ADT : A Generalized Algorithm for Beyond Born-Oppenheimer Equations for 'N' Dimensional Sub-Hilbert Space

Koushik Naskar[†,1], Soumya Mukherjee[†,1], Bijit Mukherjee[†], Satyam Ravi[†], Saikat Mukherjee[†],
Subhankar Sardar[†,‡] and Satrajit Adhikari[†,*]

[†]School of Chemical Sciences,

Indian Association for the Cultivation of Science, Jadavpur, Kolkata - 700032, India

[‡]Department of Chemistry, Bhatter College, Dantan, Paschim Medinipur - 721426, India

USER MANUAL

---

[*]Author for correspondence: e-mail: pcsa@iacs.res.in

[1]K. Naskar and S. Mukherjee contributed equally to this work.

Contents

List of Tables

List of Figures

1 Introduction

Large and complicated functional form of adiabatic-to-diabatic transformation (ADT) equations is the major bottleneck of first principle based beyond Born-Oppenheimer (BBO) treatment, specially, when the number of coupled electronic states is high ($N \geq 4$). Though explicit expressions for the ADT equations are already formulated for $N = 3$ to 6 (See Ref. 24-28 of the main text), the algorithm 'ADT' generalizes this process of evaluating the expressions for any number of coupled electronic states (N). This program package can be efficiently used to carry out the following operations:

- Formulation of analytic functional forms of ADT quantities, i.e., ADT matrix, ADT equations, coeffcient matrix of gradient of ADT angles (see Eq. 14 of the main text), coefficient matrix of nonadiabatic coupling terms (NACTs) (see Eq. 14 of the main text) and diabatic potential energy matrix for any N.

- Numerical calculation of ADT quantities (ADT angles, residue of ADT angles, ADT matrices and diabatic potential energy matrices) for any number of coupled electronic states, if the input files [*ab initio* adiabatic Potential Energy Surfaces (PESs) and NACTs] are already available.

- Interfacing of MOLPRO quantum chemistry package[1] to compute *ab initio* data, namely, adiabatic PESs and NACTs, which will serve as the input for ADT calculation if the input files (adiabatic PESs and NACTs) are not available to the user.

Finally, the program package 'ADT' can be used for the construction of diabatic surfaces of spectroscopic as well as scattering processes. For spectroscopic systems, the nuclear configuration space (CS) is expressed in terms of normal mode coordinates. On the other hand, user can employ Jacobi or hyperspherical coordinates for scattering systems.

2   Files and folder structure

Schematic Outline of Files present in 'ADT-version0' directory

ADT-version0/
```
├── setup.py ·········································· 'ADT' package installer
├── adt/
│      ├── __init__.py
│      ├── adt.py ········································ central script to run the package
│      ├── analytic/
│      │        ├── __init__.py
│      │        ├── adt_analytic.py ············· script for analytic formulation
│      │        └── anamod.py ················· collection of necessary python functions
│      ├── numeric/
│      │        ├── __init__.py
│      │        ├── adt_numeric.py ············· script for numerical calculation
│      │        └── nummod.f90 ··············· collection of necessary fortran subroutines
│      └── molpro/              If input files for numeric section are unavailable,
│               ├── __init__.py   user needs to generate those files by the molpro
│               └── adt_molpro.py  section
│                                script for ab initio calculations by MOLPRO
├── test_runs/ ······································ folder containing sample calculations
├── LICENSE ········································· license information
├── user_manual.pdf ······························· detailed instructions for users
└── README.md ····································· readme file
```
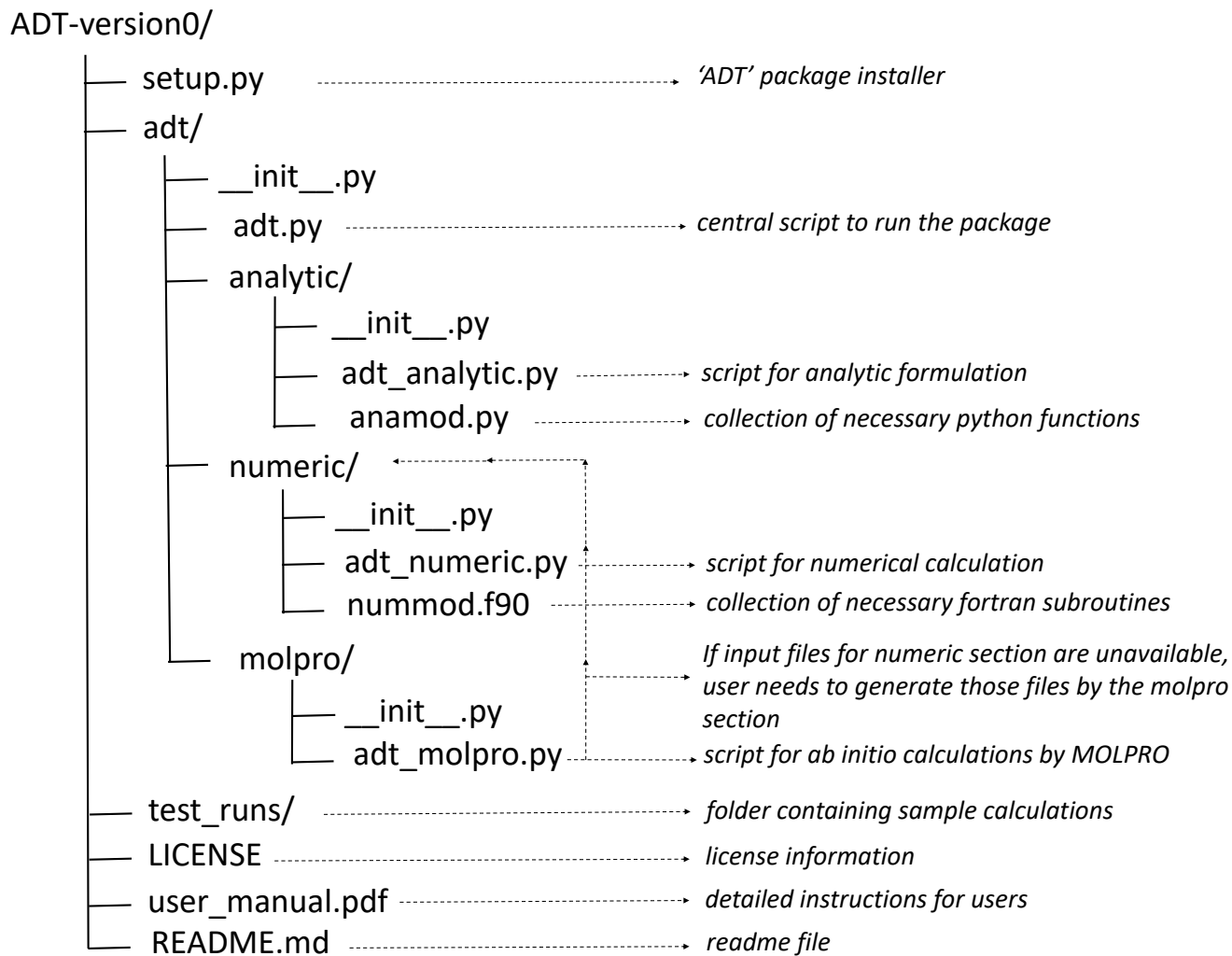
Figure 1: The above directory structure depicts the files and directories present in the parent folder, 'ADT-version0'.

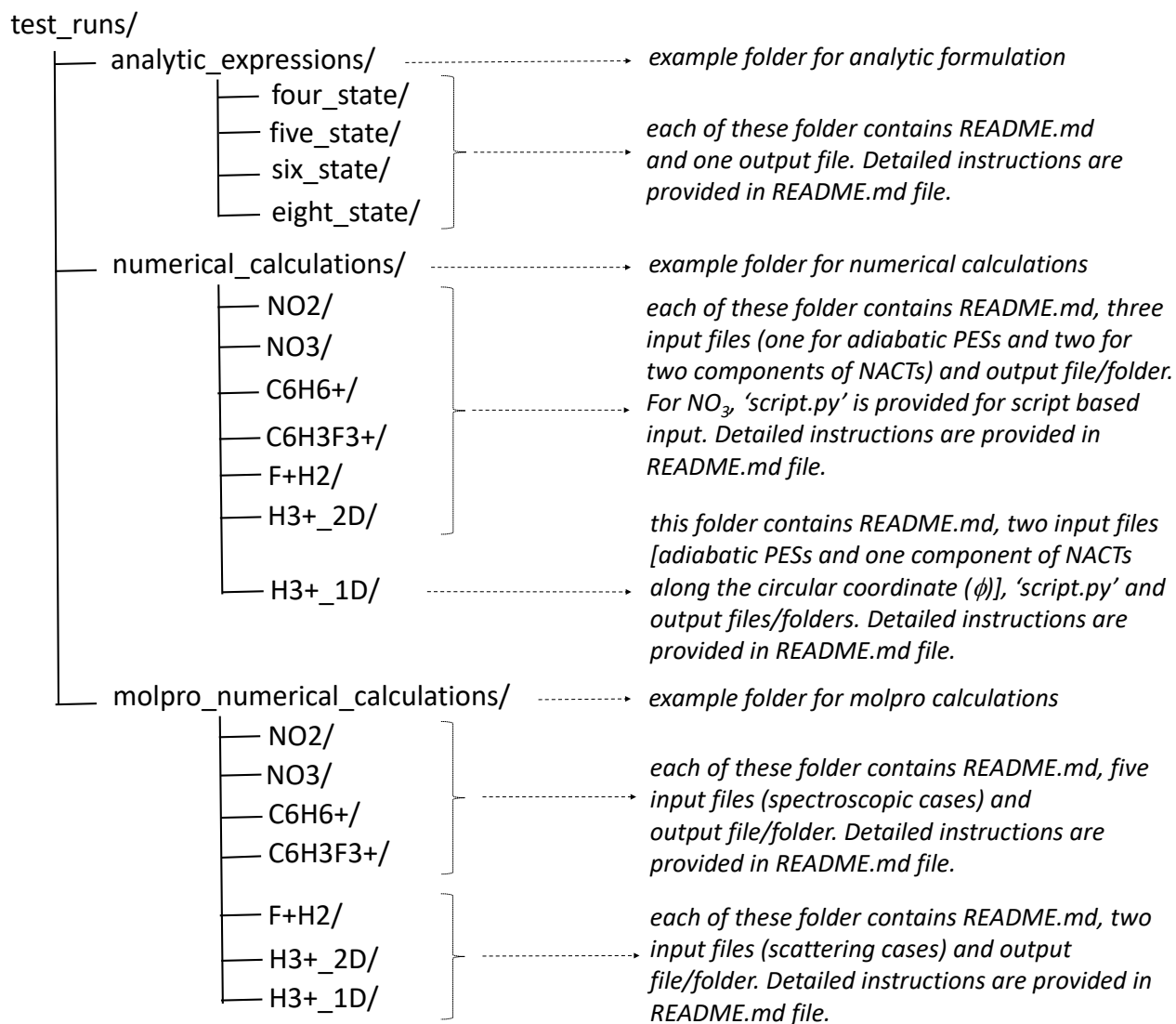**Schematic Outline of Files and Folders in 'test_runs'**

```
test_runs/
    ├── analytic_expressions/      ----------------→  example folder for analytic formulation
    │       ├── four_state/
    │       ├── five_state/                           each of these folder contains README.md
    │       ├── six_state/         ----------------→  and one output file. Detailed instructions are
    │       └── eight_state/                          provided in README.md file.
    │
    ├── numerical_calculations/    ----------------→  example folder for numerical calculations
    │       ├── NO2/                                  each of these folder contains README.md, three
    │       ├── NO3/                                  input files (one for adiabatic PESs and two for
    │       ├── C6H6+/                                two components of NACTs) and output file/folder.
    │       ├── C6H3F3+/           ----------------→  For NO3, 'script.py' is provided for script based
    │       ├── F+H2/                                 input. Detailed instructions are provided in
    │       ├── H3+_2D/                               README.md file.
    │       │
    │       │                                         this folder contains README.md, two input files
    │       │                                         [adiabatic PESs and one component of NACTs
    │       └── H3+_1D/            ----------------→  along the circular coordinate ($\phi$)], 'script.py' and
    │                                                 output files/folders. Detailed instructions are
    │                                                 provided in README.md file.
    │
    └── molpro_numerical_calculations/  ----------→  example folder for molpro calculations
            ├── NO2/
            ├── NO3/                                  each of these folder contains README.md, five
            ├── C6H6+/                 -----------→   input files (spectroscopic cases) and
            ├── C6H3F3+/                              output file/folder. Detailed instructions are
            │                                         provided in README.md file.
            │
            ├── F+H2/                                 each of these folder contains README.md, two
            ├── H3+_2D/                -----------→   input files (scattering cases) and output
            └── H3+_1D/                               file/folder. Detailed instructions are provided in
                                                      README.md file.
```

Figure 2: The above directory structure carries the information about the files and subdirectories inside the folder, 'test_runs'.

3 Prerequisites

In order to install and execute the 'ADT' program package successfully, user needs a Fortran 90 compiler and Python (2.7 or 3.x), which are to be installed apriorly on the system. In addition, the following python dependencies are also required to use the package:

1. NumPy (v 1.13.0 or above)

2. setuptools

3. h5py

The first two modules are necessary for execution of this package, whereas the third one i.e. h5py is optional and is required only for HDF5 I/O. Alternatively, one can install these libraries through any python package manager (e.g., `pip`) or build them from source.

4 Installation

For installing the 'ADT' program package with default Fortran compiler for serial mode (no parallelization), user has to execute the following command in the terminal:

```
python setup.py install
```

After compilation of all the necessary files, the package is installed in the system and a command line executable `adt` is created. By default, the installation is performed in root of the system (usually in /usr/local) and therfore, root privilege may be required. If user is unable to access the root privilege or do not wish to install in the root of system, the 'ADT' package can be installed in a local package folder for python (usually in ∼/local/) by providing the `--user` flag during installation:

```
python setup.py install --user
```

Alternatively, if `pip` is available, user can install the package by executing the following command inside the source directory:

```
pip install .
```

On the contrary, for installing the package at any other location in the system, please visit the site https://setuptools.readthedocs.io/en/latest/easy_install .html#custom-installation-locations.

In order to use the OpenMP parallelization scheme, user needs to link proper OpenMP libraries during the installation of the package by modifying the 'fort_args' and 'lib_links' variables in 'setup.py' script and execute it with corresponding Fortran compiler. As for example, uncomment the following lines in 'setup.py' to install 'ADT' using 'gfortran' compiler:

```
fort_args = ['-fopenmp']
lib_links = ['-lgomp']
```

and execute the following command:

```
python setup.py config_fc --fcompiler=gnu95 install
```

whereas in order to install the package using 'ifort' compiler, uncomment the following lines in 'setup.py':

```
fort_args = ['-qopenmp']
lib_links = ['-liomp5']
```

and execute the following command:

```
python setup.py config_fc --fcompiler=intelem install
```

During installation, the Fortran file is compiled to a python extension module by numpy distutils and to further configure the compilation process, user can visit the official NumPy distutils page (https://docs.scipy.org/doc /numpy/reference/distutils.html#module-numpy.distutils).

5 Execution

After installing the 'ADT' package, one has to execute the command line utility, `adt` to formulate analytic expressions or to perform numerical calculation or MOLPRO interfacing followed by numerical computation of ADT quantities employing the subcommands `ana`, `num` and `mol`, respectively. The details of the execution of these different sections are given in the following diagram:

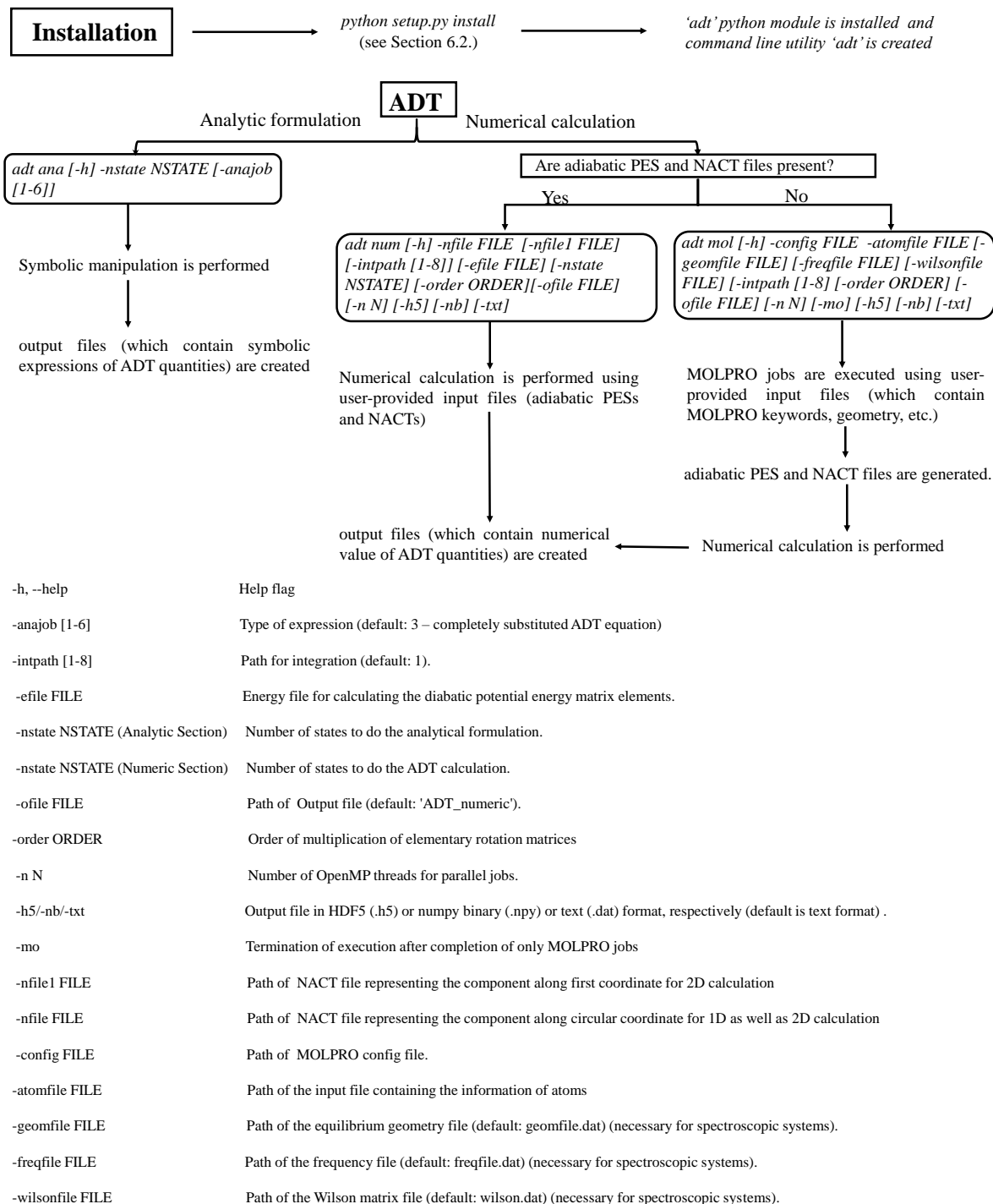**Working Principles and Command-line-arguments for Analytic, Numeric and Molpro Jobs**

Installation → *python setup.py install* (see Section 6.2.) → *'adt' python module is installed and command line utility 'adt' is created*

ADT

Analytic formulation — Numerical calculation

adt ana [-h] -nstate NSTATE [-anajob [1-6]]

Are adiabatic PES and NACT files present?

Yes — No

adt num [-h] -nfile FILE [-nfile1 FILE] [-intpath [1-8]] [-efile FILE] [-nstate NSTATE] [-order ORDER][-ofile FILE] [-n N] [-h5] [-nb] [-txt]

adt mol [-h] -config FILE -atomfile FILE [-geomfile FILE] [-freqfile FILE] [-wilsonfile FILE] [-intpath [1-8] -order ORDER] [-ofile FILE] [-n N] [-mo] [-h5] [-nb] [-txt]

Symbolic manipulation is performed

output files (which contain symbolic expressions of ADT quantities) are created

Numerical calculation is performed using user-provided input files (adiabatic PESs and NACTs)

MOLPRO jobs are executed using user-provided input files (which contain MOLPRO keywords, geometry, etc.)

adiabatic PES and NACT files are generated.

output files (which contain numerical value of ADT quantities) are created ← Numerical calculation is performed

-h, --help	Help flag
-anajob [1-6]	Type of expression (default: 3 – completely substituted ADT equation)
-intpath [1-8]	Path for integration (default: 1).
-efile FILE	Energy file for calculating the diabatic potential energy matrix elements.
-nstate NSTATE (Analytic Section)	Number of states to do the analytical formulation.
-nstate NSTATE (Numeric Section)	Number of states to do the ADT calculation.
-ofile FILE	Path of Output file (default: 'ADT_numeric').
-order ORDER	Order of multiplication of elementary rotation matrices
-n N	Number of OpenMP threads for parallel jobs.
-h5/-nb/-txt	Output file in HDF5 (.h5) or numpy binary (.npy) or text (.dat) format, respectively (default is text format) .
-mo	Termination of execution after completion of only MOLPRO jobs
-nfile1 FILE	Path of NACT file representing the component along first coordinate for 2D calculation
-nfile FILE	Path of NACT file representing the component along circular coordinate for 1D as well as 2D calculation
-config FILE	Path of MOLPRO config file.
-atomfile FILE	Path of the input file containing the information of atoms
-geomfile FILE	Path of the equilibrium geometry file (default: geomfile.dat) (necessary for spectroscopic systems).
-freqfile FILE	Path of the frequency file (default: freqfile.dat) (necessary for spectroscopic systems).
-wilsonfile FILE	Path of the Wilson matrix file (default: wilson.dat) (necessary for spectroscopic systems).

Figure 3: This diagram presents an overall schematic outline of the working principles for analytic expressions generation, numerical calculations and molpro interfacing. Detailed description of the arguments, `-anajob` and `-intpath` are presented in Tables 1 and 2.

5.1. Analytic

For analytic jobs, execute the following command line arguments,

---

```
adt ana [-h] -nstate NSTATE [-anajob [1-6]]


optional arguments:
  -h, --help     show the help message
  -anajob [1-6]  Specify the type of expression (default: 3 -
                 completely substituted ADT equation)


Required arguments:
  -nstate NSTATE Number of states
```

The possible values of **-anajob** and their significance are presented in the following table:

Table 1: This table depicts the correlation between the possible flags of the keyword argument, `-anajob` and the desired output

Keyword Argument (-anajob)	Analytical Quantities
1	elements of adiabatic to diabatic transformation (ADT) matrix
2	partially substituted forms of ADT equations [let say, for state=3, $\vec{\nabla}\theta_{12} = -\vec{\tau}_{12} - \sec\theta_{12}\tan\theta_{13}\vec{\tau}_{23} + \tan\theta_{12}\tan\theta_{13}\vec{\nabla}\theta_{13}, ..]$
3	complete forms of ADT equations (gradients in right hand side are replaced) [let say, for state=3, $\vec{\nabla}\theta_{12} = -\vec{\tau}_{12} - \sin\theta_{12}\tan\theta_{13}\vec{\tau}_{13} - \cos\theta_{12}\tan\theta_{13}\vec{\tau}_{23}, ..]$
4	elements of coefficient matrix of gradient of ADT angles (see Eq. 14 in the article for details)
5	elements of coefficient matrix of NACTs (see Eq. 14 in the article for details)
6	elements of diabatic potential energy matrix (detailed expressions)

Examples:

1. Execute the following command to formulate completely substituted ADT equations for a five electronic state sub-Hilbert space:

```
adt ana -nstate 5 -anajob 3
```

The output file, 'ADT_EQUATIONS_COMPLETE.DAT' contains the analytic expressions of ADT equations.

2. Execute the following command to formulate diabatic potential energy matrix elements for a six electronic state sub-Hilbert space:

```
adt ana -nstate 6 -anajob 6
```

The output file is 'W_MATRIX.DAT'.

3. Execute the following command to formulate coefficient matrix of gradient of ADT angles (Eq. 14, described in the article) for a three electronic state sub-Hilbert space:

```
adt ana -nstate 3 -anajob 4
```

The output file is 'GRADCOEFF.DAT'.

4. Execute the following command to formulate coefficient matrix of NACTs (Eq. 14, described in the article) for a eight electronic state sub-Hilbert space:

```
adt ana -nstate 8 -anajob 5
```

The output file is 'TAUCOEFF.DAT'.

5.2. Numeric

User can carry out numerical calculations by using the subcommand `num`,

```
adt num [-h] -nfile FILE [-mfile FILE] [-intpath [1-8]]
[-efile FILE] [-nstate NSTATE] [-order ORDER] [-ofile FILE]
[-n N] [-h5] [-nb] [-txt]


optional arguments:
  -h, --help      show the help message


  -mfile FILE     Specify the path of the input NACT file
                  representing the component of NACT
                  for the non-circular coordinate
                  (e.g. {0,pi/2} or {0,pi} or
                  {0,infinity})


  -intpath [1-8]  Specify the path for calculation (default: 1).


  -efile FILE     Specify the path of the adiabatic PES file for
                  calculating the diabatic potential energy matrix
                  elements.


  -nstate NSTATE  Specify the number of states to do the
                  calculation. By default it includes all
                  the data for calculation.


  -order ORDER    Write the order of multiplication of
                  the elementary rotation matrices.


  -ofile FILE     Specify the path of the output file/folder
                  (w/o extension) (default: 'ADT_numeric').
```

```
  -n N            Specify the number of OpenMP threads for parallel
                  jobs (default: 1).


  -h5             Write results in a HDF5 file (.h5).
                  Fast IO, smaller file size and
                  hierarchical filesystem-like data format,
                  preferable for saving and sharing large
                  datasets in an organised way.


  -nb             Write results in NumPy binary file (.npy).
                  Preferable when working with NumPy for its
                  much faster IO and easy portability.


  -txt            Write results in a text file (default
                  behaviour).

Required arguments:
  -nfile FILE     Need to specify the path of the input
                  NACT file, which is required both for
                  1D as well as 2D calculation. This file
                  represents the component of NACT for
                  the circular coordinate (e.g. {0,2pi}).
```

The probable values of -intpath and their significance are depicted in the following table:

Table 2: This table shows the relation between the possible flags of the keyword argument `-intpath` and the integration paths

Argument (`-intpath`)	Path of Integration (R : first coordinate, P : second coordinate)
1	1. At lowest value of R grid, the integration is performed for each positive step of P grid and the values of ADT angles are stored; 2. With those values of the variables at each P grid, integration is carried out for each positive increment of R grid
2	1. At lowest value of R grid, the integration is performed for each negative step of P grid and the values of ADT angles are stored; 2. With those values of the variables at each P grid, integration is carried out for each positive increment of R grid
3	1. At highest value of R grid, the integration is performed for each positive step of P grid and the values of ADT angles are stored; 2. With those values of the variables at each P grid, integration is carried out for each negative increment of R grid
4	1. At highest value of R grid, the integration is performed for each negative step of P grid and the values of ADT angles are stored; 2. With those values of the variables at each P grid, integration is carried out for each negative increment of R grid
5	1. At lowest value of P grid, the integration is performed for each positive step of R grid and the values of ADT angles are stored; 2. With those values of the variables at each R grid, integration is carried out for each positive increment of P grid
6	1. At lowest value of P grid, the integration is performed for each negative step of R grid and the values of ADT angles are stored; 2. With those values of the variables at each R grid, integration is carried out for each positive increment of P grid
7	1. At highest value of P grid, the integration is performed for each positive step of R grid and the values of ADT angles are stored; 2. With those values of the variables at each R grid, integration is carried out for each negative increment of P grid
8	1. At highest value of P grid, the integration is performed for each negative step of R grid and the values of ADT angles are stored; 2. With those values of the variables at each R grid, integration is carried out for each negative increment of P grid

User can also employ the numerical segment of this package as a python module. While carrying out 1D and 2D calculation, one can use two functions, `adt1d` and `adt2d`, respectively, which take adiabatic PESs and NACTs as inputs, and return the ADT quantities (ADT angles, residue of ADT angles, ADT matrices and diabatic PESs).

Examples

1. **Using the command line executable `adt`**

   (i) Execute the following command in terminal to carry out 2D ADT calculation, where 'Tau_Rho.dat', 'Tau_Phi.dat' and 'Adiabatic_PES.dat' contain components of NACTs along non-circular coordinate, components of NACTs along circular coordinate and adiabatic PESs, respectively. The integration is done along path 6.

   ```
   adt num -mfile Tau_Rho.dat -nfile Tau_Phi.dat
   -efile Adiabatic_PES.dat -intpath 6
   ```

   While running the command, ADT angles, residues of ADT angles, ADT matrices and diabatic potential energy matrices are computed and those are written in plain text files (default output format) inside a folder, 'ADT_numeric_6' .

   (ii) Following command can be used to perform 2D ADT calculation, where 'Tau_Theta.npy', 'Tau_Phi.npy' and 'Adiabatic_PES.npy' contain components of NACTs along non-circular coordinate, components of NACTs along circular coordinate and adiabatic PESs, respectively. The integration is done along path 4.

   ```
   adt num -mfile Tau_Theta.npy -nfile Tau_Phi.npy
   -efile Adiabatic_PES.npy -intpath 4 -nb
   ```

   Output files will be saved in NumPy binary format inside a folder, 'ADT_numeric_4'.

   (iii) Following command will carry out 2D ADT calculation, where 'Tau_Rho.h5', 'Tau_Phi.h5' and 'Adiabatic_PES.h5' contain components of NACTs along non-circular coordinate, component of NACTs along circular coordinate and adiabatic PESs, respectively. The

integration is done along path 2.

```
adt num -mfile Tau_Rho.h5 -nfile Tau_Phi.h5
-efile Adiabatic_PES.h5 -intpath 2 -h5
```

Output result will be saved as a HDF5 file, 'ADT_numeric_2.h5' and the ADT quantities will be stored as different groups of dataset in that HDF5 file.

(iv) If the 'ADT' package is installed with OpenMP enabled, user can employ `-n` flag to run the program in parallel threads. The following command executes the program package using 4 OpenMP threads.

```
adt num -mfile Tau_Rho.dat -nfile Tau_Phi.dat
-efile Adiabatic_PES.dat -intpath 6 -n 4
```

The input and output file structures are similar as the serial ones.

(v) If one wishes to perform 1D ADT, the following command can be executed with usual I/O files as the previous cases,

```
adt num -nfile Tau_Phi.dat -efile Adiabatic_PES.dat
```

The command will generate the ADT angle, residues of ADT angles, ADT matrices and diabatic potential energy matrices, and store the output in plain text files (default output format) inside a folder, 'ADT_numeric_1D'.

(vi) By default calculations are done with ADT matrix formed using the default order of multiplication of the elementary rotaion matrices $(A^{ij}(\Theta_{ij}))$ where the matrices are multiplied in increasing order of the second index (j) and then, for a fixed second index, in increasing order of the first index (i). If user wants to use any other order of multiplication, the order can be provided using the `-order` keyword. For example, for a 3 state sub-Hilbert space to form the ADT matrix using the order '12,23,13' i.e.

$$A(\Theta_{12}, \Theta_{23}, \Theta_{13}) = A^{12}(\Theta_{12}) \cdot A^{23}(\Theta_{23}) \cdot A^{13}(\Theta_{13})$$

execute the following command:

```
adt num -nfile Tau_Phi.dat -efile Adiabatic_PES.dat -order 12,23,13
```

The structure of input and output files are same as above.

2. **Using python script**

(i) **1D:** The following sample python script shows the usage of the 'ADT' package as a python module to calculate 1D ADT. The files 'Tau_Phi.dat' and 'Adiabatic_PES.dat' are the NACT (along the circular coordinate) and adiabatic PES files. As for example, those two files can be taken from the 'ADT_version0/test_runs /numerical_calculation/H3+_1D'.

```
import numpy as np
from adt.numeric import adt_numeric as an

nact = np.loadtxt('Tau_Phi.dat')
enr = np.loadtxt('Adiabatic_PES.dat')

angle, res, amat, db = an.adt1d(nact[:,0], nact[:,1:],
enr[:,1:])
```

(ii) **2D:** This python script shows 2D calculation of ADT quantities along path 6, where the files 'Tau_Rho.npy', 'Tau_Phi.npy', 'Adiabatic_PES.npy' contain components of NACTs along non-circular coordinate, components of NACTs along circular coordinate and adiabatic PESs, respectively. The input files contain *ab initio* data for five electronic manifold spanning 50×180 grid points and can be taken from the 'ADT_version0/test_runs /numerical_calculation/NO3'.

```
import numpy as np
from adt.numeric import adt_numeric as an

nact1 = np.load('Tau_Rho.npy')
nact2 = np.load('Tau_Phi.npy')
enr   = np.load('Adiabatic_PES.npy')
```

```
grid1 = np.unique(nact1[:,0])

grid2 = np.unique(nact1[:,1])


#If the number of grid points along the first and second

#coordinates are 50 and 180, respectively over 5

#electronic states with 10 NACT elements


enr = enr[:,2:].reshape(50, 180, 5)

nact1 = nact1[:,2:].reshape(50, 180, 10)

nact2 = nact2[:,2:].reshape(50, 180, 10)


#The integration is performed along path 6.


angle, res, amat,db = an.adt2d(grid1, grid2, nact1,

nact2, 6, enr)
```

5.3. Molpro

On the other hand, user can take the benefit of molpro section, where *ab initio* electronic structure calculations are performed according to the user provided input files (discussed in detail in Section 7) and then, the outputs (adiabatic PESs and NACTs) are used for subsequent ADT calculation (see Figure 3). User has to execute the following command,

```
adt mol [-h] -config FILE -atomfile FILE [-geomfile FILE]
[-freqfile FILE] [-wilsonfile FILE] [-intpath [1-8]] [-order ORDER]
[-ofile FILE] [-n N] [-mo] [-h5] [-nb] [-txt]


optional arguments:
  -h, --help      show the help message


  -geomfile FILE  Specify the geometry file containing the initial
```

```
                 grid point in 'xyz' format (in Angstrom)
                 (default: geomfile.dat).
                 (Ignore for scattering system).


-freqfile FILE   Specify the frequency information file, where
                 frequencies of normal modes are written in cm-1
                 (default: frequency.dat).
                 (Ignore for scattering system)


-wilsonfile FILE Specify the filename containing the Wilson matrix
                 of a molecular species (default: wilson.dat).
                 (Ignore for scattering system).


-intpath [1-8]   Specify the path for calculation (default: 1).


-order ORDER     Write the order of multiplication of
                 the elementary rotation matrices.


-ofile FILE      Specify the path of the output file/folder
                 (w/o extension) (default: 'ADT_numeric').


-n N             Specify the number of OpenMP threads for parallel
                 jobs (default: 1).


-mo              Terminate the execution only after completion
                 of MOLPRO jobs


-h5              Write results in a HDF5 file (.h5).
                 Fast IO, smaller file size and
                 hierarchical filesystem-like data format,
                 preferable for saving and sharing large
                 datasets in an organised way.


-nb              Write results in NumPy binary file (.npy).
                 Preferable when working with NumPy for its
                 much faster IO and easy portability.
```

```
 -txt              Write results in a text file. (default
                   behaviour)



Required arguments:
 -config FILE      Specify the MOLPRO configuration file containing
                   the necessary keywords of MOLPRO software.


 -atomfile FILE    Specify the information file constituting atomic
                   symbols and atomic masses.
```

For spectroscopic cases, user has to supply five input files:

1. MOLPRO configuration file: contains MOLPRO keywords (like memory, basis set, method of calculation, etc.), system information (like symmetry, participating normal modes) and grid information

2. atom information file: contains atomic symbols and atomic masses

3. equilibrium geometry file: contains equilibrium geometry in xyz format (in Å)

4. frequency file: contains normal mode frequencies (in cm^{-1})

5. wilson matrix file: contains mass-weighted second derivative matrix (Wilson matrix).

On the other hand, if user wishes to perform *ab initio* calculation of adiabatic PESs and NACTs for scattering systems, the first two files, i.e. MOLPRO configuration and atom information files are necessary and sufficient. The details of the above files are given in Section 7.

In order to store the geometry and MOLPRO output files for every successful and unsuccessful jobs, two folders, 'CompleteJobs' and 'IncompleteJobs' are generated, respectively during the execution of molpro section. One can use -mo argument to terminate the execution after completion of MOL-PRO jobs to check the convergence and smoothness of adiabatic PESs and NACTs. Subsequently, user can remove/refine unphysical data points prior to ADT calculation (using the numeric section).

Examples

1. For scattering case, one can run the following command in terminal,

```
adt mol -config molpro.config -atomfile atomfile.dat
-intpath 6
```

where 'molpro.config' and 'atomfile.dat' are the MOLPRO configuration file and atom information file, respectively. This will execute MOLPRO to calculate the *ab initio* data (adiabatic PESs and NACTs) over the specified grid points defined in the configuration file and subsequently, carry out the ADT calculation along path 6.

2. In order to terminate the program after running the MOLPRO jobs (before calculating the ADT quantities), one can pass the `-mo` argument,

```
adt mol -config molpro.config -atomfile atomfile.dat -mo
```

3. For spectroscopic case, user has to run the following command in terminal,

```
adt mol -config molpro.config -atomfile atomfile.dat
-geomfile geomfile.dat -freqfile frequency.dat
-wilsonfile wilson.dat -intpath 4
```

where 'geomfile.dat', 'frequency.dat' and 'wilson.dat' are the equilibrium geometry file, frequency file and Wilson matrix file, respectively. Other two files, 'molpro.config' and 'atomfile.dat' are the same as discussed above.

Limitations

- non-convergence of *ab initio* points originating from complete active space self-consistent field (CASSCF)/multi-reference configuration interaction (MRCI) methodology: The interpolation function in the python code cannot improve the output data if a large number of nuclear geometries are unrealistic.

- The swapping of *ab initio* computed NACTs due to the interchange of electronic states at CI point(s) are not considered in this software package due to molecule-specific characteristics.

- The *ab initio* adiabatic PESs can be computed only in `mrci` approach while calculating NACTs in finite difference method (`ddr`) (see Section 7.1).

6 Instructions for Input Files of Numeric Section

One can employ the numeric section of 'ADT' program package to solve the ADT equations along one dimensional as well as two dimensional contours of nuclear CS using adiabatic PES and NACT files. Those files can be provided either in HDF5 (.h5) or NumPy binary (.npy) or in text format. Detailed instructions related to the preparation of input files are given in the following paragraphs.

User can choose any one of the following coordinate systems to carry out ADT calculation:

- Spectroscopic systems:

  (i) 2D ADT calculation: plane polar counterparts (ρ and ϕ) of a pair of normal modes ($Q_i = \rho \cos \theta$ and $Q_j = \rho \sin \theta$)

  (ii) 1D ADT calculation: circular coordinate, ϕ for a fixed ρ

- Scattering systems:

  1. 2D ADT calculation:

     (a) hyperspherical coordinate (ρ and ϕ for a fixed θ or θ and ϕ for a fixed ρ)

     (b) Jacobi coordinate (q and ϕ) (see Figure 4)

  2. 1D ADT calculation: circular coordinate, ϕ for a fixed ρ and θ (hyperspherical) or ϕ for a fixed q (Jacobi).
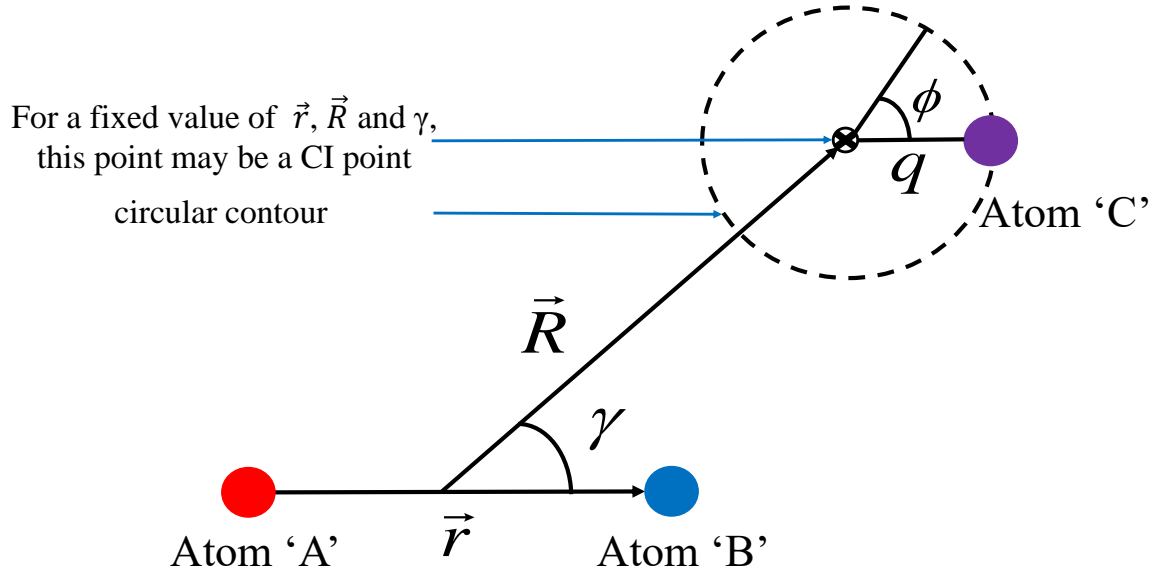
Figure 4: The above diagram briefly manifests the Jacobi coordinates of a triatomic system. The diatomic vector between Atom 'A' and Atom 'B' is represented as $\vec{r}$ with distance r, the vector from diatomic center of mass to the third atom 'C' is symbolized as $\vec{R}$ with distance R, and the angle between them is designated as γ. The circular contour is characterized by a radius (q) and an angle (ϕ) along which *ab initio* calculation is performed. In this process, two Jacobi coordinates $(R$ and $\gamma)$ are varying at a fixed value of r while performing such type of calculation.

6.1. Input files for 2D calculation

For the numeric section, three input files, namely, the adiabatic PES and components of NACTs are necessary.

- Input file for adiabatic PESs: The adiabatic PES input file must contain $2 + N$ number of columns (N is number of coupled electronic states): first two columns represent grid points of non-circular and circular coordinates, respectively and the rest of the columns contain the adiabatic PESs in increasing order of electronic states. User can supply scaled adiabatic energies for diabatization (scaled with respect to ground state minimum energy or asymptotic energy).

Example : The following table (Table 3) represents the arrangement of input file data points for adiabatic PESs of three electronic states ($N = 3$).

Table 3: Sample input file for adiabatic PESs

First Coordinate	Second Coordinate	Adiabatic Energy (ground state)	Adiabatic Energy (first excited state)	Adiabatic Energy (second excited state)
0.01	0.000000	0.002	0.123	0.125
0.01	3.141592	0.003	0.125	0.126
0.01	6.283185	0.003	0.124	0.126
0.02	0.000000	-0.001	0.119	0.121
0.02	3.141592	-0.002	0.118	0.121
0.02	6.283185	-0.002	0.118	0.122

- Input files for NACTs: Two NACT files (`-mfile` and `-nfile`) are required containing the components of NACTs for the non-circular and circular coordinates, respectively. There should be $2 + N * (N - 1)/2$ number of columns in each file, where the first two are non-circular and circular grids, respectively and the remaining ones signify one of the component of τ values within $N * (N - 1)/2$ pairs of electronic states. NACT components are represented as τ_ρ^{ij} or τ_θ^{ij} or τ_q^{ij} and τ_ϕ^{ij}. The columns of NACTs in the input files should be arranged in increasing order of first index (i) for a fixed second index (j) subjecting to the condition $i < j$ and then, by increasing the second index.

Example : For four coupled electronic states, elements of τ_ρ can be ordered as τ_ρ^{12}, τ_ρ^{13}, τ_ρ^{23}, τ_ρ^{14}, τ_ρ^{24} and τ_ρ^{34} as one of the input files (Table 4), where ϕ component of NACT will be given as τ_ϕ^{12}, τ_ϕ^{13}, τ_ϕ^{23}, τ_ϕ^{14}, τ_ϕ^{24} and τ_ϕ^{34} for the second input file of NACT (Table 5).

Table 4: Sample input file for ρ component of NACTs

ρ	ϕ	τ_ρ^{12}	τ_ρ^{13}	τ_ρ^{23}	τ_ρ^{14}	τ_ρ^{24}	τ_ρ^{34}
0.01	0.000000	0.002	0.123	0.125	0.020	0.103	0.105
0.01	3.141592	0.003	0.125	0.126	0.021	0.105	0.101
0.01	6.283185	0.003	0.124	0.126	0.023	0.107	0.100
0.02	0.000000	-0.001	0.119	0.121	0.019	0.109	0.100
0.02	3.141592	-0.002	0.118	0.121	0.023	0.108	0.101
0.02	6.283185	-0.002	0.118	0.122	0.025	0.108	0.102

Table 5: Sample input file for ϕ component of NACTs

ρ	ϕ	τ_ϕ^{12}	τ_ϕ^{13}	τ_ϕ^{23}	τ_ϕ^{14}	τ_ϕ^{24}	τ_ϕ^{34}
0.01	0.000000	0.005	0.103	0.150	0.020	0.130	0.121
0.01	3.141592	0.007	0.105	0.156	0.030	0.133	0.128
0.01	6.283185	0.009	0.107	0.156	0.030	0.137	0.130
0.02	0.000000	-0.006	0.105	0.155	0.021	0.133	0.121
0.02	3.141592	-0.009	0.107	0.159	0.027	0.137	0.125
0.02	6.283185	-0.011	0.108	0.160	0.029	0.139	0.127

6.2. Input Files for 1D Calculation

For 1D ADT calculation, one should supply two input files, namely, the adiabatic PESs and angular component of NACT (τ_ϕ) forming closed contour. There must be $1 + N$ and $1 + N * (N - 1)/2$ number of columns (N is number of coupled electronic states) in the adiabatic PES and NACT input files, respectively. The first column contains the grid points of the circular coordinate and the rest of the columns are exactly similar as the 2D input files for ADT calculation.

7 Instructions for Input Files of Molpro Section

In order to interface MOLPRO by the software package, 'ADT', user has to provide five input files for spectroscopic problems and two input files for scattering cases. The MOLPRO configuration and the atom information files are common for both cases. The details of such files are depicted below:

7.1. MOLPRO Configuration File

The MOLPRO configuration file, namely, 'molpro.config' is a standard INI configuration file, which contains six sections, namely, `molInfo`, `sysInfo`, `eInfo`, `nInfo`, `mInfo` and `gInfo` to specify all necessary keywords to run MOLPRO.

- `molInfo` contains the following three keywords:

  - `scrdir` (optional): path of scratch directory required during MOLPRO execution and storage of wavefunction (`.wfu`) files.

  - `memory` (optional): allotted memory for MOLPRO job

  - `processor` (optional): number of processors used for parallelization (default is `1`)

- `sysInfo` contains the following keyword:

  - `type`: type of the system for electronic structure calculation (`spec` for spectroscopic systems in normal mode coordinates, `scat_hyper` for scattering systems in hyperspherical coordinates and `scat_jacobi` for scattering cases in Jacobi coordinates)

  - `symmetry` (optional): any Abelian point group of the molecular species during *ab initio* calculation (default value is `nosym`).

- `eInfo` contains the following keywords for adiabatic PESs calculation:

  - `method`: method of adiabatic PESs calculation, namely, `multi` or `mrci`

  - `basis`: basis set

  - `cas`: active space

- – `electron`: number of electrons

- – `spin`: spin multiplicity ($2s$) of a molecular species

- – `charge`: charge of a molecular species

- – `state`: number of electronic states

- – `scale`(optional): nuclear geometry at which the ground adiabatic PES is set to zero.

    - ∗ spectroscopic cases: ρ,ϕ

    - ∗ scattering hyperspherical cases: ρ,θ,ϕ

    - ∗ scattering Jacobi cases : r,R,γ

- – `restricted` (optional): type of the initial hartree-fock calculation. `true` for `rhf` and `false` for `uhf`, default is `false`

- – `uhf_extra` (optional): any extra keyword for initial Hartree-Fock calculation

- – `multi_extra` (optional): extra keywords for `multi` calculation

- – `mrci_extra` (optional): extra keywords for `mrci` calculation

- • `nInfo` contains necessary keywords to calculate NACTs:

    - – `method`: method for NACT calculation, namely, `cpmcscf` or `ddr`

    - – `basis`: basis set

    - – `nact_extra` (optional): extra keywords regarding the NACT calculation.

- • `mInfo` (only for spectroscopic cases) contains the following keywords:

    - – `varying`: indexes of the variable normal modes. The indexes are assigned to the normal modes in the ascending order of the corresponding frequencies provided in the frequency file. [not similar as the standard nomenclature, e.g., indexes of Q_{4x} and Q_{4y} normal modes (degenerate components of asymmetric bending mode) for NO_3 radical will be `1,2` as the corresponding frequencies acquire the lowest values.]

- • `gInfo` contains the following keywords:

    - – For normal mode coordinates:

        - ▶ rho: ρ_{min}, ρ_{max}, $d\rho$. The grid is defined from ρ_{min} to ρ_{max} in stepsize $d\rho$, where $0 \leq \rho_{min} < \rho_{max}$ and $d\rho > 0$. ρ is expressed in mass-weighted dimensionless coordinates.

- ▶ `phi`: ϕ_{min}, ϕ_{max}, $d\phi$. The grid is defined from ϕ_{min} to ϕ_{max} in stepsize $d\phi$, where $\phi_{min} = 0$, $\phi_{max} = 2\pi$ and $d\phi > 0$. ϕ is in degrees.

  - ▶ `drho (dphi)`: geometry increment in ρ (ϕ) coordinate used for finite difference DDR method for NACT calculation. It is not required for NACT calculation using coupled-perturbed multi-configuration self-consistent field (CP-MCSCF) method.

- – For hyperspherical coordinate :

  - ▶ `rho`: ρ_{min}, ρ_{max}, $d\rho$. The grid is defined from ρ_{min} to ρ_{max} in stepsize $d\rho$, where $0 < \rho_{min} < \rho_{max}$ and $d\rho > 0$. ρ is expressed in atomic units.

  - ▶ `theta`: θ_{min}, θ_{max}, $d\theta$. The grid is defined from θ_{min} to θ_{max} in stepsize $d\theta$, where $0 < \theta_{min} < \theta_{max}$ and $d\theta > 0$. θ is in degrees.

  - ▶ `phi`:ϕ_{min}, ϕ_{max}, $d\phi$. The grid is defined from ϕ_{min} to ϕ_{max} in stepsize $d\phi$, where $\phi_{min} = 0$, $\phi_{max} = 2\pi$ and $d\phi > 0$. ϕ is in degrees.

  - ▶ `drho (dtheta/dphi)`: geometry increment in ρ (θ/ϕ) coordinate used for finite difference DDR method for NACT calculation. It is not required for NACT calculation using CP-MCSCF method.

- – For Jacobi coordinate:

  - ▶ `small_r`: r (diatom bond length) in atomic units.

  - ▶ `capital_r`: R (distance of the third atom from the center of the diatom) in atomic units.

  - ▶ `gamma`: γ (angle between $\vec{r}$ and $\vec{R}$ in degrees).

  - ▶ `q`: q_{min}, q_{max}, dq. The grid is defined from q_{min} to q_{max} in stepsize dq, where $0 < q_{min} < q_{max}$ and $dq > 0$. q is expressed in atomic units. One can provide a single value of q to calculate 1D PECs and NACTs along the ϕ grid.

  - ▶ `phi`: ϕ_{min}, ϕ_{max}, $d\phi$. The grid is defined from ϕ_{min} to ϕ_{max} in stepsize $d\phi$, where $\phi_{min} = 0$, $\phi_{max} = 2\pi$ and $d\phi > 0$. ϕ is in degrees.

  - ▶ `dq (dphi)`: geometry increment in q (ϕ) coordinate used for finite difference DDR method for NACT calculation. It is not required for NACT calculation using CP-MCSCF method.

The above illustration of 'molpro.config' is compactly presented in Table 6.

Table 6: Description and possible inputs of MOLPRO configuration file ('molpro.config')

Sections	Keywords	Description
molInfo	scrdir	path of scratch directory for wavefunction file, namely: /tmp/adtprogram
	memory	memory required for the job, e.g., 10,m
	processor	number of processors required for parallelization of the job, e.g., 2
sysInfo	type	spec or scat_hyper or scat_jacobi
	symmetry	symmetry of the molecule (default value: nosym) optional, e.g., x for H_3^+ (C_s point group: yz is the plane of symmetry)
eInfo	method	potential energy surface: multi or mrci
	basis	basis for energy calculation, namely: 6-31G** for NO_3
	cas	complete active space, e.g., occ,19;closed,11 for NO_3
	electron	number of electrons, namely: 31 for NO_3
	spin	spin multiplicity ($2s$), namely: 1 for NO_3
	charge	charge of a molecular species, namely: 1 for H_3^+
	state	number of electronic states, e.g., 5 for NO_3, 3,0 for H_3^+ (two values due to C_s symmetry)
	scale	spectroscopic: Q_i and $Q_j \Rightarrow \rho$ and ϕ, namely, 0.0,0.0 for NO_3 scattering (hyperspherical): ρ, θ and ϕ, namely, 25.0,90.0,175.4 for $F+H_2$ scattering (Jacobi): r, R and γ, namely, 1.0,24.0,0.0 for $F+H_2$
	restricted	true for rhf and false for uhf (default is false)
	uhf_extra	extra specification for uhf, namely: wf,2,1,0,1; for H_3^+
	multi_extra	extra specification for multi jobs, namely: maxiter,40 for NO_3
	mrci_extra	extra specification for mrci jobs, namely: maxiter,200,200 for $F+H_2$
nInfo	method	NACT calculation: cpmcscf for analytic gradient and ddr for numerical finite difference method.
	basis	basis set for NACT, e.g., 6-31G** for NO_3
	nact_extra	extra specification for NACT jobs, namely: accu=1.d-10 for NO_3
mInfo	varying	specification for pairwise normal modes (Q_i, Q_j), e.g., 1,2 (only required for spectroscopic calculation)
gInfo	rho	ρ grid for spectroscopic or scattering calculation in {start, end, step} format, namely: 0.1, 0.3, 0.1 or 1.0, 20.0, 0.25, respectively
	theta	θ grid (hyperspherical) for scattering calculation, e.g., 3, 90, 3
	phi	ϕ grid for spectroscopic/scattering calculation, e.g., 0, 360, 3
	small_r	diatomic bond length, e.g, 2.0
	capital_r	distance of third atom from diatomic center of mass, e.g, 1.73205
	gamma	angle between small_r and capital_r, e.g., 90.0
	q	radius of the circular path (see Figure 4) e.g., 0.5
	drho*	ρ spacing in ddr method, namely: 0.01
	dtheta*	θ spacing in ddr method, namely: 0.03
	dphi*	ϕ spacing in ddr method, namely: 0.03
	dq*	q spacing in ddr method, namely: 0.02

*The values of drho, dtheta, dphi and dq require adjustment based on the convergence of the calculated result.

Examples:

1. The following sample 'molpro.config' file presents necessary inputs for NO_2 radical, where adiabatic PESs are evaluated in CASSCF approach and analytic NACTs are calculated in CP-MCSCF method (see Table 7):

```
#Configuration file for the MOLPRO job of NO2


#Info for running molpro
[molInfo]
scrdir = /tmp/adtprogram
memory = 10,m


#for parallel molpro job
processor = 2


#Info of the molecular species
[sysInfo]
type = spec
#C_s point group (yz is plane of symmetry)
symmetry = x


#Info for energy calculation
[eInfo]
method = multi
basis = {spdf,n,cc-pvtz;spdf,o,cc-pvtz}
cas    = occ,11,3;closed,3,0
electron = 23
spin   = 1
charge = 0
state  = 2,0


#defining extra keyword required to run molpro for this system
#format should be same as in MOLPRO
multi_extra = maxiter,40
```

```
#defining scaling geometry for the adiabatic PESs
#ground state minima of NO2 w.r.t normal modes
scale = 0.0,0.0


#Info for NACT calculation
[nInfo]
method = cpmcscf
basis = {spdf,n,cc-pvtz;spdf,o,cc-pvtz}


#Info for varying modes
#put the indexes of the normal modes that will be varying
#The following two numbers represent Q_1 and Q_3 normal
#modes (bending and asymmetric stretching modes)
[mInfo]
varying = 1,3



#Put grid in format of <start>, <end>, <step>
#for spectroscopic put the 2D grid as rho and phi
[gInfo]
rho  = 0.2, 12.0, 0.2
phi = 0, 360, 1
#Qx = rho*cos(phi) and Qy = rho*sin(phi)
```

2. The following sample 'molpro.config' file represents necessary inputs for H_3^+ (1D PECs and NACTs calculation in Jacobi coordinate using C_s symmetry), where adiabatic energies are calculated in MRCI level of theory and analytic NACTs are evaluated in CP-MCSCF method (see Table 8):

```
#Configuration file for H3+ in Jacobi coordinates


#Info for running molpro
[molInfo]
scrdir = /tmp/adtprogram
memory = 10,m
```

```
#for parallel molpro job
processor = 2


#Info of the molecular species
[sysInfo]
type = scat_jacobi
#C_s point group (yz is plane of symmetry)
symmetry = x


#Info for energy calculation
[eInfo]
method = mrci
basis = cc-pv5z
cas    = occ,10,1;closed,0,0;
electron = 2
spin = 0
charge = 1
state = 3,0


#Here scaling of PECs is not necessary as Jacobi coordinate
#is used only for locating the conical intersections (CIs)
#small_r = 1.4, capital_r = 10.0, gamma = 90.0
#minima:H2 at the assymptote of H2+H+
#scale = 1.4,10.0,90.0


#defining extra keyword required to run molpro for this system
#format should be same as in MOLPRO
restricted = true
uhf_extra = wf,2,1,0,1;
multi_extra = maxiter,40


#Info for NACT calculation
#method should be either ddr or cpmcscf
[nInfo]
method = cpmcscf
```

```
basis = 6-311++G**


#put a fixed value of small_r, capital_r, gamma and q
#and provide a grid of phi
[gInfo]
small_r = 2.0
capital_r = 1.732050808
gamma = 90
#q must be smaller than capital_r
q = 0.5
#Put grid in format of <start>, <end>, <step>
phi = 0, 360,3
```

3. The following sample 'molpro.config' file depicts necessary inputs for H_3^+ (2D ADT calculation in hyperspherical coordinate using C_s symmetry), where adiabatic PESs are evaluated in MRCI level of theory and NACTs are computed in DDR method (see Table 8):

```
#Configuration file for H3+ in hyperspherical coordinates


#Info for running molpro
[molInfo]
scrdir = /tmp/adtprogram
memory = 10,m


#for parallel molpro job
processor = 2


#Info of the molecular species
[sysInfo]
type = scat_hyper
#C_s point group (yz is plane of symmetry)
symmetry = x


#Info for energy calculation
[eInfo]
```

```
method = mrci

basis = cc-pv5z

cas    = occ,10,1;closed,0,0

electron = 2

spin = 0

charge = 1

state = 3,0


#defining extra keyword required to run molpro for this system

#format should be same as in MOLPRO

restricted = true

uhf_extra = wf,2,1,0,1;

multi_extra = maxiter,40;ACCURACY,GRADIENT=1.d-4,

ENERGY=1.d-6,STEP=1.d-2


#defining scaling geometry for the adiabatic PESs

#ground state minima of H3+

#rho = 1.65, theta = 0.0, phi = 0.0 (D3h point)

scale = 1.65,0.0,0.0


#Info for NACT calculation

#method should be either ddr or cpmcscf

[nInfo]

method = ddr

basis = cc-pv5z


#Put grid in format of <start>, <end>, <step>

#for scattering put theta, phi as the 2d grid, with a given

#fixed value for rho

[gInfo]

rho   = 4.5

theta = 3,90,3

phi = 0, 360, 3


#following options are only required for ddr nact calculation

dtheta = 0.03
```

```
dphi = 0.03
```

4. The following sample 'molpro.config' file contains necessary MOLPRO keywords and other inputs for F+H$_2$ (2D ADT calculation in hyperspherical coordinate), where adiabatic PESs are generated in MRCI level of theory and analytic NACTs are calculated using CP-MCSCF method (see Table 8):

```
#Configuration file for F+H2 in hyperspherical coordinates


#Info for running molpro
[molInfo]
scrdir = /tmp/adtprogram
memory = 10,m


#for parallel molpro job
processor = 2


#Info of the molecular species
[sysInfo]
type = scat_hyper


#Info for energy calculation
[eInfo]
method = mrci
basis = aug-cc-pvtz
cas    = occ,10;closed,2;core,2
electron = 11
spin = 1
charge = 0
state = 3


#defining extra keyword required to run molpro for this system
#format should be same as in MOLPRO
multi_extra = maxiter,40
mrci_extra = maxiter,200,200


#defining scaling geometry for the adiabatic PESs
#minima of H2 at assymptotic F+H2 entrance channel
#rho = 25.0, theta = 90.0, phi = 175.4


scale = 25.0,90.0,175.4
```

```
#Info for NACT calculation
#method should be either ddr or cpmcscf
[nInfo]
method = cpmcscf
basis = 6-311++G**


#Put grid in format of <start>, <end>, <step>
#for scattering put theta, phi as the 2d grid, with a given
#fixed value for rho
[gInfo]
rho   = 5.0
theta = 0,90,5
phi = 0,360,4
```

5. The following sample 'molpro.config' file depicts necessary *ab initio* information for NO_3 radical, where adiabatic PESs are evaluated in CASSCF level of theory and analytic NACTs are calculated in CP-MCSCF method (see Table 7):

```
#Configuration file for the MOLPRO job of NO3


#Info for running MOLPRO
[molInfo]
scrdir = /tmp/adtprogram
memory = 10,m


#for parallel MOLPRO job
processor = 2


#Info of the molecular species
[sysInfo]
type = spec


#If symmetry is not specified, by default it will take 'nosym'
#C_s point group (yz is plane of symmetry)
#symmetry = x
```

```
#Info for energy calculation
[eInfo]
method = multi
basis = 6-31G**
cas    = occ,19;closed,11
electron = 31
spin = 1
charge = 0
state = 5


#defining extra keyword required to run molpro for this system
#format should be same as in MOLPRO
uhf_extra = accu,5
multi_extra = maxiter,40


#defining scaling geometry for the adiabatic PESs
#rho = 0.0, phi = 0.0
#ground state minima of NO3 PES w.r.t normal modes
scale = 0.0,0.0


#Info for NACT calculation
[nInfo]
method = cpmcscf
basis = 6-31G**
nact_extra = accu=1.d-10


#Info for varying modes
#put the indexes of the normal modes that will be varying
#The following two numbers represent the lowest frequency
#modes, but actually they are Q_4x and Q_4y normal modes
#(degenerate asymmetric bending modes)
[mInfo]
varying = 1,2
```

```
#Put grid in format of <start>, <end>, <step>
#for spectroscopic put the 2D grid as rho and phi
[gInfo]
rho = 0.1, 0.3, 0.1
phi = 0, 360, 3
#Qx = rho*cos(phi) and Qy = rho*sin(phi)
```

6. The following sample 'molpro.config' file represents necessary inputs for $C_6H_6^+$ radical cation, where adiabatic PESs are computed in CASSCF level of theory and analytic NACTs are evaluated in CP-MCSCF method (see Table 7):

```
#Configuration file for the MOLPRO job of C6H6+


#Info for running molpro
[molInfo]
scrdir = /tmp/adtprogram
memory = 100,m


#for parallel MOLPRO job
processor = 1


#Info of the molecular species
[sysInfo]
type = spec


#Info for energy calculation
[eInfo]
method = multi
basis  = 6-31G**
cas    = occ,21;closed,6
electron = 41
spin = 1
charge = 1
state = 5
```

```
#defining extra keyword required to run molpro for this system
#format should be same as in MOLPRO
uhf_extra = wf,41,1,1,1
multi_extra = maxiter,40;


#defining scaling geometry for the adiabatic PESs
#ground state minima of C6H6+ w.r.t normal modes
scale = 0.0,0.0


#Info for NACT calculation
[nInfo]
method = cpmcscf
basis = 6-31G**
nact_extra = accu=1.d-10


#Info for varying modes
#put the indexes of the normal modes that will be varying
#The following two numbers represent Q_16x and Q_16y normal
#modes (degenerate asymmetric stretching modes)
[mInfo]
varying = 23,24


#Put grid in format of <start>, <end>, <step>
#for spectroscopic put the 2D grid as rho and phi
[gInfo]
rho  = 0.1, 0.3, .1
phi = 0, 360, 3
#Qx = rho*cos(phi) and Qy = rho*sin(phi)
```

7. The following sample 'molpro.config' file depicts relevant inputs for 1,3,5-$C_6H_3F_3^+$ radical cation, where adiabatic PESs are computed in CASSCF level of theory and analytic NACTs are evaluated in CP-MCSCF method (see Table):

```
#Configuration file for the MOLPRO job of 1,3,5-C6H3F3+
```

```
#Info for running molpro
[molInfo]
scrdir = /tmp/adtprogram
memory = 100,m


#for parallel molpro job
processor = 1


#Info of the molecular species
[sysInfo]
type = spec


#Info for energy calculation
[eInfo]
method = multi
basis = cc-pvdz
cas    = occ,36;closed,27
electron = 65
spin = 1
charge = 1
state = 6


#defining extra keyword required to run molpro for this system
#format should be same as in MOLPRO
uhf_extra = wf,65,1,1,1;accu,5
multi_extra = maxiter,40


#defining scaling geometry for the adiabatic PESs
#ground state minima of 1,3,5-C6H3F3+ w.r.t normal modes
scale = 0.0,0.0


#Info for NACT calculation
[nInfo]
method = cpmcscf
basis = 6-31G**
```

```
#Info for varying modes
#put the indexes of the normal modes that will be varying
#The following two numbers represent Q_12x and Q_12y normal
#modes (degenerate C-C-C scissoring modes)
[mInfo]
varying = 16,17


#Put grid in format of <start>, <end>, <step>
#for spectroscopic put the 2D grid as rho and phi
[gInfo]
rho   = 0.02, 0.06, 0.02
phi = 0, 360, 3
#Qx = rho*cos(phi) and Qy = rho*sin(phi)
```

Table 7: Description of 'molpro.config' for NO_2, NO_3, $C_6H_6^+$ and 1,3,5-$C_6H_3F_3^+$ for the test runs

Sections	Keywords	NO_2	NO_3	$C_6H_6^+$	1,3,5-$C_6H_3F_3^+$
molInfo	scrdir	/tmp/adtprogram			
	memory	10,m		100,m	
	processor	2		1	1
sysInfo	type	spec			
	symmetry	x	—	—	—
eInfo	method	multi	multi	multi	multi
	basis	spdf,n,cc-pvtz; spdf,o,cc-pvtz	6-31G**	6-31G**	cc-pvdz
	cas	occ,11,3; closed,3,0;	occ,19; closed,11	occ,21; closed,6	occ,36; closed,27
	electron	23	31	41	65
	spin	1			
	charge	0	0	1	1
	state	2,0	5	5	6
	scale	0.0,0.0			
	uhf_extra	—	accu,5	wf,41,1,1,1	wf,65,1,1,1;accu,5
	multi_extra	maxiter,40			
nInfo	method	cpmcscf			
	basis	spdf,n,cc-pvtz; spdf,o,cc-pvtz	6-31G**	6-31G**	6-31G**
	nact_extra	—	accu=1.d-10	accu=1.d-10	—
mInfo	varying	1,3	1,2	23,24	16,17
gInfo	rho	0.2,12.0,0.2	0.1,0.3,0.1	0.1,0.3,0.1	0.02,0.06,0.02
	phi	0, 360, 1	0, 360, 3		

Table 8: Description of 'molpro.config' for H_3^+ and $F+H_2$ for the test runs

Sections	Keywords	H_3^+ (1D calculation)	H_3^+ (2D calculation)	$F+H_2$
molInfo	scrdir	/tmp/adtprogram		
	memory	10,m		
	processor	2	2	2
sysInfo	type	scat_jacobi	scat_hyper	
	symmetry	x		—
eInfo	method	mrci		
	basis	cc-pv5z		aug-cc-pvtz
	cas	occ,10,1;closed,0,0		occ,10;closed,2; core,2
	electron	2		11
	spin	0		1
	charge	1		0
	state	3,0		3
	scale	—	1.65,0.0,0.0	25.0,90.0,175.4
	restricted	true	true	—
	uhf_extra	wf,2,1,0,1;		—
	multi_extra	maxiter,40	maxiter,40;ACCURACY, GRADIENT=1.d-4, ENERGY=1.d-6, STEP=1.d-2	maxiter,40
	mrci_extra	—	—	maxiter,200,200
nInfo	method	cpmcscf	ddr	cpmcscf
	basis	6-311++G**	cc-pv5z	6-311++G**
gInfo	rho	—	4.5	5.0
	theta	—	3, 90, 3	0, 90, 5
	phi	0, 360, 3		0, 360, 4
	small_r	2.0	—	—
	capital_r	1.73205	—	—
	gamma	90.0	—	—
	q	0.5	—	—
	dtheta	—	0.03	—
	dphi	—	0.03	—

46

7.2. Atom Information File

- necessary file both for spectroscopic and scattering calculations

- First column: atomic symbols

- Second column: atomic masses in amu

Example: Sample file for NO_3 radical, 'atomfile.dat':

```
N 14.006700
O 15.999400
O 15.999400
O 15.999400
```

7.3. Equilibrium Geometry File

- necessary file for spectroscopic calculations

- contains the equilibrium geometry of the system

- Cartesian coordinates of the constituent atoms (in Angstroms) must be in a $n \times 3$ matrix form for a n atomic molecule

- order of atoms must be same as in atom information file

Example: Sample file for NO_3 radical, 'geomfile.dat':

```
 0.00000000     0.00000000     0.00000000
 1.23680091     0.00000000     0.00000000
-0.61840019    -1.07110018     0.00000000
-0.61840019     1.07110018     0.00000000
```

7.4. Frequency File

- necessary file for spectroscopic calculations

- contains frequencies of normal modes (in cm^{-1}) of a species arranged in ascending order (in a single row)

Example: Sample file for NO$_3$ radical, 'frequency.dat':

476.88	476.91	745.13	1047.52	1185.60	1185.85

7.5. Wilson Matrix File

- necessary file for spectroscopic systems

- provides the Wilson matrix of a molecular species

- the dimension of Wilson matrix will be $3n \times 3n - 5$ for linear molecules (vibrational DOFs: $3n - 5$) and $3n \times 3n - 6$ for non-linear ones (vibrational DOFs: $3n - 6$)

- columns are written as the increasing order of normal mode frequencies

- rows are sorted as per the order of atoms in atom information file

Example: Sample file for NO$_3$ radical, 'wilson.dat':

```
 0.4681243   0.0000000   0.0000000   0.0000102   0.0000000   0.7449584
 0.0000000  -0.4682375   0.0000000   0.0000000   0.7448872   0.0000000
 0.0000000   0.0000000   0.8798314   0.0000000   0.0000000   0.0000000
 0.3427462   0.0000000   0.0000000   0.5774593   0.0000000  -0.5394732
 0.0000000   0.6348331   0.0000000   0.0000000   0.0749380   0.0000000
 0.0000000   0.0000000  -0.2744083   0.0000000   0.0000000   0.0000000
-0.3903747  -0.4233146   0.0000000  -0.2887344  -0.2660973  -0.0787757
 0.4234383  -0.0983620   0.0000000  -0.4999028  -0.3859480  -0.2660775
 0.0000000   0.0000000  -0.2744055   0.0000000   0.0000000   0.0000000
-0.3903747   0.4233146   0.0000000  -0.2887344   0.2660973  -0.0787757
-0.4234383  -0.0983620   0.0000000   0.4999028  -0.3859480   0.2660775
 0.0000000   0.0000000  -0.2744055   0.0000000   0.0000000   0.0000000
```

The first three (3) rows signify 'x, y, z' coordinates of one (1) nitrogen atom and the following nine (9) rows represent Cartesian coordinates of three (3) oxygen atoms.

References

[1] Werner, H.-J.; et. al., MOLPRO, version 2010.1, a package of ab initio programs. 2010; http://www.molpro.net.