TeraChem User's Guide

Version 1.5

PetaChem, LLC
26040 Elena Road
Los Altos Hills, CA 94022

http://www.petachem.com

Contents

I Introduction

TeraChem is general purpose quantum chemistry software designed to run on Nvidia CUDA-enabled GPU architectures under a 64-bit Linux operating system. Some of TeraChem features include:

- Restricted, unrestricted, and restricted open shell Hartree-Fock and grid-based Kohn-Sham energy and gradient calculations
- Full support of s, p and d-type basis functions
- Various DFT functionals, including range-corrected and Coulomb attenuated functionals (BLYP, B3LYP, PBE, PBE0, ωPBE, ωPBEh, ωB97, ωB97x, camB3LYP, etc) and DFT grids (800 - 80,000 grid points per atom)
    - Static grid (single grid used for the entire calculation) and dynamical grid (multigrid) integration.
    - Empirical dispersion correction (DFT-D3 and DFT-D2)
- Geometry optimization (L-BFGS, Conjugate gradient, Steepest descent)
    - The optimization can be carried out either in Cartesian or internal coordinates as specified in the start file (all input geometries are provided in Cartesians). The Cartesian → internal → Cartesian coordinate transformation is performed automatically whenever required.
    - Constrained optimization with frozen atoms, constrained bond lengths, angles, and dihedrals.
- Transition state search (Nudged elastic band) in internal and Cartesian coordinates
- Ab initio molecular dynamics (NVE, NVT ensembles)
    - Reversible Born-Oppenheimer dynamics
    - Spherical boundary conditions
- Support of multiple-GPU systems
- Single/Dynamical/Double precision accuracy
- QM/MM treatment of surrounding water molecules using TIP3P force field[1]
- Natural bond orbital analysis through integration with NBO6
- Polarizabilities for HF and closed-shell DFT methods

See Table 1 for a complete list of TeraChem capabilities.

---

[1] W. L. Jorgensen, J. Chandrasekha, J. D. Madura, R. W. Impey and M. L. Klein, J. Chem. Phys. **79** 926 (1983).

Obtaining TeraChem

To purchase a copy of TeraChem visit http://shop.petachem.com. Pricing information is available at http://www.petachem.com/pricing.html.

Citing TeraChem

Any published work that utilizes TeraChem shall include the following reference:

I.S. Ufimtsev and T.J. Martinez, Quantum Chemistry on Graphical Processing Units. 3. Analytical Energy Gradients and First Principles Molecular Dynamics, *J. Chem. Theory Comput.*, 2009, **5**, p2619.

Work which uses the geometry optimization or transition state finding utilities shall include the following reference:

J. Kästner, J.M. Carr, T.W. Keal, W. Thiel, A. Wander and P. Sherwood, DL-FIND: An Open-Source Geometry Optimizer for Atomistic Simulations, *J. Phys. Chem. A*, 2009, **113**, p11856.

Whenever dispersion corrections are used, the following references should be cited:

S. Grimme, J. Antony, S. Ehrlich, and H. Krieg, *J. Chem. Phys.*, 2010, **132**, p154104.

S. Grimme, S. Ehrlich, and L. Goerigk, *J. Comput. Chem.,* 2011, **32**, p1456.

Acknowledgements

This software was developed by Ivan Ufimtsev and Todd Martinez at the University of Illinois at Urbana-Champaign and PetaChem, LLC. The authors would like to especially thank Nathan Luehr for contribution of the GPU accelerated code for construction of numerical DFT grids and Alexey Titov for contribution of GPU accelerated code for electron repulsion integrals involving d functions. Geometry optimization and transition state search calculations use the DL-FIND library created by Johannes Kästner at Stuttgart University. Hartree-Fock and DFT dispersion correction code is used with permission from Stefan Grimme at the University of Münster. TeraChem uses standard Gaussian basis sets available at EMSL website http://www.emsl.pnl.gov/forms/basisform.html

II Getting started

System requirements

This version of TeraChem was compiled and tested under 64-bit RedHat Enterprise Linux 5.3 operating system running on Intel Core2 quad-core and Intel Xeon 5520 dual quad-core CPU machines. An Nvidia compute capability 1.3 (Tesla C1060 or similar) or higher (i.e. Tesla C2050 or similar) graphics card is required to run the program. Please refer to the CUDA Programming Guide at http://www.nvidia.com/object/cuda_develop.html for the most current list of Nvidia GPU's that meet this requirement. A CUDA driver (270.41.19 or later) must be installed on the system as well as v4.0 of the CUDA Toolkit. Details on how to obtain and install the CUDA driver are provided below.

The use of Nvidia GeForce 5xx and Tesla C2075 series GPUs is not recommended. For more details, refer to the discussion thread which can be found on the PetaChem forum: http://petachem.com/forum/index.php?topic=95.0 The binary image `terachem.geforce` implements some software workarounds which may improve the experience on these GPUs and we recommend that it be used (in place of the usual binary image `terachem`) if one needs to run TeraChem on these GPUs.

Because the binary file is linked against the Intel MKL library, it is recommended to run TeraChem on Intel-based workstations.

The amount of CPU RAM needed depends on the size of the molecules that will be studied. If the molecules of interest are relatively small (less than 500 atoms), the usual 8Gb or 16Gb configuration is acceptable. For very large molecules (in excess of 10,000 basis functions), CPU RAM will often be a limiting factor. For example, molecules with 25,000 basis functions require almost 70GB of CPU memory.

Installation

First, obtain and install the latest CUDA driver (TeraChem 1.5 requires 270.41.19 or later) and the CUDA 4.0 toolkit (choose the one appropriate for your Linux OS, e.g. RedHat Enterprise Linux 5.5 for RHEL 5.5 or CentOS 5.5) available for download free of charge at

```
http://developer.nvidia.com/cuda-toolkit-40#Linux
```

Installation of the CUDA driver must be performed by a user with "root" permission, i.e. the superuser. Check with your local system administrator first, since adequate CUDA drivers may already be installed. Make sure you select the 64-bit Linux operating system. You will need to install the CUDA driver and the CUDA toolkit (ver. 4.0). The CUDA SDK is not required (but may be installed for other purposes if desired). After downloading the driver package, shut down the X server by typing

```
init 3
```

Then launch the driver binary, and follow the instructions. Unpack the tc.tar archive using the following command (in a temporary directory which you may later remove):

```
tar xvf tc.tar
```

Run the install script by typing

```
./install
```

This script will verify that your machine has a suitable graphics card, verify that you accept the license terms, and install the software in a location of your choosing (the "TeraChem installation directory" or `instdir` in the following). The script will ask you for the location of the CUDA libraries (installed when the CUDA toolkit is installed, see above). It will also create a script that sets the appropriate environment variables. The `install` script will also put a temporary license file in place so you can begin using TeraChem immediately. However, this temporary license file is time-limited, so you will want to obtain a permanent license file. The `install` script ends with a form suitable for emailing to help@petachem.com:

-------------BEGIN HERE------------------

Institution: ______________________

Ordered By: _______________________

MAC: 003048DB1D7E

IP: 171.64.125.189

---------------END HERE------------------

You can regenerate this at any time by typing

```
./machid
```

in the TeraChem installation directory. Fill in the "Institution" and Ordered By" fields and email to help@petachem.com. When we receive this from you, we will send you a permanent license file (which should be saved as `license.dat` in the TeraChem installation directory). `TeraChem` environment variable indicates where the licensing file can be found, i.e. `$TeraChem/license.dat`.

Running sample jobs

TeraChem package contains several sample jobs located at

```
instdir/TeraChem/terachem/tests
```

After installation, `cd` to `caffeine` directory and run TeraChem by typing

```
source instdir/TeraChem/SetTCVars.sh
instdir/TeraChem/terachem start.sp
```

where `instdir` is the installation directory you chose during the install (defaults to your home directory) and `start.sp` is the name of a TeraChem input file. Note that the environment variable `TeraChem` is set by "source'ing" `SetTCVars.sh`. This is needed in order for TeraChem to locate the license and basis set library files.

The `start.sp` contains the required parameters of the job (including the filename of the file which contains the atomic coordinates for the molecule of interest). Most of the parameters have default values. The complete list of parameters available in this version is presented in Table 1. An example of the configuration file used for single point energy calculations of caffeine with the BLYP functional, DFT-D dispersion corrections, and the 6-31G basis set is:

```
# Job: Single point energy of caffeine
```

```
#
# basis set
basis       6-31g
# coordinates file
coordinates caffeine.xyz
# molecule charge
charge      0
# SCF method (rhf/blyp/b3lyp/etc...): DFT-BLYP
method      blyp
# add dispersion correction (DFT-D)
dftd        yes
# type of the job (energy/gradient/md/minimize/ts): energy
run         energy
end
```

All lines beginning with the '#' character are considered comments and are ignored by TeraChem. There is no requirement on the line ordering in the start file except that the last line should be '`end`'.

Below is the output from this example job. The program first lists the parameters followed by all GPUs used in the job. Each GPU has its amount of memory and compute capability printed next to it. The program then attempts to predict the maximum size of the molecule that can be handled on the current machine based on the amount of available CPU and GPU memory. *Note that the recommended maximum size of the system is approximate.* The SCF procedure, which includes the DIIS error (the maximum component of the DIIS error vector), integrated number of electrons, exchange-correlation energy, SCF energy, and the total time elapsed per iteration, completes the program's output.

```
           ***********************************************************
           *                     TeraChem v1.44                      *
           *                  Hg Version: 901c2a55a3e3+              *
           *           Chemistry at the Speed of Graphics!           *
           ***********************************************************
           * This program may only be used in connection with        *
           * a valid license from PetaChem, LLC. Use of this program *
           * or results thereof indicates acceptance of all terms    *
           * and conditions stated in the license and that a valid   *
           * license agreement between the user and PetaChem, LLC    *
           * exists. PetaChem, LLC does not warrant the correctness  *
           * of results or their suitability for any purpose.        *
           * Please email bugs, suggestions, and comments to         *
           *                  help@petachem.com                      *
           *                                                         *
           *  Compiled by ufimtsev     Fri Mar 11 19:28:58 PST 2011  *
           ***********************************************************

     Job started   Sat Mar 12 23:05:03 2011
     On MTZ08 (available memory: 66614 MB)

            ********************** License Check **********************
            * Use license from: /home/toddmtz/TeraChem/license.dat
            * Available Host id: 00044B01C3C8
            * License expires: 2011-01-10
            ********************** License OK *************************

    XYZ coordinates caffeine.xyz
    Orbitals will be written to ./scr/orbitals.log every 1000000000 time step
    Spin multiplicity: 1
    Using DIIS algorithm to converge WF
    WF convergence threshold: 3.00e-05
    Maximum number of SCF iterations: 100
    Incremental Fock matrix formation
    Will switch to conventional Fock is diffuse functions are detected
    PRECISION: DYNAMIC
    X-matrix tolerance: 1.00e-04
    Method: BLYP with dispersion correction
```

```
  Becke 1988 exchange functional:      1.0
  Lee-Yang-Parr correlation functional: 1.0
  _________________________________

 |         DFTD3 V2.0 Rev 1        |
 | S.Grimme, University Muenster   |
 | Fri Dec 17 17:08:12 CET 2010    |
 |        (Hawaii Version)         |
  _________________________________

 Please cite DFT-D3 work done with this code as:
 S. Grimme, J. Antony, S. Ehrlich and H. Krieg,
 J. Chem. Phys, 132 (2010), 154104.
 If used with BJ-damping cite also
 S. Grimme, S. Ehrlich and L. Goerigk,
 JCC, submitted.
 For DFT-D2 the reference is
 S. Grimme, J. Comput. Chem., 27 (2006), 1787-1799

DFT grid type: 1
Using dynamic DFT grids.
Initial guess generated by maximum overlap

******************************************
**** SINGLE POINT ENERGY CALCULATIONS ****
******************************************

using 3 out of 3 CUDA devices
   Device 0:  GeForce GTX 480, 1535MB, CC 2.0 -- CPU THREAD 0
   Device 1:  GeForce GTX 480, 1535MB, CC 2.0 -- CPU THREAD 1
   Device 2:      Tesla C1060, 4095MB, CC 1.3 -- CPU THREAD 2
---------------------------------------------------------------
 CPU Memory Available: 9055.71 MegaWords
 GPU Memory Available: 191.96 MegaWords
 Maximum recommended basis set size: 9200 basis functions
 (limited by GPU memory)
---------------------------------------------------------------

Basis set:       6-31g
Total atoms:     24
Total charge:    0
Total electrons: 102 (51-alpha, 51-beta)
Total orbitals:  146
Total AO shells: 90 (62 S-shells; 28 P-shells; 0 D-shells; 0 F-shells; 0 G-shells)
The spin state is singlet
Nuclear repulsion energy (QM atoms): 933.356431723821 a.u.

0: CUBLAS initialized, available GPU memory: 1106MB
1: CUBLAS initialized, available GPU memory: 1106MB
2: CUBLAS initialized, available GPU memory: 4040MB
Setting up the DFT grid...
time to set the grid = 0.01 s
DFT grid points: 23506 (979 points/atom)
Setting up the DFT grid...
time to set the grid = 0.02 s
DFT grid points: 65912 (2746 points/atom)

                      *** Start SCF Iterations ***
 Iter     DIIS Error       Energy change      Electrons        XC Energy          Energy        Time(s)
---------------------------------------------------------------------------------------------------------
         >>> Purifying P... IDMP = 1.51e-14 <<<
THRESPDP set to 1.00e+00
                >>> SWITCHING TO GRID 1 <<<
   1     0.3478153024    -677.4494252222   101.9995046055   -90.0198981966    -677.4494252222     0.80
                >>> SWITCHING TO GRID 0 <<<
   2     0.9764151815      +6.4286938544   101.9966690298   -93.6326514035    -671.0207313678     0.22
   3     0.3117654963      -7.9655469155   101.9979491111   -93.2843271878    -678.9862782834     0.19
   4     0.1680723502      -0.8674289098   101.9978465991   -91.6341017607    -679.8537071932     0.29
   5     0.1886208597      -0.0178432775   101.9975267409   -92.0886995503    -679.8715504707     0.18
   6     0.0524384204      -0.1104283837   101.9977508683   -92.0994119259    -679.9819788543     0.20
   7     0.0128652845      -0.0112165349   101.9977297818   -92.0548280000    -679.9931953892     0.19
   8     0.0057005743      -0.0010061156   101.9977327019   -92.0454367265    -679.9942015049     0.18
                >>> SWITCHING TO GRID 1 <<<
   9     0.0041083048      -0.0001794039   101.9996596555   -92.0499852070    -679.9943809088     0.37
  10     0.0006777352      -0.0000808676   101.9996596199   -92.0536204430    -679.9944617764     0.30
THRESPDP set to 3.56e-03
  11     0.0003222704      +0.0000139354   101.9996620014   -92.0544149668    -679.9944478409     0.48
  12     0.0003152979      -0.0000005484   101.9996619866   -92.0540590401    -679.9944483893     0.42
  13     0.0001861062      -0.0000002483   101.9996620977   -92.0539773303    -679.9944486376     0.42
  14     0.0000253330      -0.0000002257   101.9996620305   -92.0541017937    -679.9944488633     0.47
---------------------------------------------------------------------------------------------------------
```

```
FINAL ENERGY: -679.9944488633 a.u.
CENTER OF MASS: {0.008017, 0.006143, 0.000066} ANGS
DIPOLE MOMENT: {3.583599, -0.524503, 0.000038} (|D| = 3.621780) DEBYE
Writing out molden info
Total processing time: 4.88 sec

 Job finished: Sat Mar 12 23:05:19 2011
```

III TeraChem I/O file formats

Input files

In addition to the start file containing job parameters, TeraChem requires two other files to start a job: coordinates and basis set.

Atomic coordinates

The coordinates file fed to the `coordinates` parameter should be either in XMol or PDB format. In an XMol coordinates file, the first line specifies the number of atoms, and the second line provides a description of the system (it can be left blank). Atomic coordinates are listed starting from the third line. All coordinates are either in Angstroms (default) or Bohrs (this can be specified in the start file). Here is an example coordinates file for a hydrogen molecule:

```
2
Hydrogen Molecule - Xmol format
H   0.0      0.0   0.0
H   0.7      0.0   0.0
```

Some jobs (for example, transition state search using NEB method) require several sets of coordinates (frames). In this case all frames should be listed in the coordinates file one by one, i.e.

```
2
Hydrogen Molecule - Xmol format frame 1
H   0.0      0.0   0.0
H   0.7      0.0   0.0
2
Hydrogen Molecule - Xmol format frame 2
H   0.0      0.0   0.0
H   0.8      0.0   0.0
```

Note that there should be no blank lines between individual frames.

The PDB format often used for protein molecules is also supported and will be automatically assumed if the filename for the coordinates ends in '`.pdb`'. More details on PDB file format are available at http://www.wwpdb.org/docs.html

Basis set file format

TeraChem supports standard atom-centered Gaussian type basis sets such as those available at EMSL website http://www.emsl.pnl.gov/forms/basisform.html. The basis set information used by TeraChem is provided by a set of files in `basis` directory. This directory should be located as `$TeraChem/basis`, where the environment variable `TeraChem` is set to the TeraChem installation directory by default. All basis sets are listed in individual files.

For every atom in a basis set, the first four lines specify the atom type, number of core orbitals, number of valence orbitals, and the total number of contracted basis functions centered on the atom. Consider an iron atom in STO-3G basis set.

```
ATOM Fe
COR 9
VAL 4
BFS 19
```

Here, `Fe` is an alias of the atomic nuclear charge, `Z`, and `BFS` is the total number of atomic shells multiplied by the number of basis functions per shell (1 for `S-`, 3 for `P-`, 6 for `D-shell`, etc). The next block contains coefficients of atomic orbitals (AOs) that need to be precalculated separately for each atom and each basis set. Below are 2 out of 13 AOs of iron in STO-3G basis set.

```
 9.92667130e-01   1.93498346e-02  -1.55903797e-02  -7.99232061e-04  -0.00000000e+00
-0.00000000e+00  -0.00000000e+00  -0.00000000e+00  -0.00000000e+00  -0.00000000e+00
-0.00000000e+00  -0.00000000e+00  -0.00000000e+00  -0.00000000e+00  -0.00000000e+00
-0.00000000e+00   5.90570755e-03   5.90570755e-03   5.90570755e-03
                          ••••••••••••••••••••••••••
 0.00000000e+00   0.00000000e+00   0.00000000e+00  -2.44836578e-11   0.00000000e+00
 0.00000000e+00   0.00000000e+00   0.00000000e+00   0.00000000e+00   0.00000000e+00
 0.00000000e+00   0.00000000e+00   0.00000000e+00  -5.06687307e-01  -2.83366410e-01
 4.39883475e-01  -1.02744686e-02  -5.88180817e-01   5.98455286e-01
```

This information is required to construct the initial guess as described in

J.-M. Langlois, T. Yamasaki, R.P. Muller, and W.A. Goddard III, Rule-Based Wave Function for Generalized Valence Bond Theory, *J. Phys. Chem.*, 1994, **98**, p13498.

In the AO block, core orbitals (`COR`) are listed first, followed by valence orbitals (`VAL`). Each atomic orbital thus is represented by `BFS` coefficients. Therefore, the total number of AO coefficients is equal to `BFS*(COR+VAL)`. Basis shells, including the shells' type, number of primitive functions and corresponding contraction coefficients and exponents are listed at the end of the atom description.

```
S 3
   1447.4004110            0.1543289673
    263.6457916            0.5353281423
     71.35284019           0.4446345422
            ••••••••••••••••••••
P 3
    111.9194891            0.1559162750
     26.00768236           0.6076837186
      8.458505490          0.3919573931
            ••••••••••••••••••••
D 3
      6.411803475          0.2197679508
      1.955804428          0.6555473627
      0.754610151          0.2865732590
```

It should be noted that the ordering of basis functions in the AO expansion used to construct the initial guess must be the same as in the GTO block above. Within P-type shells, the basis functions are ordered as X, Y, Z, and within D-type shells as XY, XZ, YZ, XX, YY, ZZ.

It is straightforward to mix different basis sets for the same molecule. To do that, one needs to create a separate basis file which will contain all required basis functions. Here is an example calculation of a hydrogen molecule with STO-3G and 3-21G basis set used for the first and the second hydrogen atom, respectively:

```
ATOM  H1
COR 0
VAL 1
BFS 1
 1.00000000E+00

S  3
             3.4252509    0.154328967295
             0.6239137    0.535328142282
             0.1688554    0.444634542185

ATOM  H2
COR 0
VAL 1
BFS 2
 3.73407177E-01 7.17324343E-01

S  2
             5.4471780    0.156284978695
             0.8245472    0.904690876670
S  1
             0.1831916    1.000000000000
```

When mixing basis sets, the atom names in the coordinates file should be consistent with those provided in the basis files. Here is an example of the corresponding XYZ file.

```
2
Hydrogen Molecule
H1  0.0      0.0   0.0
H2  0.7      0.0   0.0
```

Note that a user can use a local copy of the basis directory if desired by setting the `TeraChem` environment variable appropriately. In that case, one should also ensure that a copy of the `license.dat` file exists in `$TeraChem`.

Output files

In addition to the information displayed on the screen, TeraChem creates several output files.

All calculations:

`scr/c0` (`scr/ca` and `scr/cb` in UHF and UKS jobs) – the converged WF binary file containing the MO coefficients `C[i][j]` where `i` (row) is the MO and `j` (column) is the AO basis function index. This file can be used as initial WF guess in subsequent calculations.

`scr/prjct` (`scr/prjcta` and `scr/prjctb` in UHF and UKS jobs) – the converged WF projected onto another (usually, larger) basis set. These files are generated by `project` jobs and used as an efficient initial guess.

`scr/orbitals.log` – the canonical MO orbitals in GAMESS format. This file name can be modified by the `orbitals` parameter. Because the orbitals require much disk space, they can be written every n^{th} MD step, specified by `orbitalswrtfrq`. The orbitals can be visualized by VMD. Note that the only meaningful information contained in this file is the MO coefficients. Everything else is printed for VMD to parse the file correctly.

`scr/molden.molf` – the canonical MO orbitals in Molden format.

`scr/charge.xls` – a tab-separated file containing Mulliken atomic charges.

`scr/oe.xls` – MO energies.

Geometry Optimization/Transition State Search:

`scr/optlog.xls` – a tab-separated file containing 7 columns of which only one (the first one) is currently used. The first column contains the SCF energy during geometry optimization or transition state search.

`scr/optim.xyz` – geometry optimization or transition state search (depending on the job type) trajectory file in the XMol format. The trajectory can be visualized by VMD.

Transition State Search:

`scr/neb_n.xyz` – an XMol file containing trajectory of the n^{th} NEB image. The last image (i.e. the `neb_10.xyz` if `min_image` equals 10) is the actual transition state that that is also stored in `optim.xyz` file.

`scr/nebinfo` – contains energies of all (`min_image-1`) NEB images along the converged NEB path.

`scr/nebpath.xyz` – contains XYZ coordinates of all (`min_image-1`) NEB images along the converged NEB path.

MD simulation:

`scr/log.xls` – a tab separated file containing 7 columns: 1) SCF energy, 2) currently not in use, 3) Kinetic energy, 4) Temperature, 5) Total energy (SCF + Kinetic), 6) HOMO energy, 7) LUMO energy. All energies are in Hartree and the temperature is in degrees Kelvin. In NVT dynamics, the Total energy does not include the contribution from the damping force, and thus should not be conserved.

`scr/coors.xyz` – the MD trajectory geometry file in the XMol format. The trajectory can be visualized by VMD.

`scr/vel.log` – contains atomic velocities along in the MD trajectory. The format is similar to that of `scr/coors.xyz` except there is a blank line separating each set of velocities.

`scr/restart.md` – a binary MD restart file containing all information (wavefunction, coordinates, velocities, etc) required to restart an MD job. By default, this information is stored at every 100^{th} MD iteration. A user can change it by specifying `restartmdfreq` parameter in the start file.

IV More details on some TeraChem capabilities

Wavefunction projection

Sometimes, especially when transition metals or diffuse basis functions are present in a system, the SCF procedure does not converge due to insufficiently accurate initial guess. In such cases, it often helps to converge the wavefunction using a smaller basis set (mini, sto-3g, etc) with no diffuse functions, then project the solution onto the desired basis set and start another SCF procedure using the projected wavefunction as initial guess. Below is an example of such `project` job. Here, the calculations are performed in a smaller (`sto-3g`) basis set, and the converged wavefunction is projected onto `6-31g**` basis set.

```
run                 project
basis               sto-3g
projectbasis        6-31g**
charge              0
coordinates         water.xyz
end
```

The projected wavefunction is stored in `scr/prjct` file.

MD with spherical boundary conditions

In an MD simulation, it is possible to impose spherical boundary conditions to prevent "evaporation" events or constrain a system to a given density. The spherical boundary conditions are provided in the form of a sum of two harmonic terms,

$$U_{constr}(r) = k_1\left(\left(r - R_{center}\right) - R_1\right)^2 + k_2\left(\left(r - R_{center}\right) - R_2\right)^2,$$

where R_{center} is the center of mass of the system calculated for the first MD frame and then fixed. By default, k_1 is set to 10.0 kcal/(mol Å^2) and k_2 is set to zero. The simplest way to impose spherical boundary conditions is to set

```
mdbc            spherical
md_density      1.0
```

in the start file. `md_density` specifies the density of the system in g/mL used to automatically adjust R_1. If `md_density` is not provided, R_1 needs to be set explicitly using the `md_r1` keyword. Note that constraining forces are not applied to hydrogen atoms.

QM/MM Functionality

Limited QM/MM functionality is available in this release. Only water molecules are available for MM treatment and they will be modeled using the TIP3P force field. One signals that a QM/MM calculation is desired by including the `qmmm` keyword in `inputfile`. The keyword specifies the location of a file with coordinates for the desired MM water molecules. Note that the coordinates of the MM waters are *not* in the `coordinates` file. The water molecule positions must appear in a specific order – grouped by water molecule and in the order O,H,H. We provide an example for one QM water molecule surrounded by one MM water molecule.

`inputfile:`

```
basis       6-31g
qmmm        mmwater.xyz
coordinates qmwater.xyz
run         minimize
end
```

qmwater.xyz:

```
3

 O          6.144353       -0.788526       -6.483525
 H          6.378922       -1.564817       -5.989856
 H          6.475062       -0.043630       -5.995619
```

mmwater.xyz:

```
3

 O          6.450776        0.962914       -3.536427
 H          6.338156        1.514355       -2.771897
 H          6.519847        1.524502       -4.303300
```

Integration with NBO6

This release of TeraChem is interfaced with the NBO6 package of Weinhold and coworkers.[2] NBO deletions are not currently supported, nor are analyses which require the matrix elements of the dipole operator. However, many of the usual natural bond orbital (NBO) and natural population analyses (NPA) are supported. Setting the `nbo` keyword in the `startfile` to `npa` or `full` will give natural population analysis or full NBO/NPA analysis. You may also use advanced keywords (documented in the NBO manual) if desired. In this case, the `nbo` keyword should be set to `advanced` and the `startfile` should contain a `$nbo` group as documented in the NBO manual. An example of this usage of advanced keywords for CH_3NH_2 is provided in `TeraChem/tests/nbo`. Note that the `NBOEXE` environment variable needs to point to the NBO6 executable file in order for successful integration with TeraChem. This is currently set in the `SetTCVars.sh` script. The NBO6 executable is included with the TeraChem distribution and resides in the TeraChem directory.

Geometry Optimization and Transition State Search

The `instdir/tests/sp` directory contains a simple configuration file (`start.go`) used for geometry optimization of a spiropyran molecule:

```
# basis set
basis       6-31g
# coordinates file
coordinates sp.xyz
# molecule charge
charge      0
# SCF method (rhf/blyp/b3lyp/etc...): RHF
method      rhf
```

---

[2] http://www.chem.wisc.edu/~nbo5

```
# type of the job (energy/gradient/md/minimize/ts): geometry
# optimization
run         minimize
end
```

The optimization is triggered by the '`minimize`' keyword. Note that the `sp.xyz` file in fact contains two sets of coordinates (frames) one by one. In geometry optimization jobs, only the first frame is taken into account while the others (if any) are ignored. The second frame, however, is required by transition state search jobs and represents the second NEB endpoint in NEB calculations. The TS search is triggered by the '`ts`' keyword, i.e. file (`start.ts`)

```
# basis set
basis       6-31g
# coordinates file
coordinates sp.xyz
# molecule charge
charge      0
# SCF method (rhf/blyp/b3lyp/etc...): RHF
method      rhf
# type of the job (energy/gradient/md/minimize/ts): TS search
run         ts
end
```

`$constraints ... $end` group in the configuration file allows one to impose geometrical constraints during optimization (frozen atoms, fixed bond lengths, angles, and dihedrals). `instdir/tests/constraints` directory contains two example start files for constrained geometry optimization jobs.

Constrained geometry optimization can be performed in any type of coordinates except for jobs with frozen atoms which cannot be performed in DLC coordinates. Below is an example of a job configuration start file. Note that all constraints should be listed in separate lines, and enumeration of atoms begins from 1. Constrained coordinates will be constrained to the value they have in the starting geometry.

```
# basis set
basis       6-31g
# coordinates file
coordinates C10H22.inp
# molecular charge
charge            0
# optimize geometry
run   minimize
end

$constraints
# bond connecting atoms 32 and 29
bond        32    29
# 32-29-26 angle
angle             32    29    26
# 32-29-26-27 dihedral
dihedral    32    29    26    27
$end
```

Finally, it is straightforward to freeze all hydrogen or non-hydrogen atoms by `atom hydrogens` or `atom heavy`, respectively.

V Trajectory visualization

To create a simple MD trajectory, you can use the example in `instdir/tests/caffeine`. The configuration file `start.md` is:

```
# basis set
basis             6-31g
# coordinates file
coordinates       caffeine.xyz
# molecule charge
charge            0
# SCF method (rhf/blyp/b3lyp/etc...): DFT-BLYP
method            blyp
# type of the job (energy/gradient/md/minimize/ts): MD
run               md
# number of MD steps
nstep             10
# dump orbitals every MD step
orbitalswrtfrq    1
end
```

The only difference between this file and the file used for the single point calculations are the `nstep` and `orbitalswrtfrq` parameters, which are now set to 10 and 1, respectively. The keyword `orbitalswrtfrq` ensures that the molecular orbitals will be written at every MD step (out of 10, total).

The TeraChem output coordinates file format is compatible with VMD. To open the trajectory file, go to VMD → File → New Molecule. Browse to the output `coors.xyz` file and make sure that the file type is correctly determined. Otherwise, select XYZ in the [Determine file type] menu. After the trajectory is loaded, you can adjust the representation settings so that the final molecule looks as shown in Figure 1.

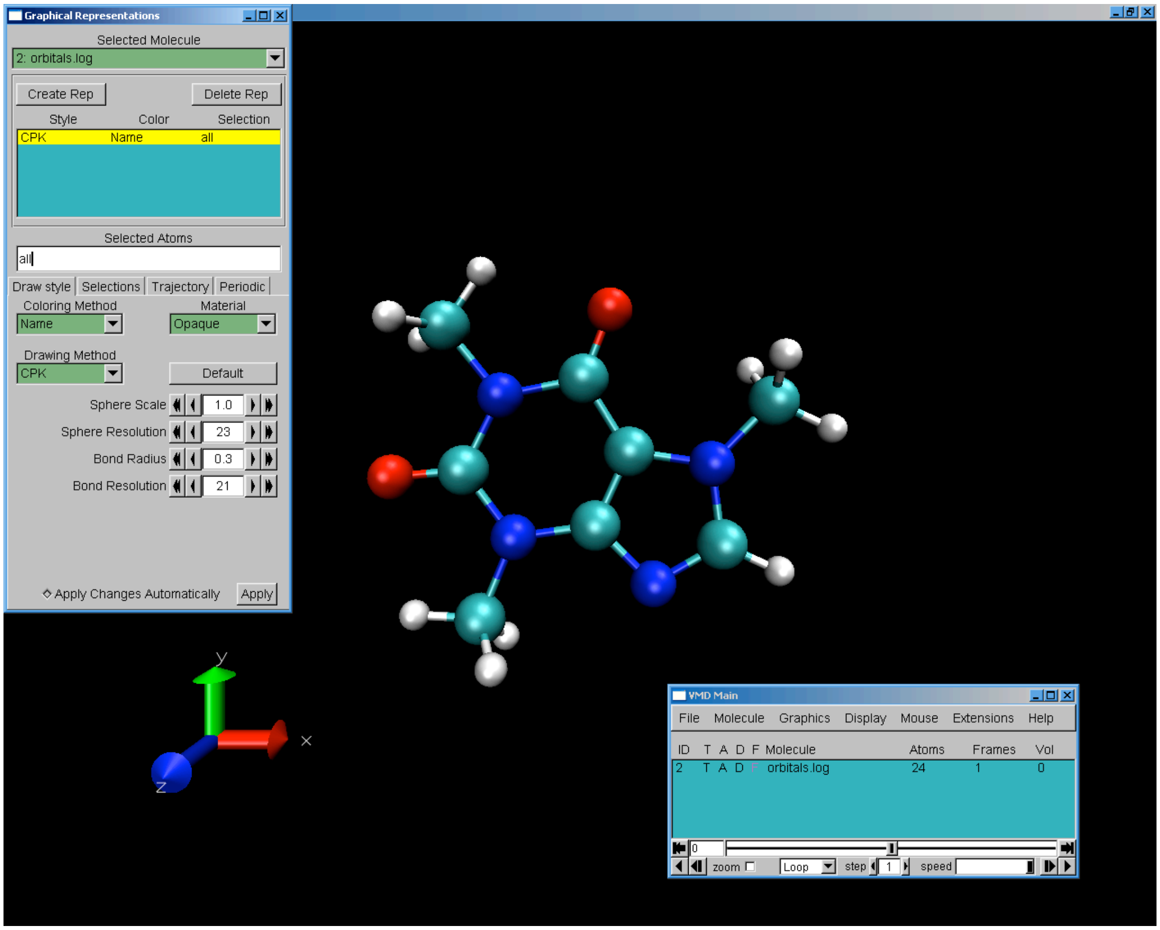

Figure 1. Caffeine molecule geometry output (`coors.xyz`) in VMD.

The molecular orbitals can be visualized with VMD in similar fashion. Again, go to VMD → File → New Molecule. Browse to the output `orbitals.log` file and make sure that the determined file type is GAMESS. Otherwise, select GAMESS in the [Determine file type] menu. To display both positive and negative isosurfaces you will need to create two representations for each orbital (one for the positive isosurface and one for the negative isosurface). After adjusting the atomic representation, create a new one and select "orbital" in the [Drawing Method] menu. All MO's will be listed in the [Orbital] dropdown along with the orbital energy. The [Isovalue] scroll bar controls the isosurface value (in a.u.).

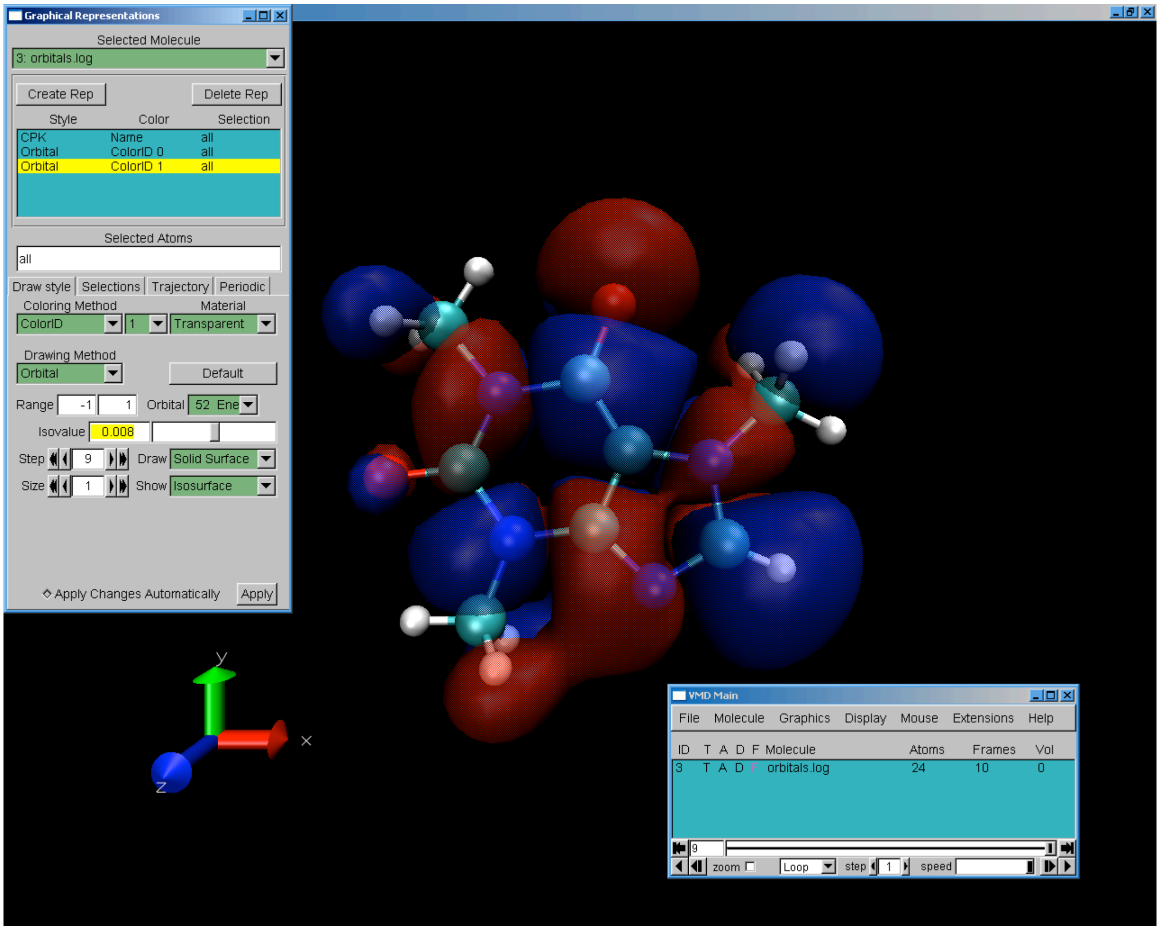

Figure 2. Caffeine molecule orbitals output (`orbitals.log`) in VMD.

VI Interactive calculations

Interactive calculations are especially suitable for remote jobs when TeraChem is running on a remote machine (cluster) and the trajectory visualization is performed on a local desktop. TeraChem can visualize the geometry optimization, TS search, and molecular dynamics trajectories in real time, i.e. interactively as the calculations run. These interactive molecular dynamics (IMD) runs are practical for molecules with up to approximately 20 atoms using a hardware solution with eight GPUs. Future versions will also allow for user manipulation of the molecule, i.e. imposing external forces on atoms. In the `instdir/tests/benzene` directory, you will find the files needed to try IMD for benzene. The configuration file `start.imd` reads:

```
# basis set
basis       sto-3g
# coordinates file
coordinates C6H6.pdb
# molecule charge
charge      0
# SCF method (rhf/blyp/b3lyp/etc...): Restricted Hartree-Fock
method      rhf
# type of the job (energy/gradient/md/minimize/ts): MD
run   md
# initial temperature in K
tinit       1000
# this triggers interactive molecular dynamics
# imd specifies the port VMD should connect to
imd   54321
# number of MD steps
nstep       1000
end
```

Open two terminal windows. In the first window, launch VMD and load in the coordinates of the benzene molecule from C6H6.pdb In the second terminal window, launch TeraChem, which will initialize the simulation and pause for connection to be made with VMD. Now, in VMD, select Extensions→Simulation→IMD Connect (NAMD). Type `localhost` (or the IP address of the remote machine on which TeraChem is currently running) in the `Hostname` field and `54321` (the port specified by the `imd` keyword in the TeraChem input file) in the `Port` field. Click the `Connect` radio button and you should see TeraChem executing in its window while the benzene molecule vibrates in the VMD display window.

VII TeraChem job parameters

Available job parameters are listed in Table 1. Note the use of the character "|" which should be interpreted as "or." For example, x|y means that *one* of x or y should be entered.

Table 1. Available TeraChem job parameters.

<table>
<tr><td>Parameter</td><td>Description</td><td>Default value</td></tr>
<tr><td colspan="3">General parameters</td></tr>
<tr><td>scrdir</td><td>Scratch directory</td><td>./scr</td></tr>
<tr><td>run</td><td>Sets calculation type
energy – Single point energy
project – Wavefunction projection for better initial guess
gradient – Energy and gradient
minimize – Optimize geometry
ts – Transition state search
md – Born-Oppenheimer Molecular Dynamics</td><td>energy</td></tr>
<tr><td>gpus</td><td>Number of GPUs to use in parallel. Sometimes one needs to change the default order of the devices. For example when device zero is used solely for display purposes. In such cases, the following line specifies which GPUs should be used (the enumeration starts from 0):
gpus 3   1 3 2
or
gpus 3
to use the default device order 0 1 2</td><td>all</td></tr>
<tr><td>precision</td><td>single|mixed|dynamic|double
By default, TeraChem uses a dynamic precision scheme where two-electron integrals larger than some threshold value are computed with double precision and the rest are computed with single precision. The threshold which determines the splitting between single precision and double precision evaluation of the integrals is determined automatically during the SCF calculation. The mixed precision scheme uses a static threshold to determine which integrals should be evaluated in double precision (those larger than threspdp, which can be set in inputfile). Users can</td><td>dynamic</td></tr>
</table>

	also request to use full single or double precision depending on the job purposes. Note that accumulation of all quantities on GPU is *always* performed with double precision accuracy to avoid floating point summation errors.	
`threspdp`	Threshold used to split single and double precision work in mixed precision calculations (atomic units).	`0.001`
`coordinates`	Name of file containing atomic coordinates	not set
`qmmm`	Name of file containing MM water coordinates The presence of this keyword triggers a QM/MM calculation	not set

`basis`	All available basis sets are located in `basis/` directory. In addition, users can construct customized basis sets. Here are examples of correct entries:
	`sto-3g, 6-31++G**, 6-311++G(3df,3pd),`
	`aug-cc-pvdz, mini(s), midi!`
	When the string is parsed, all `'*'` symbols are replaced by `'s'` and all parenthesis are replaced by brackets to comply with Unix file naming rules.
		not set

`projectbasis`	In `project` jobs, the converged wave function is projected onto `projectbasis` and stored in `scr/prjct` file (`scr/prjcta scr/prjctb` in unrestricted methods).	not set
`charge`	The total charge of the molecule (integer).	not set
`spinmult`	Spin multiplicity, `2S+1`, of wavefunction (integer).	`1`
`method`	Restricted, unrestricted, and restricted open shell HF and KS. Restricted wavefunction: `rhf|svwn|blyp|b3lyp|b3lyp1|b3lyp5|pbe|revpbe|pbe0|revpbe0|wpbe|wpbeh|bop|mubop|camb3lyp|b97|wb97|wb97x` b3lyp or b3lyp1: VWN1 correlation b3lyp5: VWN5 correlation Unrestricted calculations are invoked by adding `'u'` prefix, i.e. `uhf|ublyp|ub3lyp`, etc Restricted open shell calculations are invoked by adding `'ro'` prefix, i.e. `rohf|roblyp|rob3lyp`, etc	`rhf`

`rc_w`	Range correction scaling parameter ω required by DFT functionals with a fraction of exact long range exchange operator	`depends on DFT functional`
`c_ex`	Long range exact exchange operator contribution	`depends on DFT functional`
`dftgrid`	Integer value within [0-5] range, inclusive. Larger numbers are denser grids (and hence provide more accurate results). Grid 0 contains ~800 grid points/atom and grid 5 contains ~80,000 points per atom. The default grid (type 1) contains about 3,000 points per atom.	`1`
`dynamicgrid`	`yes|no` This parameter enables use of dynamical DFT grids. When it is on, grid 0 is used to converge the wavefunction until the DIIS error reaches the `gridthre` value (default: 0.01). Grid `dftgrid` is then used to finally converge the wavefunction.	`yes`
`gridthre`	Threshold for switching dynamical DFT grids.	`0.01`

`guess`	`generate` or `path/to/the/WFfile`
	
	`generate` means the initial WF guess is generated from scratch using maximum orbital overlap;[3] otherwise, it is loaded from the WF file. The WF is dumped in the end of each calculation to the `scr/c0` file (`scr/ca scr/cb` in unrestricted methods).
	
	To load WF in unrestriced calculations, `guess` requires two parameters, for example
	
	`guess scr/ca scr/cb`
		`generate`

`scf`	`diis` – Use Pulay's DIIS[4] for SCF convergence `oda` – Use optimal damping[5] for SCF convergence	`diis`
`start_diis`	When using optimal damping, switch to DIIS once DIIS error falls below threshold (float).	`0.01`
`fock`	`incremental` – Force incremental Fock matrix formation `conventional` – Force conventional Fock matrix	`auto`

3 J.-M. Langlois et al, J. Phys. Chem. **98**, 13498 (1994).

4 P. Pulay, J. Comp. Chem. **3**, 556 (1982).

5 E. Cances and C. Le Bris, Int. J. Quantum Chem. **79**, 82 (2000).

	formation `auto` – Use conventional algorithm if diffuse basis functions are detected or incremental otherwise	
`maxit`	Maximum number of SCF iterations (integer).	`100`
`convthre`	WF convergence threshold (float).	`3.0e-5`
`threall`	Two-electron integral threshold (float). Two electron integrals less than this threshold are neglected.	10^{-11}
`xtol`	Basis set linear dependency threshold (float). When diffuse basis functions are used, `xtol` and `convthre` may need to be raised up to `~1.0e-3`	`1.0e-4`
`dispersion` `dftd (alias)`	Should dispersion corrections[6] be used? `yes|no|d3|d2` D2 and D3 are two different dispersion parameterizations developed by S. Grimme.	`no`
`units`	Units used for coordinates (`angstrom` or `bohr`)	`angstrom`
`nbo`	NBO analysis (`no|yes|npa|full|advanced|$nbo`) `no` – no NBO analysis `yes\|npa` – Natural population analysis only `full` – Full NPA/NBO analysis `advanced\|$nbo` – Analysis with $NBO input TeraChem uses v6.0 of the NBO package. See www.chem.wisc.edu/~nbo5 for documentation of the earlier v5.0.	`no`
`polarizability`	Calculate polarizability tensor? (`yes|no`)	`no`
Geometry Optimization and Transition State Search		
`min_print`	How much output is desired (`something|verbose|debug`)	`verbose`
`nstep`	Maximum number of optimization/TS search steps	`100`
`min_tolerance`	Termination criterion based on the maximum energy gradient component	$4.5*10^{-4}$
`min_tolerance_e`	Termination criterion based on the SCF energy change	10^{-6}
`min_coordinates`	Type of coordinates in which optimization/TS search	`dlc`

[6] S. Grimme, J. Comp. Chem. **27**, 1787 (2006).

	is performed (cartesian|dlc|dlc_tc|hdlc|hdlc_tc)	
min_method	Optimization/TS search method (sd|cg1|cg2|lbfgs) sd – steepest descent cg1 and cg2 – conjugate gradient lbfgs – L-BFGS	lbfgs
min_hess_update	Hessian update algorithm (never|powell|bofill|bfgs) If never, the Hessian is recalculated using finite differences at each step	bfgs
min_init_hess	Initial Hessian (fischer-almlof|one-point|two-point|diagonal|identity) one-/two-point – exact Hessian from finite differences diagonal – only diagonal elements are calculated using final differences, off-diagonal elements equal zero identity – initial Hessian is an identity matrix fischer-almlof – chemically-motivated guess Hessian, usually the best choice	fischer-almlof
min_delta	Atomic displacement in finite difference calculations	0.003
min_max_step	Maximum step size in internal coordinates	0.5
min_restart	Whether the optimization/TS search job is started from scratch (no) or loaded from the checkpoint files (yes)	no
min_dump	How often the checkpoint files are written. By default at each 10th step.	10
ts_method	Transition state search method (neb_free|neb_restricted|neb_frozen| neb_free_cart|neb_restricted_cart| neb_frozen_cart) neb_free – Nudged Elastic Band (NEB) with free endpoints neb_restricted – NEB with endpoints allowed to move perpendicularly to their tangent direction neb_frozen – NEB with frozen endpoints	neb_free

	neb_x_cart – only initialization is performed in min_coordinates coordinates, while the TS search is done in Cartesians	
min_image	Number of NEB images in the TS search calculations. Should be greater than one. The images are listed in the input coordinates file (specified by coordinates). If the number of images found is smaller than min_image, the program will automatically generate missing images by interpolation. At least two images (endpoints) should be listed in the coordinates file. The last one (i.e. the min_imageth) is the climbing image.	10
Molecular dynamics parameters		
nstep	Total number of MD steps. Set nstep to 0 for single-point energy calculations (integer).	10^6
integrator	The way the initial WF guess is generated in subsequent MD iterations. In the first several iterations the initial guess is constructed from scratch. (reversible_d|reversible|regular|reset) reversible_d: time-reversible integrator with dissipation[7] reversible: time-reversible integrator without dissipation[8] regular: initial WF guess is taken from converged WF at previous MD step reset: initial WF guess is generated from scratch at each MD step	reversible_d
rseed	Seed for random number generator. Set this to a different integer for different MD runs (or set it to a known seed in order to reproduce an earlier run).	1351351
timestep	MD integration time step in femtoseconds (float)	1.0
thermostat	Temperature control – velocity rescaling or Langevin dynamics (rescale or langevin)	rescale
rescalefreq	When velocity rescaling is used, determines how often the velocities are rescaled. For instance, setting rescale to 1000 will force rescaling at every 1000th	$2 \cdot 10^9$

[7] A.M.N. Niklasson et al, J. Chem. Phys. **130**, 214109 (2009).

[8] A.M.N. Niklasson et al, Phys. Rev. Lett. **97**, 123001 (2006).

	MD step. To obtain NVE dynamics, set `rescalefreq` to a value larger than `nstep`.	
`tinit`	Initial temperature (K) sampled from Boltzmann distribution of velocities at T = `tinit` (float)	`300.0`
`t0`	Thermostat temperature (K) (float)	`300.0`
`lnvtime`	The Langevin damping time (fs), only used when `thermostat` is set to `langevin`.	`1000.0`
`orbitals`	Path to an output file containing the canonical molecular orbitals. The orbitals are printed in GAMESS format and can be visualized by VMD.	`orbitals.log`
`molden`	Path to an output file containing the canonical molecular orbitals in Molden format	`molden.molf`
`orbitalswrtfrq`	Determines how often the orbitals are written to the output file. Due to large size (sometimes the orbitals require 100MB and even more of disk space) it does not make sense to write orbitals at every MD iteration. (integer)	$2 \cdot 10^9$
`restartmd`	Path to the MD restart binary file. If set, the data in this file (rather than coordinates, velocities, temperature, etc) is used to start MD trajectory.	not set
`restartmdfreq`	Specifies how often the restart data is dumped to `scr/restart.md`	`100`
`mdbc`	`spherical` Enables spherical boundary conditions	not set
`md_density`	If set, R_1 is automatically adjusted to the specified density in g/mL	`1.0`
`md_r1, md_r2`	R_1 and R_2 parameters for spherical boundary conditions in Å	`0.0, 0.0`
`md_k1, md_k2`	k_1 and k_2 parameters for spherical boundary conditions in kcal/(mol Å^2)	`10.0, 0.0`
`imd`	Invokes interactive molecular dynamics by specifying the corresponding VMD port	not set

Contact information

We will be pleased to receive any feedback on your experience with TeraChem. Should you have any suggestions, concerns, or bug reports, please email them to help@petachem.com Also, please feel free to participate in the TeraChem User Forum at http://www.petachem.com/forum. TeraChem developers monitor this forum frequently and are happy to answer questions about the software.

Copyright

End user license agreement

IMPORTANT: PLEASE READ THIS END USER LICENSE AGREEMENT (“AGREEMENT”) CAREFULLY BEFORE DOWNLOADING, INSTALLING, OR USING THE SOFTWARE.

BY DOWNLOADING, INSTALLING, UPDATING, COPYING, OR USING THE SOFTWARE, YOU AGREE TO BE BOUND BY THIS AGREEMENT. IF YOU DO NOT AGREE TO ALL OF THE TERMS OF THIS AGREEMENT, DO NOT DOWNLOAD, INSTALL, UPDATE, COPY, OR USE THE SOFTWARE.

1. Grant of License. Subject to the conditions and limitations set forth herein, PetaChem, LLC (PetaChem) hereby grants you a limited, nonexclusive, and nontransferable license to use in binary form one (1) copy of the Software and any updates, patches, enhancements, technical support, documentation, and any related materials provided by PetaChem (collectively, “Software”). You may install and use the Software on one computer with up to eight central processing unit cores and eight graphical processing unit cards (the “Licensed Computer”). The Licensed Computer may be a storage device, such as a network server, used only to install or access the Software from other computers over an internal network, provided that each separate computer which accesses or uses the Software is a Licensed Computer.
2. License Termination. This license and Agreement is effective until terminated. This Agreement will terminate with or without notice on any breach by you of this Agreement. You may terminate this Agreement at any time. Promptly upon termination, you will cease using and destroy all copies of the Software in your possession or under your control. Sections 3-9 of this Agreement will survive any termination of this Agreement.
3. License Limitations. The Software is licensed and not sold. All rights not expressly granted herein are reserved by PetaChem. Except as expressly authorized in this Agreement, you will not:
    a. Sell, rent, lease, distribute, sublicense, assign, publish, or otherwise transfer (including by loan or gift) the Software, or any full or partial copies thereof;
    b. Copy, modify, or create derivative works of the Software in any way;
    c. Remove any copyright notice or licensing information contained in the Software;
    d. Disassemble or otherwise reverse compile or reverse engineer the Software; or
    e. Install the Software on any computer which is not a Licensed Computer.
4. Copyright. All content included in the Software, including, without limitation, text, graphics, images, logos, audio or video clips, digital downloads, data compilations, is the property of PetaChem or licensors and protected by the laws of the United States and other countries and international treaties. All trademarks that are not owned by PetaChem that appear in the Software are the property of their respective owners, which may or may not be affiliated with or connected to PetaChem.
5. Any reports or published results obtained with the Software will acknowledge its use by appropriate citation as follows:

    TeraChem v 1.5, PetaChem, LLC (2011). See http://www.petachem.com.

Any published work which utilizes TeraChem shall include the following reference:

"I. S. Ufimtsev and T. J. Martínez. Quantum Chemistry on Graphica Processing Units. 3. Analytical Energy Gradients and First Principles Molecular Dynamics, Journal of Chemical Theory and Computation, 5:2619-2628, 2009."

Electronic documents will include a direct link to the official TeraChem page at http://www.petachem.com.

6. Feedback. All suggestions, comments, or other feedback concerning your experience with or use of the Software that may be given to PetaChem ("Feedback") will be given voluntarily and without obligation or restriction of any kind. PetaChem may use such feedback for any purpose. Due to the nature of development work, PetaChem will not commit to correcting any reported errors or discrepancies. Feedback will not create any confidentiality obligation for PetaChem. You will not give Feedback that is subject to license terms that seek to require any product, technology, or service that incorporates or is derived from Feedback, or any intellectual property, to be licensed to or otherwise shared with you or any third party.

7. Disclaimers of Warranty and Liability.

    a. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, PETACHEM PROVIDES THE SOFTWARE "AS IS" AND WITH ALL FAULTS. PETACHEM HEREBY DISCLAIMS ALL EXPRESS, IMPLIED OR STATUTORY REPRESENTATIONS, WARRANTIES AND CONDITIONS REGARDING THE SOFTWARE, INCLUDING BUT NOT LIMITED TO MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE, QUIET ENJOYMENT, OR NONINFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY OF THE SOFTWARE, ITS USE OR PERFORMANCE, REMAINS WITH YOU.

    b. IN NO EVENT WILL PETACHEM BE LIABLE FOR ANY LOSS OR DAMAGE, INCLUDING WITHOUT LIMITATION ANY SPECIAL, INCIDENTAL, INDIRECT, SPECIAL, OR CONSEQUENTIAL LOSS OR DAMAGE WHATSOEVER, WHETHER FOR BREACH OF CONTRACT, IN NEGLIGENCE OR ON ANY OTHER THEORY OF LIABILITY, ARISING OUT OF OR IN ANY WAY RELATED TO THE SOFTWARE, THE PROVISION OF OR FAILURE TO PROVIDE TECHNICAL SUPPORT, OR OTHERWISE UNDER OR IN CONNECTION WITH THIS AGREEMENT, EVEN IF PETACHEM HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE. YOU ACKNOWLEDGE THAT THIS ALLOCATION OF RISKS IS A PART OF THE BARGAIN OF THIS AGREEMENT. Notwithstanding, PetaChem's total liability under or in connection with this Agreement and your exclusive remedy for all of the foregoing, however arising, is limited to direct damages up to the cost of the Software to you. The foregoing limitations, exclusions, and disclaimers will apply to the maximum extent permitted by applicable law, even if any remedy fails its essential purpose. Some states do not allow the exclusion or limitation of incidental or consequential damages, so the above limitations may be inapplicable to you.

8. Export. You may not use or otherwise export or reexport the Software except as authorized by United States law and the laws of the jurisdiction in which the Software was obtained. In particular, but without limitation, the Software may not be exported or reexported (a) into any U.S. embargoed countries or (b) to anyone on the U.S. Treasury Department's list of

Specially Designated Nationals or the U.S. Department of Commerce Denied Person's List or Entity List. By using the Software, you represent and warrant that you are not located in any such country or on any such list. You also agree that you will not use the Software for any purpose prohibited by United States law, including, without limitation, the development, design, manufacture or production of nuclear, missiles, or chemical or biological weapons.

9. Miscellaneous.
   a. This Agreement will be governed by and construed in accordance with the laws of the State of California as if entered into and performed wholly within the state and without regard to the principles of conflicts of law. You consent to exclusive jurisdiction and venue in the courts within the Northern District of California.
   b. This Agreement constitutes the entire agreement between parties with respect to the Software and merges all prior and contemporaneous communications. If any provision of this Agreement is held to be void or unenforceable for any reason, that provision will be enforced to the maximum extent permissible so as to effect the economic benefits and intent of the parties, and the remaining provisions of this Agreement shall remain in full force and effect.
   c. Neither party's failure or delay in exercising any of its rights will constitute a waiver of such rights. Any waiver or amendment of any provision of this Agreement will be effective only if in writing and signed by authorized representatives of both parties.
   d. Neither party may represent or bind the other in any way and nothing in this Agreement shall be construed as creating of the relationships of joint venturers, partners, employer and employee, franchisor and franchisee, master and servant, or principal and agent.