of such methods which are applied to the chemical problems [4, 10, 11, 13, 16, 18]. Modern state of the art software libraries, such as LAPACK [1], provide routines for computing the eigenvalues of complex Hermitian and complex general matrices, the former being faster nearly by an order of magnitude, but no support for the complex symmetric matrices. In case of complex symmetric matrices, the use of unitary transformations in the reduction stage (the current approach) destroys symmetry so that these matrices have to be treated as complex general. Thus, for large matrices (order 5000 and more), finding eigenvalues becomes extremely time consuming (taking up to 90% of the total time) and dominates the computational effort of the chemical reaction modelling.

Recently, a new algorithm for the complex symmetric eigenproblem, similar in concept to the Hermitian eigensolver, has been proposed in [3]. It is based on the application of complex orthogonal transformations (instead of the commonly used unitary transformations) to preserve symmetry and recovery transformations to maintain stability. As a result, the computational complexity of the new algorithm remains similar to that of the Hermitian eigensolver. In this paper, we present experimental results that demonstrate the performance of the new algorithm on several state of the art computers: the DEC Alpha server and workstation, the SGI Power Challenge 8000, and the Cray J-916. As can be seen, the new algorithm is much faster than the general eigensolver, the current method of choice for solving the complex symmetric eigenproblem.

The paper is organized as follows: we begin with a comparison of the general and Hermitian eigensolvers on the Cray, in Section 2. Then we review the characteristics of the new algorithm and

[1]We thank The Inter University Computation Center (IUCC) of the Israel Science Ministry and NCSA in Urbana-Champaign for computer time.

its performance on the Cray, in Section 3. Finally, we compare these algorithms on the superscalar machines, in Section 4.

2. General and Hermitian eigensolvers

The standard method for computing the eigenvalues of general complex matrices proceeds as follows [6, 21] :

Householder Reduction We reduce the dense matrix $A \in \mathcal{C}^n$ into an upper Hessenberg matrix $H \in \mathcal{C}^n$, by a sequence of unitary transformations. As these transformations preserve the spectral structure of the transformed matrices. Thus, the final Hessenberg matrix has the same set of eigenvalues as the original matrix.

Hessenberg QR We transform the Hessenberg matrix $H \in \mathcal{C}^n$ into an upper triangular matrix $U \in \mathcal{C}^n$, again by unitary transformations preserving the spectral structure of the original matrix. Thus, the eigenvalues are found on the main diagonal of the triangular matrix.

The reduction process is deterministic and requires only $\approx \frac{40}{3}n^3$ flops, see [6, page 366]. However, the QR process is iterative and convergence depends on the specific spectral structure of the underlying matrix. Hence, assuming on average $k \geq 2$ iterations per eigenvalue[2], the number of flops is of order $\approx \frac{28k}{3}n^3$ (ibid. page 364), for a total of $\approx \frac{28k+40}{3}n^3$ flops. Thus the algorithm becomes very time consuming as is evident from Table 1.

We illustrate the performance of the LAPACK routines ZGEHRD for the Reduction, and ZHSEQR for the QR step on the Cray J-916. We report the average CPU times, in seconds, and the average performance, in Mflops, for random general complex matrices, collected using the Cray's *perftrace* utility. As can be observed, the running time becomes significant even for matrices of order 1800. The QR step dominates the solution process and is characterized by the lower performance.

Table 1. The general eigensolver

	Reduction		QR	
$n =$	Seconds	Mflops	Seconds	Mflops
200	0.6	178	3.3	65
600	15	190	58	98
1000	69	193	219	106
1400	189	194	572	110
1800	401	194	1214	113

Table 2. The Hermitian eigensolver

	Reduction		QR	
$n =$	Seconds	Mflops	Seconds	Mflops
200	0.3	146	0.1	21
600	6.5	178	0.8	20
1000	29	185	2.2	20
1400	78	188	4.0	21
1800	164	190	7.1	21

We next consider the Hermitian eigensolver which can be treated as an application of the general eigensolver to the complex Hermitian[3] matrices. Here, the unitary transformations preserve the original Hermitian structure of the transformed matrices so that the final Hessenberg matrix, being Hermitian, is also necessarily tridiagonal. We gain considerably in the number of flops which now becomes only $\approx \frac{16}{3}n^3$. The QR process, that previously dominated the running time, now takes only $O(n^2)$ flops (see Table 2).

In Table 2 we present the performances of the LAPACK routines, ZHETRD for the Reduction, and ZSTEQR for the QR step, on the Cray J-916. As before we report the average CPU times and the performance, for random complex Hermitian matrices, collected by the Cray's *perftrace* utility. Observe that, indeed, the Reduction process runs ≈ 2.5 times faster than before. As predicted above the QR step, although running at a much lower Mflop rate, is computationally insignificant.

[2] two iterations are assumed in [6, page 366].

[3] but non symmetric.

Table 3. General vs. Hermitian

	Running Times			Mflops/195	
$n=$	Gen	Her	Spu	Gen	Her
200	3.9	0.4	9.8	42%	58%
600	73	7.3	10	60%	82%
1000	288	31	9.3	65%	89%
1400	761	82	9.3	67%	92%
1800	1615	171	9.4	68%	94%

In Table 3, we compare the overall performance of the complex general and the complex Hermitian eigensolvers (as reported in Tables 1 and 2). We also present the average performance in percentage of the practical peak performance of the Cray J-916, which is 195 Mflops [14]. As can be seen, the more efficient implementation of the Hermitian eigensolver yields a speedup of ≈ 10 over the general eigensolver.

3. THE COMPLEX SYMMETRIC EIGENSOLVER

Consider the standard Hermitian process applied to dense matrices $A=A^{(n+1)}$, i.e.,

$$A^{(k)} = Q_k^* A^{(k+1)} Q_k, \qquad k = n, \ldots, 3, \qquad Q_k^* Q_k = I$$

with $T=A^{(3)}$ being tridiagonal Hermitian. Then, at the k^{th} step

$$A^{(k+1)} = \begin{pmatrix} A_k^{(k)} & \beta_{k+1} & & \\ \overline{\beta}_{k+1} & \alpha_{k+1} & \ddots & \\ & \ddots & \ddots & \beta_n \\ & & \overline{\beta}_n & \alpha_n \end{pmatrix}, \qquad A_k^{(k)} = \begin{pmatrix} A_{k-1}^{(k)} & b \\ b^* & \alpha_k \end{pmatrix}, \qquad Q_k = \begin{pmatrix} \mathcal{Q}_{k-1} & \\ & I_{n-k+1} \end{pmatrix}, \tag{1}$$

$$\mathcal{Q}_{k-1} = I - 2\frac{v\,v^*}{v^*v}, \qquad v = b + \frac{b_{k-1}}{(b_{k-1}^* b_{k-1})^{\frac{1}{2}}} (b^*b)^{\frac{1}{2}} e_{k-1}, \tag{2}$$

where e_{k-1} is the $(k-1)^{\text{th}}$ standard unit vector. Hence, $A_{k-1}^{(k-1)} = \mathcal{Q}_{k-1}^* A_{k-1}^{(k)} \mathcal{Q}_{k-1}$ and

$$\beta_k = -\frac{b_{k-1}}{|b_{k-1}|} \|b\|_2 \ .$$

For simplicity, let A and A' denote $A_{k-1}^{(k)}$ and $A_{k-1}^{(k-1)}$ respectively. Then,

$$\begin{aligned} A' &= A - 2w(w^*A) - 2(Aw)w^* + 4(w^*Aw)ww^* & \gamma &= \|u'\|_2, u' = Aw \\ &= A - 2\gamma(wu^* + uw^* - 2(w^*u)ww^*) & u &= \frac{1}{\gamma}u' \\ &= A - 2\gamma(wr^* + rw^*), & r &= u - (w^*u)w, \end{aligned}$$

and the complexity of the process is dominated by one matrix vector product, one outer vector product, and one matrix subtraction for a total of $\approx 16k^2$ flops. Let us try to modify this process in order to apply it to the complex symmetric matrices. A simple solution is to replace the conjugate transpose operation $(\cdot)^*$, with the transpose operation $(\cdot)^T$. For example, let

$$A = \begin{pmatrix} 0 & 3 & 4 & 5i \\ 3 & 0 & 5i & 4 \\ 4 & 5i & 0 & 3 \\ 5i & 4 & 3 & 0 \end{pmatrix}, \qquad X = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}, \qquad \Lambda = \begin{pmatrix} 7+5i & & & \\ & 1-5i & & \\ & & -1-5i & \\ & & & -7+5i \end{pmatrix},$$

with $AX = X\Lambda$. Then following (2) we have to use

$$Q = I - 2\frac{v\,v^t}{v^t v}, \qquad v = b \pm (b^t b)^{1/2} e_3, \qquad b = (5i, 4, 3)^T,$$

which is now undefined since $v^T v = 0$. Note that such breakdowns are not characteristic to the ill conditioned problems as the spectrum of the matrices above is very simple. Moreover, it is possible that matrices Q have a relatively large norm or that the cumulative effect of applying these matrices lead to a large element growth in the transformed matrices. In this case the rounding errors may make the algorithm practically unstable.

We therefore suggest the following more stable implementation: let $H = A + iB$ and denote the k^{th} leading submatrix as in (1), and let

$$A = \begin{pmatrix} A' & a \\ a^t & \alpha \end{pmatrix}, \qquad B = \begin{pmatrix} B' & b \\ b^t & \beta \end{pmatrix}, \qquad \begin{aligned} Q_I &= I - 2ww^t, \qquad w = \frac{v}{\|v\|_2}, \\ v &= b + \operatorname{sign}(b_{k-1})\,\|b\|_2\, e_{k-1}. \end{aligned} \tag{3}$$

We then apply

$$Q_I = \begin{pmatrix} Q_I & \\ & 1 \end{pmatrix}$$

to H and denote the transformed matrix by $H_I = A_I + iB_I$,

$$A_I = \begin{pmatrix} A_I' & \overline{a} \\ \overline{a}^t & \alpha \end{pmatrix} = \begin{pmatrix} A_I'' & \tilde{a} & \overline{a'} \\ \tilde{a}^t & \alpha' & x \\ \overline{a'}^t & x & \alpha \end{pmatrix}, \qquad B_I = \begin{pmatrix} B_I' & y \\ y & \beta \end{pmatrix} = \begin{pmatrix} B_I'' & \tilde{b} & \\ \tilde{b}^t & \beta' & y \\ & y & \beta \end{pmatrix},$$

with

$$\begin{aligned} A_I' &= Q_I^T A' Q_I, & \overline{a} &= Q_I^T a, \\ B_I' &= Q_I^T B' Q_I, & y &= \pm\,\|b\|_2\,. \end{aligned}$$

Let

$$Q_R = I - 2ww^t, \qquad w = \frac{v}{\|v\|_2}, \qquad v = \overline{a}' + \operatorname{sign}(\overline{a}'_{k-2})\,\|\overline{a}'\|_2\, e_{k-2},$$

and apply

$$Q_R = \begin{pmatrix} \overline{Q}_R & & \\ & 1 & \\ & & 1 \end{pmatrix}$$

to H_I. Then, the transformed matrix $H_R = A_R + iB_R$ satisfies

$$A_R = \begin{pmatrix} A_R'' & \hat{a} & z \\ \hat{a}^t & \alpha' & x \\ z & x & \alpha \end{pmatrix}, \qquad B_R = \begin{pmatrix} B_R'' & \hat{b} & \\ \hat{b}^t & \beta' & y \\ & y & \beta \end{pmatrix},$$

$$\begin{aligned} A_R'' &= Q_R^T A_I'' Q_R, & \hat{a} &= Q_R^T \tilde{a}, & z &= \pm\,\|\overline{a'}\|_2, \\ B_R'' &= Q_R^T B_I'' Q_R, & \hat{b} &= Q_R^T \tilde{b}. \end{aligned}$$

Finally, let

$$Q_C = \frac{1}{s_k}\begin{pmatrix} x + iy & z \\ -z & x + iy \end{pmatrix}, \qquad s_k^2 = (x + iy)^2 + z^2, \qquad Q_C^T Q_C = I.$$

Then, unless

$$\frac{|s_k|}{\max(|x|,|y|,|z|)}$$

is relatively small, we apply

$$Q_C = \begin{pmatrix} I_{k-3} & & \\ & Q_C & \\ & & 1 \end{pmatrix}$$

to H_R. Hence, $H' = Q^T H Q$ with $Q = Q_I Q_R Q_C$, $Q^T Q = I$. Here, Q_I and Q_R being real orthogonal, are norm preserving. Moreover, Q_C applies only to the last two rows and columns of H, and as is shown in [3], does not substantially increase the norms of the transformed matrices. For the special case of $x \approx 0$ and $|y| \approx |z|$, we apply simple recovery transformations that eliminate this breakdown effect. For example, when $\beta_{k+1} \neq 0$ we apply one step of the QR algorithm to the trailing tridiagonal matrix in (1), whereby the algorithm would return to normal behaviour as usual[4]. Here, the recovery transformation take $O(k)$ flops, and since their occurrences is relatively rare

$$H^{(t+1)} = (Q^{(t)})^T H^{(t)} Q^{(t)}, \qquad t = 0, \ldots, m-1, \qquad m = O(n),$$

the algorithm takes $\approx \frac{16}{3} m^3 \approx \frac{16}{3} n^3$ flops. In practice, due to the rounding errors,

$$\hat{H}^{(t+1)} = (\hat{Q}^{(t)})^T \hat{H}^{(t)} \hat{Q}^{(t)} + E^{(t)}, \qquad |E^{(t)}| \leq \epsilon |\hat{H}^{(t)}|,$$

with ϵ being the machine precision, so that

$$\hat{T} = V^T (H + E) V, \qquad E = \sum_{t=1}^{m} V^{(t)} E^{(t)} (V^{(t)})^T = \sum_{t=1}^{m} \hat{E}^{(t)}, \qquad V^{(t)} = \hat{Q}^{(1)} \cdots \hat{Q}^{(t)}.$$

By construction, the norms of $E^{(t)}$ are relatively small, and the special structure of the transformations in $V^{(t)}$ suggests that they should not increase that norm substantially. Hence, we would expect the eigenvalues to be backward stable. This has been confirmed in practice for the quantum application discussed in the introduction, i.e. the observed accuracy of the computed eigenvalues is similar to that for the general eigensolver.

We conclude with a full description of the complex symmetric eigensolver:

Orthogonal Reduction We reduce the complex symmetric matrix $A \in \mathcal{C}^n$ into a complex symmetric tridiagonal matrix $T \in \mathcal{C}^n$, by a sequence of complex orthogonal transformations. These transformations preserve the spectral structure of the transformed matrices so that the final tridiagonal matrix has the same set of eigenvalues as the original matrix.

Tridiagonal QR We transform the tridiagonal matrix into a diagonal matrix $\Lambda \in \mathcal{C}^n$, again by a sequence of complex orthogonal transformations preserving the spectral structure of the original matrix. Hence, the eigenvalues are the diagonal entries $\lambda_i, i = 1, \ldots, n$.

The Reduction process, as described above, requires $\approx \frac{16}{3} n^3$ flops, and the QR process, using the routine CMTQL1 [5] only $O(n^2)$ flops. In Table 4, we present the average performance of the algorithm on random complex symmetric matrices, which are representative of the quantum applications discussed in the introduction. Random matrices were used for the ease of the work since the latter require many more runs and their dimension could not be easily determined beforehand. As before, the experimental data was collected on the Cray J-916 using the *perftrace* utility. As expected, the performance of the Reduction process behaves as in Table 2. However, the QR process is considerably more time consuming although its significance decreases for larger dimensions[5]. This effect, we believe, can be partially traced to the inefficient implementation of the routine CMTQL1 which was originally designed for sequential scalar machines and not for vector machines such as the Cray J-916. More research should be done in order to improve this performance on the Cray.

[4] For the more simple case of $\beta_{k+1} = 0$ see [3].

[5] The inherent $O(n^2)$ complexity of this process comes into play.

Table 4. The complex symmetric eigensolver

	Reduction		QR	
$n =$	Seconds	Mflops	Seconds	Mflops
200	0.4	118	1.5	15
600	7.1	165	13	15
1000	31	173	35	15
1400	83	177	69	15
1800	174	180	114	15

4. Comparisons with superscalar machines

In addition to the Cray J-916 vector-computer we have studied the performance characteristics of all three algorithms on the following platforms: the DEC Alpha Workstation 600 5/266 (DEC1), the DEC Alpha Server 8400 (DEC2), and the Silicon Graphics Power Challenge 8000 (SGI). Their practical peak performance, as established by matrix multiplication [14], was 185 Mflops for the Alpha workstation, 220 Mflops for the Alpha server and 290 Mflops for the SGI. As the algorithms we have studied are computationally intensive ($O(n^3)$ flops), for large matrices, the performance achieved should be close to the practical peak. However, the results obtained were surprisingly different.

In Table 5, we illustrate the performance (time in seconds) of the general algorithm on the three machines. Observe that the Cray computer is significantly faster (see Table 1) in the Reduction step, although its theoretical peak performance is the second lowest. Furthermore, the SGI (a RISC based superscalar machine similar to the DEC computers, but theoretically almost twice as fast) is almost twice slower in the QR step.

In Table 6, we present the performance (time in seconds) of the Hermitian eigensolver on these machines. The Cray computer, is again the fastest in the Reduction step (see Table 2), while the superscalar machines behave similarly. The running time of the QR step is insignificant for all superscalar machines. Note that the Hermitian code seems to be well optimized on all machines, possibly because of its extensive use in many applications.

Table 5. The general eigensolver

	Reduction				QR			
$n =$	DEC1	CRAY	DEC2	SGI	DEC1	CRAY	DEC2	SGI
200	0.8	0.6	0.7	0.4	1.6	3.3	1.6	3.5
600	33	15	27	16	43	58	43	78
1000	186	69	180	183	226	219	226	369
1400	713	189	754	649	644	572	644	1116
1800	1518	401	1621	1278	1411	1214	1199	2429

Table 6. The Hermitian eigensolver

	Reduction				QR			
$n =$	DEC1	CRAY	DEC2	SGI	DEC1	CRAY	DEC2	SGI
200	0.3	0.3	0.3	0.2	0.1	0.1	0.1	0.1
600	12.5	6.5	8.2	4	0.1	0.8	0.2	0.4
1000	63	29	53	42	0.7	2.2	0.6	1.2
1400	174	78	164	163	1.5	4.0	1.3	2.4
1800	376	164	378	404	2.3	7.1	2.0	3.8

Table 7. The complex symmetric eigensolver

	Reduction				QR			
$n =$	DEC1	CRAY	DEC2	SGI	DEC1	CRAY	DEC2	SGI
200	0.4	0.4	0.4	0.8	0.4	1.5	0.3	0.5
600	14	7.1	12	20	3.1	13	2.8	5.1
1000	91	31	80	105	8.6	35	7.7	14
1400	282	83	259	327	17	69	15	28
1800	633	174	623	751	28	114	25	45

Table 8. Speedup for $n = 1800$

	Reduction		Total	
Machine	Hermitian	New	Hermitian	New
DEC1	4	2.4	7.7	4.4
CRAY	2.4	2.3	9.4	5.6
DEC2	4.3	2.6	7.4	4.4
SGI	3.2	1.7	9.1	4.7

In Table 7 we present the corresponding results for the new algorithm. Once again, in the Reduction step, Cray outperforms other machines (compare with Table 4). The performance of the superscalar machines in the Reduction step of the new algorithm is worse than their performance in the Reduction step of the corresponding Hermitian eigensolver, suggesting that further research should be done in order to improve this performance. As indicated before the poor performance of the Cray in the QR step can be partly attributed to the scalar characteristics of the routine CMTQL1.

Finally, in Table 8, we compare the speedup of the Hermitian and the new complex symmetric eigensolvers with respect to the general eigensolver. We present the speedup of the Reduction step separately, and the speedup of the complete algorithm. We observe that the new solver is approximately 5 times faster than the general eigensolver, while the results presented above may indicate that further improvements are still possible.

5. Conclusion and future research

We have presented a number of experimental results comparing the performance of the new complex symmetric eigensolver with that of the standard method applied to this problem in the current computational practice. The results clearly indicate the superiority of the new algorithm. However, as we have pointed out, more research should be carried out in order to enhance the performance of the Reduction process on the super scalar machines, and the tridiagonal QR process on the Cray vector-machine. Moreover, the numerical stability of the proposed algorithm requires further research, and more specifically, the effect of using complex orthogonal but non unitary transformations in the Reduction process should be further studied, as well as the accuracy and efficiency of the QR algorithm for complex symmetric tridiagonal matrices. Further research should also be directed toward an efficient parallel implementation of the algorithm. We hope to consider these issues in the very near future.

6. Acknowledgement

We would like to thank the anonymous referee for the help in preparing the final version of the paper.

REFERENCES

[1] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J.D. Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov, D. Sorensen. *LAPACK Users' Guide*, SIAM, 1992.

[2] E. Balslev, J. Combes. Spectral properties of many body Schroedinger operators with dilation analytic interactions. *Commun. Math. Phys.*, **22**: 280–294, 1971.

[3] I. Bar-On, V. Ryaboy. Fast diagonalization of large and dense complex symmetric matrices, with applications to quantum reaction dynamics. *SIAM J. on Scientific Computing*, **18**: 1412–1435, 1997.

[4] D. Brown, J. Light. Evaluation of thermal rate constants in the eigenbasis of a Hamiltonian with an optical potential. *J. Chem. Phys.*, **97**: 5465–5471, 1992.

[5] J.K. Cullum, R.A. Willoughby. *Lanczos Algorithms for Large Symmetric Eigenvalues Computations.* Birkhauser, Boston, 1985.

[6] G.H. Golub, C.F.V. Loan. *Matrix Computations.* The Johns Hopkins University Press, 1989.

[7] Y. Ho. The method of complex coordinate rotation and its applications to atomic collision processes. *Phys. Rep. C*, **99**: 1–68, 1983.

[8] G. Jolicard, E. Austin. Optical potential stabilization method for predicting resonance levels. *Chem. Phys. Lett.*, **121**: 106–110, 1985.

[9] B. Junker. Recent computational developments in the use of complex scaling in resonance phenomena. *Adv. At. Mol. Phys.*, **18**: 207–263, (1982).

[10] D.N.I. Last, M. Baer. The application of negative imaginary arrangement decoupling potential to reactive scattering: conversion of a reactive scattering problem into a bound-type problem. *J. Chem. Phys.*, **96**: 2017–2024, 1992.

[11] N. Lipkin, N. Moiseyev, C. Leforestier. A three dimensional study of NeICl predissociation resonances by the complex scaled discrete variable representation method. *J. Chem. Phys.*, **98**: 1888–1901, 1993.

[12] N. Moiseyev. Resonances, cross sections and partial widths by the complex coordinate method. *Isr. J. Chem.*, **31**: 311–322, 1991.

[13] D. Neuhauser, M. Baer. The time-dependent Schroedinger equation: application of absorbing boundary condition. *J. Chem. Phys.*, **90**: 4351–4355, 1989.

[14] M. Paprzycki, C. Cyphers. Multiplying matrices on the Cray — Practical considerations. *CHPC Newsletter*, 1991.

[15] W. Reinhardt. Complex coordinates in the theory of atomic and molecular structure and dynamics. *Annu. Rev.. Phys. Chem.*, **33**: 223–255, 1982.

[16] V. Ryaboy, N. Moiseyev. Cumulative reaction probability from Siegert eigenvalues: model studies. *J. Chem. Phys.*, **98**: 9618–9623, 1993.

[17] V. Ryaboy, N. Moiseyev. Three dimensional study of predissociation resonances by the complex scaled discrete variable representation method: HCO/DCO. *J. Chem. Phys.*, **103**: 4061–4067 1995.

[18] T. Seideman, W. Miller. Calculation of the cumulative reaction probability via a discrete variable representation with absorbing boundary condition. *J. Chem. Phys.*, **96**: 4412–4422, 1992.

[19] J. Simon. Quadratic form techniques and the Balslev-Combes theorem. *Commun. Math. Phys.*, **27**: 1–9, 1972.

[20] J. Simon. Resonances in n-body quantum systems with dilation analytic potentials and the foundations of time dependent perturbation theory *Ann. Math.*, **97**: 247–274, 1973.

[21] J.H. Wilkinson. *The Algebraic Eigenvalue Problem.* Oxford University Press, 1965; reprinted in Oxford Science Publications, 1988.