Computing for Science (CFS) Ltd.,
CCLRC Daresbury Laboratory.

Generalised Atomic and Molecular Electronic Structure System

G A M E S S - U K

USER'S GUIDE and REFERENCE MANUAL

Version 8.0 June 2008

PART 7. RPA and MCLR CALCULATIONS

M.F. Guest, J. Kendrick, J.H. van Lenthe and P. Sherwood

Contents

1 Introduction

Under the common runtype RESPONSE, the user may perform calculations of electronic transition energies and corresponding oscillator strengths, using either the Random Phase Approximation (RPA) method or the Multiconfigurational Linear Response (MCLR) procedure. The RPA calculations may be performed either within the conventional approach where the two–electron integrals are transformed or with a "direct" implementation. The next sections describe in detail how to perform calculations with the linear response module.

2 RPA and TDA calculations

Calculations of excitation energies and oscillator strengths based on the Random Phase Approximation (RPA) are initiated by specifying the data line

```
RUNTYPE RESPONSE RPA
```

in the input file. Data input characterising the details of the calculation is presented immediately after the RUNTYPE data line. Termination of this data is accomplished by presenting a valid *Class 2* directive, such as VECTORS.

RUNTYPE RESPONSE is in fact a combination of tasks, requesting integral generation, SCF, integral transformation (in conventional RPA calculations) and, finally, the response calculation itself. While in simple cases it may be feasible to perform all steps in a single calculation, it will often be necessary to break up the calculation into multiple jobs, driving through each of the tasks under control of the appropriate RUNTYPE directive, with use made of the BYPASS directive in the latter stages of the computation.

The data input required when performing an RPA calculation in a single job is shown schematically below:

```
TITLE
H2CO - TZVP + R(SP) BASIS - RPA CALCULATION
SUPER OFF NOSYM
 .
 .
RUNTYPE RESPONSE RPA
 .
 .
 RPA Data
 .
ENTER
```

When splitting the RPA calculation into multiple steps, we will be involved in performing the following tasks:

- A closed-shell SCF calculation requesting, through use of the SUPER directive, full integral list generation required in the subsequent transformation. Note that in most cases this

part of the calculation may also be conveniently broken up into two parts, the first being a normal SCF where the integrals are generated in supermatrix form, and the second being a restart job with the "SUPER OFF NOSYM" data line. Dumpfile (ED3) and Mainfile (ED2) from the SCF calculation must be kept.

```
TITLE
H2CO - TZVP + R(SP) BASIS - SCF PRIOR TO RPA CALCULATION
SUPER OFF NOSYM
 .
 .
 .
ENTER
```

- An integral transformation, using the MOs from the SCF job as input vectors. The Dumpfile (ED3) and the transformed integral file (ED6) must be kept.

```
RESTART
TITLE
H2CO - TZVP + R(SP) BASIS - INTEGRAL TRANSFORMATION
SUPER OFF NOSYM
BYPASS SCF
  .
RUNTYPE TRANSFORM
  .
ENTER
```

  Note that the SCF computation is BYPASS'ed, with the SCF vectors from the first run now restored from the default closed shell SCF eigenvector section (section 1) and used in the transformation.

- The final RPA job must be declared as a restart job, and BYPASS's the integral transformation.

```
RESTART
TITLE
H2CO - TZVP + R(SP) BASIS - CONVENTIONAL RPA CALCULATION
SUPER OFF NOSYM
BYPASS TRANSFORM
  .
RUNTYPE RESPONSE RPA
  .
  RPA Data
  .
ENTER
```

  The RPA input data is terminated by the ENTER directive, where the default SCF eigenvectors section is again in the absence of section specification.

In the following sections the directives controlling the RPA calculation are described.

2.1 RPA Data - SYMM

The computation of excitation energies and corresponding oscillator strengths is initiated by the SYMM directive comprising the variables TEXT1, ISYM, ILOW, TEXT2, IHIGH using format (A,2I,A,I).

- TEXT1 should be set to the character string SYMM.
- TEXT2 should be set to the character string TO.

The excited states no. ILOW to IHIGH of the irreducible representation ISYM are then computed.

2.1.1 SYMM Example

Calculation of the excitation energies for the lowest five states of each of the optically allowed symmetries B_{1u}, B_{2u}, B_{3u} of a molecule with D_{2h} symmetry requires the data lines

```
SYMM 2 1 TO 5
SYMM 3 1 TO 5
SYMM 5 1 TO 5
```

in the input.

2.2 RPA Data - TDA

By presenting the data line

```
TDA
```

an additional Tamm-Dancoff (TDA) calculation is performed for the specified irreps and roots, corresponding to a CI in the space of single excitations. The line

```
TDA ONLY
```

can be used to suppress the RPA calculation, performing a TDA calculation only.

2.3 RPA Data - POLA

This directive initiates the computation of dynamic polarisabilities within the time-dependent Hartree-Fock model. Each data line should begin with the character string POLA, followed by one or several (up to 10) floating point numbers, representing the frequencies (in atomic units) for which the polarisabilities are requested. If more than 10 frequencies are required, a new data line must be presented.

2.4 RPA Data - MAXRED

The maximal size of the reduced matrices in the iterative RPA algorithm can be adjusted with the MAXRED directive, consisting of a single dataline read to the variables TEXT, MAXR, MAXRR using format (A,2I).

- TEXT should be set to the character string MAXRED.
- MAXR is an integer, specifying the maximal size of the reduced matrices in the iterative eigenvalue search. The default value of MAXR is 50.
- MAXRR is an integer, specifying the maximal size of the reduced matrices during solution of the linear response equations for polarisability calculations. The default value of MAXRR is 50. Within the MAXRED directive, the specification of MAXRR is optional.

Note that there is no danger in specifying a value of MAXR (or MAXRR) which is too large with respect to the memory available. The program automatically adjusts the value of MAXR if the iterative procedure consumes too much memory. If the user, however, absolutely insists on the desired value for MAXR, she/he should increase the available memory using the MEMORY predirective. The value of MAXR should at least be three times as large as the largest number of eigenvalues wanted for one specific symmetry, except for very large calculations. The following table gives recommended values for MAXR:

Number of eigenvalues desired	Recommended value of MAXR
10	50 - 80
20	80 - 100
30	90 - 120
50	150 - 200
100	200 - 300
200	400

If the iterative algorithm reaches the maximum dimension before convergence, it restarts the procedure, using the current approximate eigenvectors as starting vectors.

2.5 RPA Data - MAXIT

This directive sets the maximal number of iterations for the iterative algorithms in the RPA module. The data line

```
MAXIT 50 10
```

sets the maximum number of cycles to 50 for the eigenvalue problem and to 10 for the linear equation solver. The default value is 30 for both algorithms. As for MAXRED, specification of the second integer is optional within the MAXIT directive.

2.6 RPA Data - ANALYSE

The result table printed after successful completion of the iterative TDA/RPA procedure contains the most important one-electron excitations of the corresponding states. If $(\boldsymbol{y}, \boldsymbol{z})$ denotes an RPA eigenvector, then all components of the vector $\boldsymbol{y} - \boldsymbol{z}$ with modulus larger than a certain threshold which may be specified in the THRESH directive (see below) are listed in this table. With the help of the ANALYSE directive which consists of the single data line ANALYSE, the user may, in addition, examine smaller components, without having them listed in the result table. The corresponding threshold is again specified by means of the THRESH directive (see below). The additional output generated by the ANALYSE directive also contains the dipole integrals, useful for an investigation which mono-excitations contribute to a large oscillator strength, and the weights of the vectors $\boldsymbol{y}$ and $\boldsymbol{z}$ in the RPA eigenvectors $(\boldsymbol{y}, \boldsymbol{z})$.

2.7 RPA Data - THRESH

It is possible to define various thresholds to control the convergence and output. The first data line of the THRESH directive should be set to the character string

```
THRESH
```

Each following line should begin with one of the character strings

```
EIGEN, EQSYS, TABLE, ANALYSIS
```

followed by a floating point number, using format (A,F). The last data line should be set to the character string

```
END
```

The numbers following the keywords EIGEN, EQSYS define the thresholds to which the residual in the eigenvalue algorithm and equation system solver, respectively, is converged. The default value is 0.001 for both algorithms. The data line

```
TABLE 0.1
```

causes printing of all eigenvector components with modulus larger than or equal to 0.1 in the result tables for TDA and RPA. The default value is 0.2. Finally, the data line

```
ANALYSIS 0.01
```

causes printing of all eigenvector components c_i with $c_i^2 \geq 0.01$, provided a detailed analysis of the eigenvectors is requested by means of the ANALYSE directive.

2.8 RPA Data - BEGIN

In the result table of an RPA calculation, the leading components of the eigenvectors are listed, attached with symmetry labels of the orbitals involved in the corresponding one-electron excitation, e.g.

$$0.85\ (2\,a_1 \rightarrow 1\,b_2)$$

Usually the labeling starts with the first MO, i.e. the one with the lowest energy. Now, if there is a core of orbitals from which virtually no excitations are expected in the lowest excited states, the user may wish to start labeling the orbitals at the first valence MO. This is accomplished with the BEGIN directive which consists of a single data line, read to variables TEXT, NBEGIN using format (A,I).

- TEXT should be set to the character string BEGIN.
- NBEGIN is an integer specifying the number of the first MO to be labeled.

2.8.1 BEGIN Example

In an all-electron calculation on $MgNa_6$, there are 70 electrons in doubly occupied core orbitals. The first valence MO thus carries the number 36, and the data line

```
BEGIN 36
```

starts the orbital counting with this MO.

2.9 RPA Data - QZ

Usually the RPA calculation is carried out for the ground state of a molecule. In this case the symmetric RPA matrix is positive definite, and the projected generalized eigenvalue problems can be reduced to ordinary eigenvalue problems. This is the default strategy of the iterative algorithm. If, however, the user wishes to perform a calculation on a state different from the ground state, or if the RPA matrix is extremely ill-conditioned, she/he can resort to the QZ algorithm of Moler and Stewart which treats generalized indefinite eigenvalue problems. This path detects and discards complex eigenvalues. It is initiated by presentation of the single data line

```
QZ
```

Note that the QZ algorithm is considerably slower than the standard method for positive definite matrices. The occurrence of complex eigenvalues during the standard reduction is usually indicated by the message

```
PROBLEM IN ITERATION STEP 1: ERED2 IS NOT POSITIVE DEFINITE
```

which terminates the RPA procedure for that particular symmetry. In this case the user should first check if she/he is really calculating on the Hartree-Fock ground state. Note that the QZ algorithm is the default in Direct RPA calculations.

2.10 Auxiliary files

The RPA program automatically generates a file named `rpa_spectrum` (and/or `tda_spectrum`, if a TDA calculation is performed), which contains a table of the excitation energies (in a.u.) and corresponding oscillator strengths. This file may serve as input for a suitable plotting program. If the user wishes to keep this file for plotting, she/he should place a corresponding command line at the end of the shell script file which moves `rpa_spectrum` to the user's permanent directory.

The files `tda_table.tex` and `rpa_table.tex` contain the LaTeX input for a list of the excited states computed with TDA and RPA, respectively, with excitation energies, oscillator strengths and most important single excitations.

If a calculation is performed on the states of symmetry i $(1 \leq i \leq 8)$, a file `tm4`i is generated which contains the input for a plot of the RPA analogue of the transition density matrix.

Finally, we list the complete input data for an RPA plus polarisability calculation on the water molecule, using a 4-31G basis:

2.10.1 SCF calculation

```
TITLE
H2O 4-31G BASIS
SUPER OFF NOSYM
ZMAT
O
H 1 1.809
H 1 1.809 2 104.52
END
BASIS 4-31G
ENTER
```

2.10.2 Integral transformation

```
RESTART
TITLE
H2O 4-31G BASIS + INTEGRAL TRANSFORMATION
BYPASS SCF
SUPER OFF NOSYM
ZMAT
O
H 1 1.809
H 1 1.809 2 104.52
END
BASIS 4-31G
RUNTYPE TRANSFORM
ENTER
```

2.10.3 RPA calculation

```
RESTART
TITLE
H2O 4-31G BASIS RPA CALCULATION
BYPASS TRANSFORM
SUPER OFF NOSYM
ZMAT
O
H 1 1.809
H 1 1.809 2 104.52
END
BASIS SV 4-31G
RUNTYPE RESPONSE RPA
TDA
SYMM 1 1 TO 5
SYMM 2 1 TO 5
SYMM 3 1 TO 2
MAXRED 25
MAXIT 20
ANALYSE
THRESH
 EIGEN    0.001
 EQSYS    0.0001
 TABLE    0.25
 ANALYSIS 0.05
END
POLA 0.0 0.1 0.2
ENTER
```

3 Direct RPA calculations

For large atomic orbital basis sets, the integral transformation step in conventional RPA calculations can become prohibitive. In this case it is possible to resort to a "Direct SCF" like implementation of the RPA procedure which breaks up the four–index transformation into two two–index transformation whenever the RPA matrix acts on a trial vector. The Direct RPA module is started by the input lines

```
RUNTYPE RESPONSE RPA DIRECT
```

in the input file. In this case the only preparatory run is a closed shell SCF calculation which may be conventional or direct, and in which the integrals may be generated in supermatrix or 2E format. Only the Dumpfile of the SCF calculation must be kept. All directives that are available for conventional RPA calculations can also be used for the Direct RPA case, with two exceptions: The polarisability module has not yet been implemented, and the QZ directive is redundant (*vide supra*). The following additional directives are available:

3.1 PREF

This directive controls the prefactor tolerance for the integral generator. It consists of a single data line with variables TEXT, EXP using format (A,I), where TEXT is set to the character string PREF and EXP is a positive integer, setting the prefactor tolerance to $10^{-\mathsf{EXP}}$. The default value for EXP is 7.

3.2 MAXVEC

The direct RPA procedure is organized in such a way that a maximal number of trial vectors, with respect to the main memory available, is contracted with the integrals generated in one batch. It may, however, sometimes be necessary to reduce this number since valuable memory is needed for other purposes, e.g., to increase the maximal size of the reduced matrices during the iterations. The MAXVEC directive allows to limit the number of trial vectors which are treated in one batch to a specific value M. It consists of a single data line with the variables TEXT, M using format (A,I), where TEXT is set to the character string MAXVEC and M is the above mentioned integer.

3.3 Dumping and restoring trial vectors

Since Direct RPA calculations on larger systems are rather time-consuming, it is desirable to have the possibility to interrupt a calculation and restart it at a later time. Dumping and restoring intermediate results is possible with the DUMP and RESTORE directives. The user may dump vectors by specifying the data line

```
DUMP file
```

where *file* is (the complete path of) a file in some permanent directory that may be chosen freely. In case the RPA (or TDA) calculation aborts in the middle of the iterative process (either by runtime problems or by a user's operation), this file contains the approximate eigenvectors of the last iteration step that was completed. These vectors may be used in a later job to resume the calculation at that iteration step. This is accomplished by the data line

```
RESTORE RPA file
```

where *file* is the above mentioned file containing the saved vectors. Note that the keyword `RPA` has to be replaced by `TDA` if the job was interrupted during a TDA calculation. Note also that the restart run must begin with the particular irrep in which the termination occurred, and that the same number of roots must be specified.

4 Multiconfigurational linear response calculations

A multiconfigurational linear response (MCLR) calculation [1], also known under the term time–dependent multiconfigurational SCF, is performed by presenting the data line

```
RUNTYPE RESPONSE MCLR
```

in the input file. A necessary condition for performing an MCLR calculation is the successful completion of a corresponding multiconfigurational SCF calculation with the GAMESS–UK MCSCF module. The Dumpfile (ED3) and the transformed integral file (ED6) must be saved. The MCLR calculation is then performed as a restart job. In the following we list the directives which are available in the MCLR module. Note that the first four are obligatory.

4.1 MCLR Data - ORBITAL

The set of active orbitals must be specified by means of the ORBITAL directive. The individual lines of this directive must be identical to those presented in the preceding MCSCF calculation. Since this directive is described in detail in the MCSCF part (and is usually copied into the MCLR job from the MCSCF job), we refer the reader to the corresponding section.

4.2 MCLR Data - SECTIONS

In order to perform an MCLR calculation, several vectors have to be retrieved from the Dumpfile. The SECTIONS directive specifies in which sections of the Dumpfile the corresponding vectors are stored. Note that these sections must still be specified even if the default sections have been used at vector generation time (in the absence of explicit section specification on the corresponding ENTER directive).

The data lines

```
SECTIONS
   SCF        1
   MCSCF      8
   CANONICAL  10
   CIVEC      9
```

instruct the program to read the SCF eigenvectors from section 1, the MCSCF MOs from section 8, the pseudocanonical MCSCF orbitals from section 10 and the MCSCF CI vector from section 9 of the Dumpfile. Note that with the possible exception of CANONICAL (as specified by the CANONICAL MCSCF directive), these section numbers correspond to the defaults in place at vector generation time in the closed-shell SCF and MCSCF module. Note that the default section for the MCSCF NOs is now section 10, and it is this section number that should be specified in the data above. The indentation is, of course, not necessary but convenient for better reading.

4.3 MCLR Data - SYMM

This directive controls the calculation of excited states. It consists of the variables TEXT, ISYM, IHIGH using format (A,2I). TEXT should be set to the character string SYMM. The integer ISYM indicates the irreducible representation, and IHIGH is the number of roots to

be calculated in the irrep ISYM. Note that the syntax of this directive is different from the corresponding directive in the RPA module. In particular, it is not possible to calculate an interval [ILOW,IHIGH] of roots with ILOW different from 1.

4.4 MCLR Data - END

This directive which consists of the single keyword END terminates the input which controls the MCLR calculation. It must always be present.

4.5 MCLR Data - REDUCE

The default algorithm for solving the small generalized eigenvalue problems during the iterative MCLR calculation is the QZ algorithm (see the description of the QZ directive in the RPA module). The REDUCE directive which consists of a single data line with the keyword REDUCE forces the program to reduce the generalized eigenvalue problems to standard eigenvalue problems.

4.6 MCLR Data - SPLIT

This directive initiates the use of split trial vectors, as described in [2]. It consists of variables TEXT, ITER using format (A,I), where TEXT is set to the character string SPLIT and ITER is an integer indicating after which iteration step split trial vectors are to be used. Usually there is no loss of speed of convergence when split trial vectors are used from the very beginning. Setting ITER to zero is therefore recommended with this directive.

4.7 MCLR Data - Further directives

The directives MAXRED, MAXIT, BEGIN, POLA and THRESH are also available in MCLR calculations. They are identical to the corresponding directives in the RPA module, with one exception: the THRESH directive must *not* be terminated by the keyword END since this is reserved for signaling the end of the MCLR directives. The reader is referred to the corresponding sections for a detailed description of the directives.

We list below the complete input for an MCLR calculation on the water molecule, using a TZVP basis and performing a full valence space calculation. We start with the MCSCF job, omitting the SCF job(s):

4.7.1 MCSCF calculation

```
RESTART
TITLE
H2CO TZVP MCSCF
BYPASS
ZMAT
0
```

```
H 1 1.809
H 1 1.809 2 104.52
END
BASIS TZVP
RUNTYPE SCF
SCFTYPE MCSCF
MCSCF
ORBITAL
DOC1 DOC1 DOC3 DOC1 DOC2 UOC1 UOC3
END
ENTER
```

4.7.2 MCLR calculation

```
RESTART
TITLE
H2CO TZVP MCSCF
BYPASS MCSCF
ZMAT
O
H 1 1.809
H 1 1.809 2 104.52
END
BASIS TZVP
SCFTYPE MCSCF
MCSCF
ORBITAL
DOC1 DOC1 DOC3 DOC1 DOC2 UOC1 UOC3
END
RUNTYPE RESPONSE MCLR
ORBITAL
DOC1 DOC1 DOC3 DOC1 DOC2 UOC1 UOC3
END
SECTIONS
   SCF       1
   MCSCF     8
   CANONICAL 10
   CIVEC     9
SYMM 1 5
SYMM 2 5
SYMM 3 5
SPLIT 0
END
VECTORS 2 3
ENTER 2 3
```

References

[1] C. Fuchs, V. Bonačić-Koutecký and J. Koutecký, J. Chem. Phys. **98** (1993) 3121, doi:10.1063/1.464086.

[2] P. Jørgensen, H. J. Aa. Jensen, and J. Olsen, J. Chem. Phys. **89** (1988) 3654, doi:10.1063/1.454885.