

## COMPARISON OF PARTIAL DIAGONALIZATION OF MATRICES WITH STANDARD PROGRAM COMPLEXES

P. Bogdanovich<sup>1\*</sup>, A. Štikonas<sup>2</sup> and O. Rancova<sup>1</sup>

<sup>1</sup>Institute of Theoretical Physics and Astronomy, Vilnius University, Goštauto 12, LT-01108, Vilnius, Lithuania

<sup>2</sup>Physics department, Vilnius University, Saulėtekio 9, LT-10222, Vilnius, Lithuania

\*E-mail: Pavelas.Bogdanovicus@tfai.vu.lt

**Abstract.** Effectiveness of the method of partial diagonalization of matrices created in Department of Theory of Atom at Institute of Theoretical Physics and Astronomy is compared with efficiency of application of corresponding programs from LAPACK and PRIMME libraries.

### 1. Introduction

Diagonalization of matrices is one of the most popular mathematical methods met in various fields of theoretical physics. This method is very important in the atomic theory as well if one finds the energy values of atomic states and their characterizing wave-functions. In the atomic theory the task comes to diagonalization of the real symmetric energy operator matrix. In general, the larger is the order  $M$  of the matrix that can be diagonalized with a computer available, the higher is a precision of the results that are obtained.

In the beginning of nineties in Department of Theory of Atom the well-known program complex [1] was started to exploit widely. The program for calculation and diagonalization of the energy operator matrices [2] is included in the complex. At that calculations were performed by rather imperfect personal computers at those days. Experience accumulated while employing this program revealed that the diagonalization method applied in [2] is machine-time demanding and does not take the peculiarities of the exploited configuration interaction method into account in full. The fact is that the orders of the diagonalized matrices  $M$  are rather high but the number of physically meaningful eigen-values and eigen-vectors  $N$  is quite small and amounts to some units usually. In all real calculations  $N \ll M$ . In order to exploit this feature effectively the method of partial diagonalization of matrices and the corresponding computer programs DIAGAT have been created in Department of Theory of Atom. This method was first described in [3]. The short description can also be found in [4]. In paper [3] the program DIAGAT was compared with effectiveness of the program [2]. In accordance with the possibilities of personal computers at those times the orders of the diagonalized matrices were not high. No other consecutive investigations of the partial diagonalization method have been performed. At the same time experience in application of the created method revealed some peculiarities of it not known earlier which forced to improve the algorithm. During recent decade the capabilities of exploitable personal computers increased significantly and GRID-technologies became available for users. That enabled one to perform investigations of rather complex problems in atomic physics which require diagonalizing matrices of a very high order. In this connection there is a need of a new analysis of possibilities of the exploitable method and its comparison with potential of programs from available libraries.

In the next section the method of partial diagonalization of matrices is briefly described. In third section there are results of comparison with diagonalization performed by programs from LAPACK and PRIMME libraries. Main inferences are presented in the conclusion.

## 2. Description of the method of partial diagonalization of matrices

The idea of partial diagonalization is based on the Jacobi elementary rotations method. Here during one iteration step two rows and their symmetric columns are changed and off-diagonal matrix elements (OME) at their intersection become zero. At the same time the eigenfunction coefficients are recalculated. When standard full diagonalization of a matrix is executed by Jacobi method the large OMEs are zeroed sequentially until the largest one becomes smaller than certain preassigned value of a small parameter  $\varepsilon$ . In the case of partial diagonalization by DIAGAT program such elimination of OME is performed only for rows and columns which diagonal matrix elements have physical meaning and are specified when calling a program. There is no requirement for the sought energies to be extreme. Thus both greater and smaller values than the sought ones can exist among the eigen-values without physical meaning. When all OME related to the sought energy values become smaller than  $\varepsilon$ , diagonalization can be stopped since further diagonalization process does not influence the obtained eigen-values and eigen-functions. This algorithm allows one to lessen the number of iterations by a few orders and to reduce essentially machine time costs.

Practical application of the algorithm described revealed some features caused by the properties of the diagonalized energy operator  $H$  matrices. Though the matrices are not completely filled and OMEs are not large in general, in some cases

$$\langle i | H | j \rangle \gg \langle i | H | i \rangle - \langle j | H | j \rangle. \quad (2.1)$$

Presence of such OME can slow down the iterative process significantly if the element does not belong to the number of eliminable ones. In this case the diagonalization algorithm provides for nullifying of the largest OME of the whole matrix. It fastens the diagonalization immediately. Moreover, existence of large OME can cause the situation when sought eigenvalue previously related to the diagonal matrix element  $\langle k | H | k \rangle$  gains another position in the matrix. It can be easily obtained examining the eigen-functions. In such case the set of sought energy values is increased and the diagonalization process is corrected in such a way that the sums of squares of expansion coefficients of eigen-functions correspondent to the sought eigen-values over the states with physical meanings become near one. The properties of the atom energy operator matrix allow one to do it without any problems usually. If the employed algorithm is not able to solve problems arose it means that the list of sought eigen-values is set incorrectly and needs to be extended.

Since the matrix is symmetric the lower triangle of the matrix is calculated and diagonalized. The matrix itself is set and the eigen-values are obtained in REAL\*8 format. At the same time the expansion coefficients of eigen-functions are obtained in REAL\*4 format. Such precision of eigen-functions is sufficient for atomic tasks and it allows one both to reduce the calculation time to some degree and to lessen the necessary size of a random-access memory. Here the size of memory used for the coefficients is slightly smaller than the size of memory needed to store the matrix itself. In order to make maximum use of the available random-access memory the method of partial diagonalization was supplemented with the separate diagonalization in DIAGAT-SEP program [4]. The idea of the method is based on the following procedure. First the eigen-values are obtained and four parameters of each elementary rotation (OME indices, sin and cos of rotation angle) are recorded sequentially to a file on a hard disk. After all eigen-values are found the energy matrix is replaced by the coefficients matrix in random-access memory and the eigen-functions are obtained using the information on elementary rotations. The separate diagonalization allows

one to increase the order of diagonalized matrices by 42% approximately. Experience in application of this method revealed though the procedure is prolonged due to recording of information to and reading it from a hard disk the total diagonalization time is not extended significantly. The data on calculation time presented in the following section is obtained without the separate diagonalization.

### 3. Comparison of effectiveness of DIAGAT program with other computer programs

The following programs have been selected for comparison: DSPEVX program (Double Symmetric Packed Eigen Vector) from LAPACK library [5] and JDQMR program from PRIMME library [6] which uses least time out of 14 programs in this library while searching for a large number of eigen-values. DSPEVX program exploits Householder transformation to reduce the symmetric matrix to tridiagonal form. Further the eigen-values are obtained. JDQMR program utilizes Jakobi-Davidson method. In all cases the programs used the same formats for calculations as DIAGAT program does and the same precision of diagonalization  $\varepsilon$  has been set. The same compiler with the identical optimization parameter has been employed for compilation.

The size of random-access memory  $L$  exploited for diagonalization is a rather important characteristic of the diagonalization program when personal computers are used. The programs under investigation have following properties (evaluated in 8-byte words):

$$\begin{aligned} L_{\text{DIAGAT}} &= M^2 + 2.5M; \\ L_{\text{DIAGAT-SEP}} &= 0.5M^2 + 0.5M; \\ L_{\text{DSPVEX}} &= M^2 + 16M; \\ L_{\text{JDQMR}} &= 0.5M^2 + 5.5M. \end{aligned} \quad (3.1)$$

As seen from (3.1) DSPEVX program requires memory above all. DIAGAT program needs slightly less of it. JDQMR program is much more perfect in this regard, however it is little inferior to DIAGAT-SEP program.

Employment of GRID-technologies facilitates significantly the problem of available random-access memory. Thus the diagonalization time  $\tau$  issue appears in the foreground. In order to test this program characteristic the model matrices have been diagonalized. The elements of these matrices have been set by simple principles:

$$A_{i,j} = \begin{cases} i, & \text{if } i = j; \\ 0.1, & \text{if } i \neq j. \end{cases} \quad (3.2)$$

$$B_{i,j} = \begin{cases} i, & \text{if } i = j; \\ 0.03, & \text{if } i \neq j. \end{cases} \quad (3.3)$$

$$C_{i,j} = \begin{cases} i, & \text{if } i = j; \\ 0.3, & \text{if } i \neq j, i+j \text{ even}; \\ 0., & \text{if } i \neq j, i+j \text{ odd}. \end{cases} \quad (3.4)$$

None of the programs exhibited significant deviations in computational time depending on a type of a model matrix. Therefore all the results presented below were obtained using (3.2) type matrix.

The matrices of various orders  $M$  were diagonalized obtaining different numbers of eigen-values  $N$  and the central processor time  $\tau$  was registered in order to compare their effectiveness. In figure 1 the dependence of the diagonalization time on the order of a matrix is shown when five eigen-values are obtained. As seen in this case DSPEVX requires significantly more time than JDQMR and DIAGAT do. At that JDQMR program is little slower than DIAGAT program.

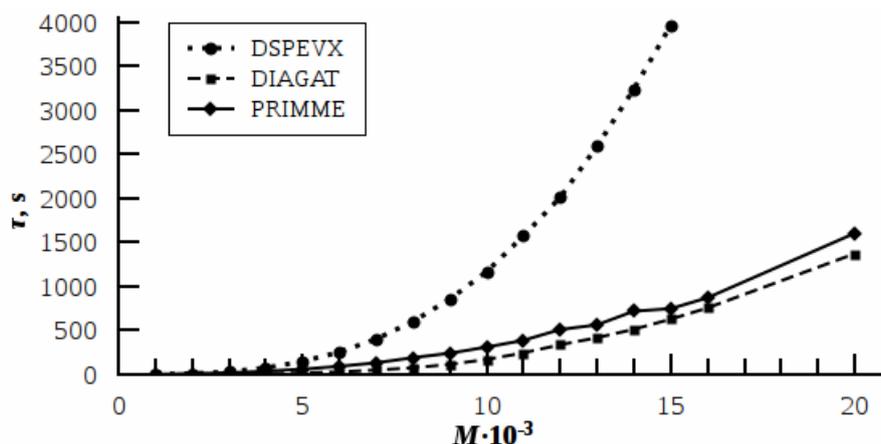


Fig 1. Diagonalization time  $\tau$  dependence on the matrix order  $M$  when  $N = 5$ .

In figures 2 and 3 the results for  $L = 10$  and  $L = 15$  are presented. For convenience of representation two time-scales are used in the figures: one for  $M \leq 8000$  matrices and another for  $M > 8000$  ones. As seen from presented figures DSPEVX program always requires significantly more time than other. At the same time DIAGAT and JDQMR programs compete with each other. DIAGAT program is leading for small values of  $M$  and JDQMR program performs diagonalization faster when orders of matrices reach 10000 and more.

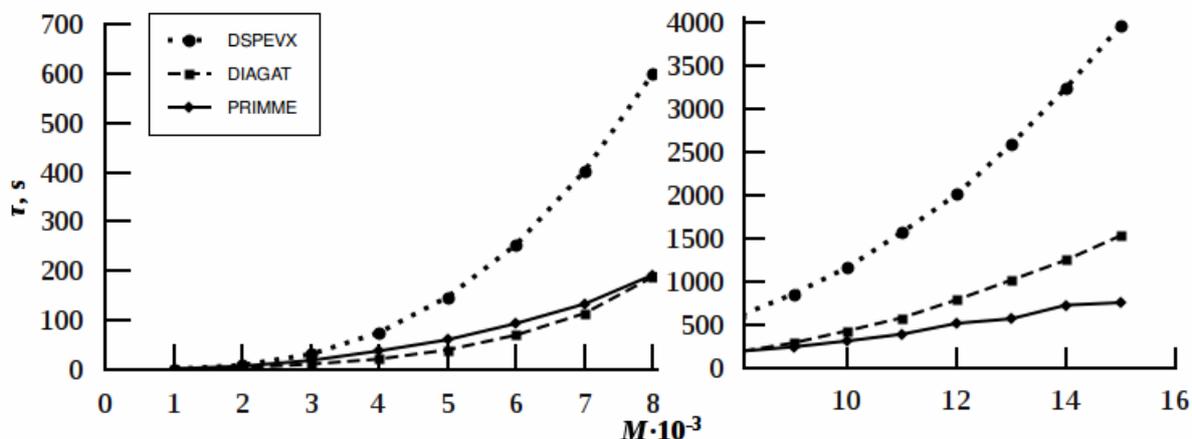


Fig 2. Diagonalization time  $\tau$  dependence on the matrix order  $M$  when  $N = 10$ .

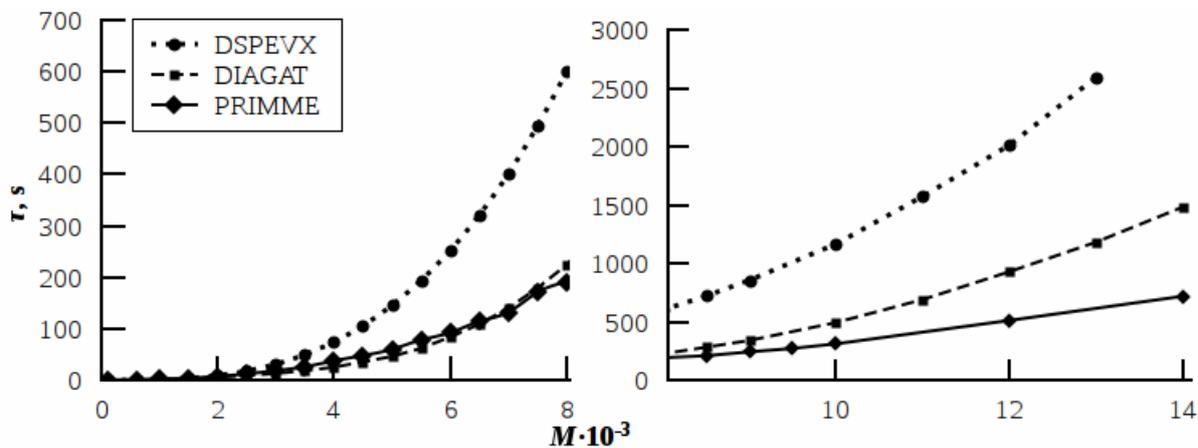


Fig 3. Diagonalization time  $\tau$  dependence on the matrix order  $M$  when  $N = 15$ .

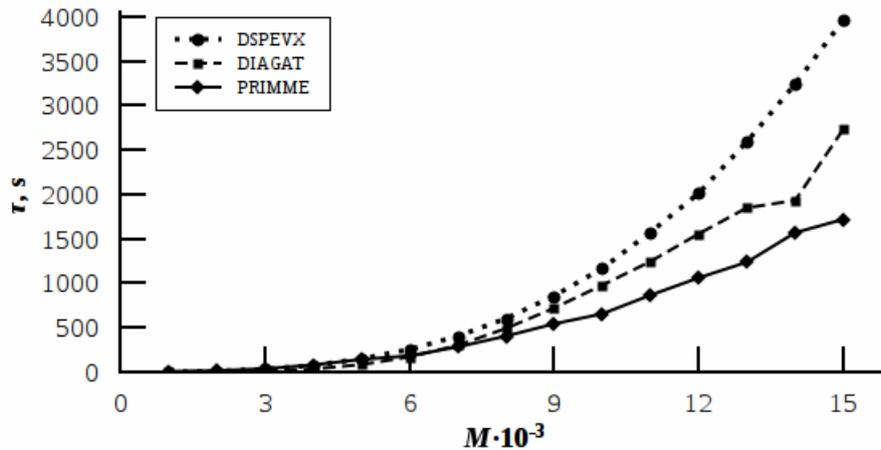


Fig 4. Diagonalization time  $\tau$  dependence on the matrix order  $M$  when  $N = 20$ .

It is seen from figure 4 that the same dependence of the central processor time on the order of a matrix exists for  $L = 20$  as well. The diagonalization time depends on the order of a matrix in a cubic form in all cases investigated. The dependence of the central processor time  $\tau$  on the number of the obtained eigen-values and eigen-functions for the fixed order of a matrix is of interest as well. This data is presented in figure 5 for  $M = 5000$ , in figure 6 for  $M = 10000$ , and in figure 7 for  $M = 20000$ .

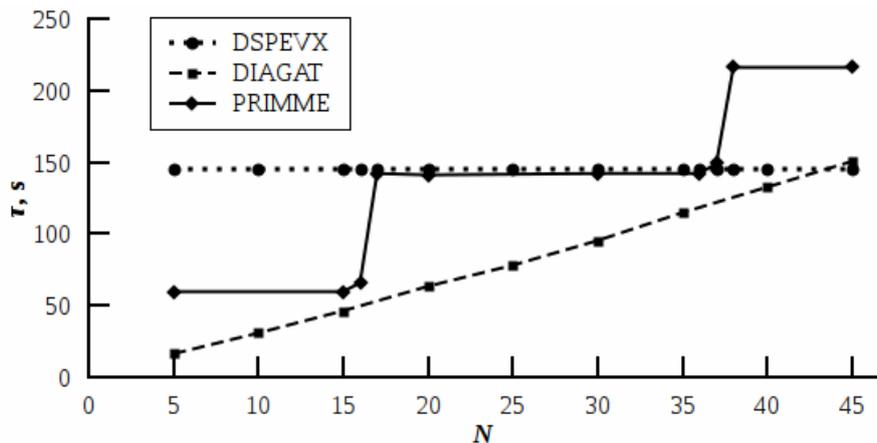


Fig 5. Diagonalization time  $\tau$  dependence on the number of  $N$  when  $M = 5000$ .

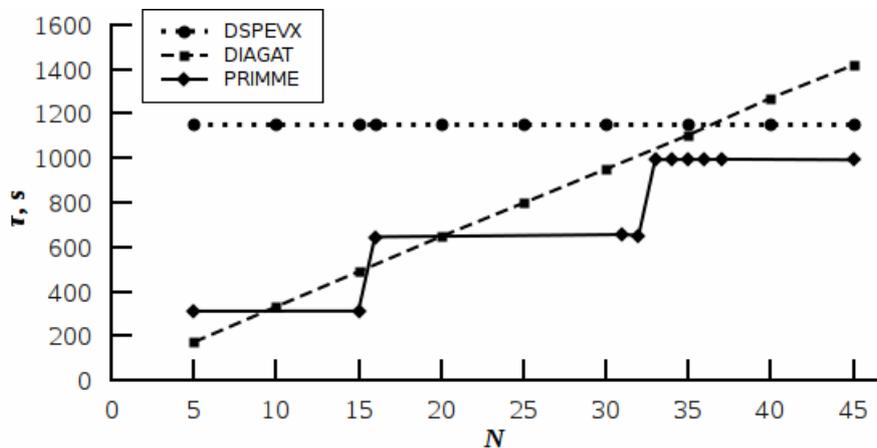


Fig 6. Diagonalization time  $\tau$  dependence on the number of  $N$  when  $M = 10000$ .

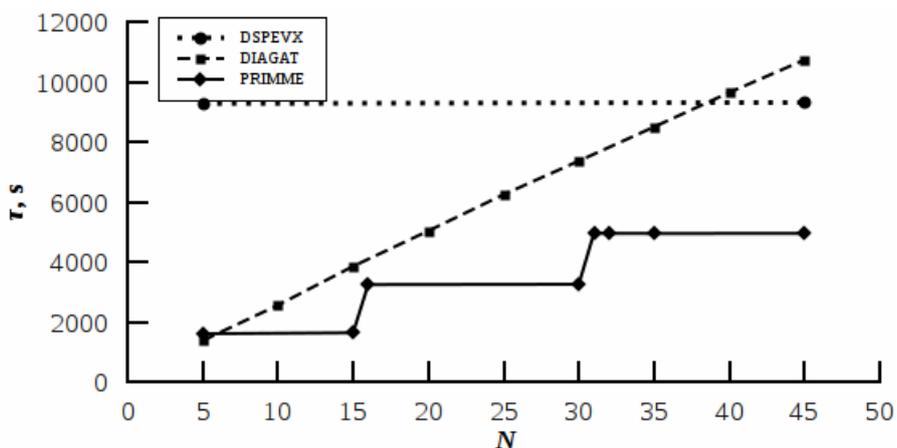


Fig 7. Diagonalization time  $\tau$  dependence on the number of eigen-values  $N$  when  $M = 20000$ .

As seen from the figures the central processor time does not depend on  $N$  in the case of DSPEVX program. It is natural and conforms to the properties of the exploited algorithm of reduction of a matrix to a triangular form. Noticeable advantages of this program show up when  $N$  values exceed 50. However there is no such number of the eigen-values under our investigation in the tasks of atomic theory as this number does not usually exceed 10. Jakobi-Davidson method develops in step-like dependence on  $N$ . Its considerable advantage comparing to DIAGAT shows up on for high order of matrices and values of  $N$  exceeding 10.

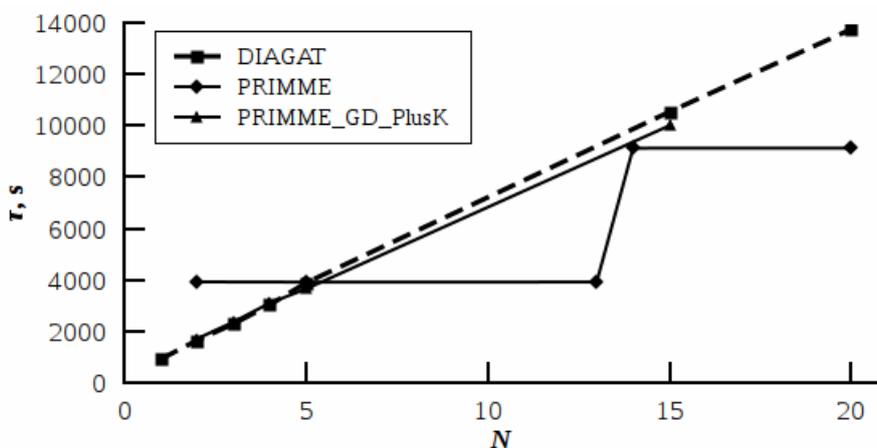


Fig 8. Diagonalization time  $\tau$  dependence on the number of eigen-values  $N$  when  $M = 30000$ .

In order to evaluate the effectiveness of the programs at low numbers of eigen-values the matrices with  $M = 30000$  were diagonalized. The generalized DAVIDSON algorithm with locally optimal restarting for  $k$  eigen-values (PRIMME-GD-pulsK) from the same library was used besides the mentioned earlier. The obtained data are presented in figure 8. As seen there the results of DIAGAT and PRIMME-GD-pulsK almost coincide with each other.

#### 4. Conclusion

Performed investigation of the effectiveness of programs for diagonalization of symmetric real matrices revealed that DIAGAT program created in Department of Theory of Atom at Institute of Theoretical Physics and Astronomy is sufficiently effective for the tasks of atomic

physics. It is caused by the facts that the orders of the matrices possible to diagonalize by available computers do not exceed forty thousands and the number of sought eigen-values does not go beyond 6-7 as a rule. Conversion to GRID-technologies enables one to operate much larger matrices. It in turn gives an opportunity to solve physically more complicated tasks. DIAGAT program needs to be parallelized in order to utilize multi-processor systems. The algorithm is perfectly suitable for that. It is necessary to incorporate JDQMR program to the complex for calculation and diagonalization of energy matrices and to explore its possibilities in real calculations. Since the method applied in DSPEVX program becomes effective only when a number of sought eigen-values is relatively large it is not appropriate to utilize it in our programs.

### **Acknowledgments**

This work was supported, in part, by the Lithuania-Ukraine projects V-20/2009 (PB and OR).

### **References**

- [1]. Ch. Froese Fischer // *Comp. Phys. Comm.*, 128, (2000) 635.
- [2]. Ch. Froese Fischer, // *Comp. Phys. Comm.*, 64, (1991).473.
- [3] P. Bogdanovich, S. Šadžiuvienė and V. Tutlys, // *Lith. Phys. J.*, 36 (1996) 354.
- [4] P. Bogdanovich, R. Karpuškienė and A. Momkauskaitė, // *Comp. Phys. Comm.*, 143, (2002) 174.
- [5] E. Anderson, Z. Bai, C. Bischof et al., *LAPACK Users' Guide*, Society for Industrial and Applied Mathematics, 1999, 0-89871-447-8.
- [6] A. Stathopoulos and J. R. McCombs // *Sci. Comput.*, 29, (2007) 2162.