MULTIMODE Version 4.9 Release 0

****
IMPORTANT ------DECLARATION
****

MULTIMODE Version 4.9.0 is subject to Copyright, and is for use only by the group (or groups) to which it is supplied. It must not be distributed to groups outside the Institution from which the order was placed. Any changes made by the user should be communicated to us at the e-mail address given below. Failure to do this may render the user forfeit to any subsequent maintenance/support. I would be obliged if any new users of MULTIMODE would confirm by e-mail that they are willing to abide by this condition of sale.

Any publication that results from the use of MULTIMODE must contain a suitable reference:

'MULTIMODE' is a variational code for the calculation of rovibrational
energies of large polyatomic molecules; written by S. Carter,
with contributions from B.J. Braams, J.M. Bowman and N.C. Handy;
Refs
(a) S. Carter, S. J. Culik and J. M. Bowman, J. Chem. Phys. 107, 10458
(1997); (b) S. Carter and J. M. Bowman, J. Chem. Phys. 108, 4397 (1998);
(c) S. Carter, J. M. Bowman, and N. Handy, Theoretical Chem. Accnts. 100,
191 (1998); (d) S. Carter and N.C. Handy, J. Chem. Phys. 113, 987 (2000);
(e) R. Burcl, S. Carter and N.C. Handy, Chem. Phys. Lett. 380, 237 (2003)

****
END OF DECLARATION
****

1. Description of files and folders

(0) There are four folders in mm4.9.0: DOCS, src, Tutorial and TESTS

(1) The folder "src" contains the source code of MULTIMODE Version 4.9.0:

```
    blas.vscf.3.4.f
    contr.vscf.4.9.0.f
    curve.vscf.4.9.0.f
    diag.vscf.4.9.0.f
    drive.vscf.4.9.0.f
    eckart.vscf.4.9.0.f
    lanczos.vscf.4.9.0.f
    lap.vscf.3.4.f
    lib.vscf.3.4.f
    matinv.vscf.2.3.f
    memo.vscf.4.9.f
    molpro.vscf.4.9.0.f
    normal.vscf.4.9.0.f
    projec.vscf.4.9.0.f
    prop.vscf.4.9.0.f
    react.vscf.abinitio.f
    react.vscf.eckart.f
    react.vscf.noneckart.f
    rot.vscf.4.9.0.f
    timit.vscf.2.3.f
    user.vscf.4.9.0.f
    vib.vscf.4.9.0.f
```

(3) The folder DOCS contains three files: User.pdf, Input.pdf, and Guide.pdf.
    Guide.pdf covers the basic knowledge and features of MULTIMODE. User.pdf
    focuses on some user-specified routines. Input.pdf explains the meanings
    of all the input parameters.

(4) The folder "Tutorial" contains simple examples of VSCF/VCI calculations
    on a water monomer. We urge that users first run these examples as is.
    Then we recommend they run the same example without using the symmetry.

(5) Directory TESTS contains additional examples and it is further divided
    into directories FURAN, H2CS, HOOH, MOLPRO, RPH containing test inputs
    and outputs relating to specific features of MULTIMODE 4.9.0.

2. Instructions for the tutorial example

(1) Compiling the source code:
    All the source codes in the folder can be compiled using the Makefile
    provided. The "ifort" and "gfortran" compilers have been tested. After
    compilation, an executable called "h2o.x" is generated. You may get
    warnings during the compilation, but the program can produce the correct
    results. No external library routines are required with MULTIMODE 4.9.0.

    Note: The "user.H2O.f" is an additional, user supplied file. The most
    important subroutine in this file is the "GETPOT", which returns the
    potential energy of a configuration when the Cartesian coordinates are
    provided. To use MULTIMODE for other systems, the user need to provide
    their own "use.XXX.f" file similar to this one.

(2) Input and output files:
    MULTIMODE 4.9.0 always uses "fort.1" as the input file and "fort.2" as
    the output file. All the fort.1.VSCF.* and fort.1.VCI.* files inside the
    Tutorial folder are example inputs for VSCF or VSCF+VCI calculations of
    water. We added the suffix in file names in order to distinguish them.
    The users should rename them to fort.1 when running the program.

    The meaning of all the input parameters are described in the manual.

(3) Running the program:
    Put the executable (h2o.x) and the input file (fort.1) in the same folder.
    Also in the same folder, please create an empty folder called "temp".
    Temporary files will be stored in this temp folder.

    Then simply run the program as
    ./h2o.x
    and the output file "fort.2" is generated. Users can compare their results
    with the expected ones fort.2.XXX.