Priroda Documentation

Материал из KNCWiki.

By Dr. D.N.Laikov (http://www.physto.se/~laikov/) . [Corresponds to the version 5 of the code -- Grigory]

Содержание

Methods

Performance Considerations for Using Various Methods

GGA Density Functional Theories

These methods are the fastest in the program and allow analytic evaluation of the second derivatives of the energy.

It is advantageous to use a GGA DFT for preliminary structure optimizations and for generating starting hessians to speed up further calculations using other more expensive methods like hybride DFT, RI-MP2 and so on. We highly recommend this strategy.

The memory requirements for these methods are approximately 8*N*N*3 bytes for energy and gradient, and twice this amount for analytic hessian calculation, where N is the basis set dimension. Disk space requirements are very modest and grow cubically for hessian calculation up to some few gigabytes in largest cases.

In parallel execution most of the memory and all disk storage is used on the master node. WARNING: make sure that temporary files are written to a fast local disk, and not to a network drive.

Hybrid DFT Methods using Resolution-of-Identity (RI) approximation.

These are brand new in the program and have been just implemented at the end of November 2004. Note that these methods require special auxiliary basis sets. The rate limiting step of the calculation is the formation of the Exchange Matrix using three-center Coulomb-type integrals. The right choice of memory and disk settings is important for achieving high performance in large calculations. See the discussion of the RI-MP2 theory below.

Second-Order M\oller-Plesset Perturbation Theory within RI approximation.

The Resolution-of-Identity approximation consists in representing products of basis functions as linear combinations of auxiliary basis functions so that a four-index Coulomb integral can be evaluated in terms of three- and two-index quantities:

```
 (ij|kl) ~= (ij|m) [m|n] (n|kl)
 where [m|n] is the inverse of the Coulomb matrix (m|n).
```

The advantage of such approximation is the reduced storage requirement, faster integral evaluation and better scalability in parallel execution. In our RI-MP2 implementation the Hartree-Fock (HF) equations are solved using the RI aproximation, followed by the MP2 energy and the three- and four-index desity matrix formation, as well as subsequent coupled-perturbed HF solution.

The integrals over basis functions are written to disk on each node, the various transformed integrals are formed and sorted in memory and stored on disk. For large calculations more memory available to the program reduces disk transfer and improves performance. In the worst case memory requirements grow quadratically with the system size and disk storage has a cubic scaling.

It is highly IMPORTANT to use separate disk drives for each running process during parallel runs on multiprocessor systems. *[Unfortunately, for our clusters it is not possible -- Grigory 14:23, 30 июня 2006 (MSD)]*

Our implementation is scalable in principle up to Nproc = min (Nat, Nocc), where Nproc is the number of processors, Nat is the number of atoms and Nocc is the number of occupied correlated orbitals. It is important that the Nocc were as close as possible to a multiple of Nproc, for example if Nocc=14 then Nproc=2, 7 or 14 would be the best choice, Nproc=3 or 5 is acceptable, Nproc=6 is quite poor, and Nproc=13 a very bad choice.

The storage of all three-index quantities is distributed. On many parallel architectures with gigabytes of memory per node one can use RAM-disks to avoid disk usage, particularly if there is only one physical disk per two processors, which is unfortunately a typical configuration of a computer cluster.

Sofisticated Wavefunction Methods

Coupled Cluster and Perturbation Theory

Our implementation of vatious correlated wavefunction methods is intended for use on parallel platforms and is scalable up to the number of processors equal to the number of correlated electrons in beta subsystem.

For all four-index quantities distributed storage is used, memory requirements grow cubically and disk storage grows as fourth power of the problem size. When the number of virtual orbitals is much larger than the number of occupied orbitals, the rate limiting step for CCSD or MP4SDQ is the evaluation of the doubles-doubles term including two-electron integrals with all four indices in the virtual space. To improve performance in this case it is desirable to use more memory.

Input Summary

The program uses a kind of NAMELIST free format input. Data are organized in groups marked with a **$keyword** and ending with **$end** , for example **$system mem=128 $end**. Within each input group variable names can be abbreviated, like in above where "mem" was used as an abbreviation for "memory". Variables can be scalar or arrays, in the latter case one has an option to modify only some elements of an array using indices, for example **$guess na(85)=86,85 $end** assigns value 86 to 85th element of "na", and value 85 to 86th element.

Most of the parameters have appropriate default values... Some should be set according to the problem being studied...

$system

[Please not that on our clusters this group not to be used, if you submit your jobs with a pbsprirodaX script. The $system group will be created automatically based on your -mem and -disk command-line options -- Grigory 14:23, 30 июня 2006 (MSD)]

```
memory=%d    memory limit in megabytes
disk=%d      positive: disk space in gigabytes
              negative: amount of memory in megabutes
             to use as disk emulation
path=%s      directory path for temporary disk files.
              Colon-separated list of paths for each processor
              (optional for parallel execution) can be used,
              for example path=/tmpa:/tmpb:/tmpc:/tmpd
              specifies that for a four-processor execution
             process #1 writes files to /tmpa
             process #2 writes files to /tmpb
             and so on. In this case /tmpa , /tmpb , /tmpc and /tmpd
              should be mounted on different physical drives
             for good performance.
              **NOTE**: For all methods except GGA DFT the performance will be
              degraded terribly if more than one process read and write
              large files on the same physical drive!
              By defaul the program uses the TMPDIR environment variable,
             or the current working directory.
              **NOTE**: one should NEVER use an NFS (network file system)
             for temporary files.
```

$control

```
task=%s       job type, values are:
              energy
              gradient
              hessian
              optimize   - geometry optimization          - see $optimize
              irc        - intrinsic reaction coordinate  - see $optimize
              scan       - relaxed potential energy surface scan
                                                          - see $optimize
              nmr        - NMR shielding tensors (DFT only)
              dipole     - numeric evaluation of dipole polarizabilities
                           and their first derivatives w.r.t. nuclear coordinates
theory=%s     theoretical method:
              dft        - GGA DFT using density fitting
              ridft      - hybrid DFT with exact exchange using
                           resolution-of-identity integral approximiation (NEW)
              tddft      - GGA time-dependent DFT
              rimp2      - second-order M\oller-Plesset perturbation theory
                           using the RI approximation
              hf
              cis        - for RHF and UHF
              mp2        - for RHF, UHF and ROHF
              mp3        - for RHF
              mp4dq      - for RHF
              mp4sdq     - for RHF
              mp4sdtq    - for RHF
              ccd        - for RHF and UHF
              ccsd       - for RHF and UHF
              ccsdsd     - linear-response CCSD for excited states,
                           for RHF
              .....      - undocumented
state=%d,%d              for excited-state methods:
                         compute properties for n-th excited state
                         (first value) and include N states (second value)
                         in iterative solution of the equations
basis=%s                 file name containing basis set data,
                         if not specified - basis set data are
                         read from the input file
four=%d                  four-component option
                         0 - non-relativistic theory (default)
                         1 - scalar-relativistic theory
nucleus=%s               nuclear model:
                         finite - Gausian nuclear model
                                  (default for four-component theory)
                         point  - point nucleus model
                                  (default for non-relativistic theory)
print=%s                 a string specifying various printing options
                         which may be appended one after another:
                         +charges   - print Hirschfeld charges (for DFT methods)
                         +bonds     - print Mulliken popultions and bond
                                      orders (for SCF only)
                         +esr       - print ESR parameters
                         +vectors   - print SCF vectors
                         +molden    - generate input data for MOLDEN program:
                                      for task=hessian these are vibrations,
                                      and if +vectors+molden is specified,
                                      than the scf orbitals are also prepared
                                      for visualization using MOLDEN.
save=%s                  file name for saving and reading
                         scf vectors - see $guess
```

$guess

```
read=%d                  reading scf vectors from file
                        1 - read
                         0 - do extended H"uckel guess
mix=%d                   mixing HOMO and LUMO for broken-spin UHF
                         0 - don't(default)
                          1 - do mix. Only sensible for unrestrichted HF (restrict=0 in $scf for singl
na(%d)=%d,%d,...         reorderig of starting vectors for alpha spin
nb(%d)=%d,%d,...         reorderig of starting vectors for beta  spin
                         these arrays default to
                         na(1)=1,2,3,4,5,...,N
                         nb(1)=1,2,3,4,5,...,N
                          example: na(25)=26,25 - swap 25th and 26th vectors
```

$dft

```
functional=%s            exchange-correlation approximation (for DFT)
                          PBE  - Perdew-Burke-Ernzerhof (default GGA)
                                 Phys. Rev. Lett. 77 (1996) 3865-3868
                         mPBE - Adamo-Barone modification of PBE
                                 J. Chem. Phys. 116 (2002) 5933-5940
                         BLYP - Becke-Lee-Yang-Parr
                                Phys. Rev. A 38 (1988) 3098-3100
                                Phys. Rev. B 37 (1988) 785-789
                          OLYP - Handy-Cohen exchange + LYP correlation
                                 J. Chem. Phys. 117 (2002) 1441-1449
                         PBE1 - a hybrid of PBE with 1/4 exact exchange:
                                 J. Chem. Phys. 110 (1999) 6158-6170
                                 (defaul for theory=riDFT)
                          B3LYP - Becke's three-parameter hybrid of the form
                                  Exc = 0.2*Ex(HF) + 0.8*(0.9*Ex(B) + 0.1*Ex(S))
                                      + 0.81*Ec(LYP) + 0.19*Ex(PW)
                                 where
                                  Ex(HF)  is the exact Hartree-Fock exchange,
                                  Ex(B)   is the Becke's GGA for exchange
                                          (Phys. Rev. A 38),
                                  Ex(S)   is the Slater's local exchange,
                                  Ec(LYP) is the LYP correlation
                                          (Phys. Rev. B 37)
                                  Ec(PW)  is the local correlation from
                                          Phys. Rev. B 45 (1992) 13244-13249
                                  Note that some comercial programs use
                                  Ec(VWN) instead of Ec(PW),
                                  even though A.D. Becke himself used Ec(PW),
                                  see J. Chem. Phys. 98 (1993) 5648-5652.
                                  Ec(PW) is a better approximation
                                  to the uniform electron gas correlation energy
                                  in comparison with Ec(VWN)
```

$scf

```
restrict=%d              0 - spin-unrestricted
                          1 - spin-restricted (available only for few methods)
                         By default restrict=1
                          if the system has an even number of electrons,
                         and restrict=0 otherwise.
convergence=%f,%f        scf/cp-scf convergence:
                          1e-6, 1e-3 by default (recommended).
                         You may want to decrease it to 1e-5
                          during preliminary geometry optimizations
                         in difficult cases - to save the time.
iterations=%d,%d         number of iterations/microiterations.
procedure=%s             a procedure to solve the scf equations:
                          nr    - Newton-Raphson quadratically-convergent
                                  (default for GGA DFT)
                         bfgs  - BFGS hessian update
                          If one procedure fails you may want to try another...
d1small=%d               1 - include small component density when solving
                              linear equations of quadratically-convergent SCF
                              (for scalar-relativistic pure-DFT)
                          0 - discard these contributions (default)
                              This results in faster calculation,
                              but may fail to converge for molecules
                             with heaviest atoms.
core=%d                   number of core electrons (for HF-based methods),
                          by default it is computed from an atom table
                         -- see $atoms
```

$atoms

```
core=%d,%d,...           for HF-based methods this specifies the number
                          of core shells for L=0,1,2,3 for each atomic number.
                         Example:
                         core(280)=2,1,0,0
                          specifies that 1s,2s,2p shells (10 electrons)
                         for Nickel belong to the core.
                         Note: these data are used only to compute
                          the total number of core electrons in the molecule.
mcore=%d                 a simple way to specify the number of outer core shells
                          to be included in correlation treatment.
                          For example, mcore=M means to add all shells with
                         principal quantum number M less than that
                         of the outermost valence shell.
```

$grid

```
accuracy=%f              accuracy (of xc-energy per atom) of the adaptively
                         generated grid (for DFT methods),
                         1e-8 by default (fine grid).
                         Value 1e-7 can be used to speed up some
                          geometry optimizations with tolerance<=1e-4,
                         Value 1e-9 can be used for very accurate
                         calculations.
```

$optimize

```
saddle=%d                 1 - optimize a saddle point
                          0 - optimize to a minimum (default)
tolerance=%f,%f           tolerance on gradient (first value)
                          and displacement (second value) in au.
                          default is 0.0001,0.01 ,
                          if only the first is given, the second is
                          computed as 100 times as much (recommended default)
trust=%f,%f,%f            trust radius - maximum value, minimum value,
                          and initial value. If only the first is specified,
                          then the minimum value is 1/16 thereof,
                          and the initial value is 1/4 thereof.
follow=%d                 follow n-th eigenvector (for saddle point search)
steps=%d                  number of optimization cycles (50 by default)
cartesian=%d              0 - use redundant internal coordinated (defaul)
                          1 - use cartesian coordinated (last resort)
back=%d                   1 - trace IRC path backwards
                          0 - trace IRC path forwards
points=%d                 optimize N points during an IRC run,
                          or do N constrained geometry optimization
                          during a scan run.
radius=%f                 radius in mass-weighted coordinates
                          for IRC path optimization : the distance
                          between adjacent IRC points will be roughly twice
                          as much.
coordinated=%d,%d,...     a list of redundant internal coordinates to use.
                          By default it is generated automatically.
                          Each coordinate is specified by 5 integers:
                          1,i,j,0,0  - distance i-j
                          2,i,j,k,0  - angle    i-j-k
                          3,i,j,k,l  - torsion  i-j-k-l
                          4,i,j,k,l  - special coordinate for linear fragments
                          5,i,j,k,l  - special coordiante for linear fragments
fix=%d,%d,...             a list of internal coordinated to be kept fixed
                          during geomety optimization. The format is like above
value=%f,%f,...           the values of the fixed internal coordinates
                          (in angstroms and degrees). By default these are
                          computed from input geometry. For the scan run type
                          this is a list of all initial values followed
                          by the list of final values. Example:
                          $optimize
                            fix   =1,7,9,0,0, 1,8,10,0,0, 3,8,7,9,0, 3,10,9,7,0
                           val(1)=1.6,       1.7
                           val(5)=2.0,       2.1
                           points=5
                          $end
                          In this case 5 geometry optimizations will be done
                          with values 1.6, 1.7, 1.8, 1.9, 2.0 for distance 7-9
                               values 1.7, 1.8, 1.9, 2.0, 2.1 for distance 8-10
                          and with angles 8-7-9 and 10-9-7 fixed at their
                          input values.
gradient=%d               when using second derivatives precomputed
                          by a theoretical method/basis set other than currently
                          used, specify gradient=0 to exclude gradient terms
                          in internal/cartesian transformation at the first
                          geometry. Otherwise use the default.
print=%d                  printing:
                          0 - minimal
                          1 - detailed
```

$thermo

```
sigma=%d                 symmetry number for thermochemistry
temperature=%f,%f,...   an array of up to 50 temperatures in K.
```

$d2edr2

```
length=%f            step length for numeric differentiation of
                     potential energy surface,
                     for numerical hessian calculation.
displace=%d          1 - use one displacement for each coordinate
                     2 - use two displacements (more accurate but twice
                         slower, default)
                     Note that in some cases you may want to increase
                     SCF convergence to get more accurate numerical hessians.
steps=%d,%d          initial and final numerical differentiation steps for
                     this run: a useful option for numerical hessian restarts:
                     let the reference geometry be #0, 1st displaced
                     geometry #1 and so on up to 3N or 6N (1 or 2 disp.).
                     default is: 0,-1 which means to do all necessary steps
                     from 0 to 3N|6N in one run.
                     Other possibility can be illustrated by the example:
                     for a 12-atom molecule one can prepare 4 input files
                     which differ only in that:
                     $d2edr2 disp=1 steps= 0, 9 $end - for file mol.in1
                     $d2edr2 disp=1 steps=10,18 $end - for file mol.in2
                     $d2edr2 disp=1 steps=19,27 $end - for file mol.in3
                     $d2edr2 disp=1 steps=28,36 $end - for file mol.in4
                     Then 4 calculation can be started in parallel on
                     different computers, after which the data can be
                     extracted from all output files and appended to one
                     input file:
                     grep "num>" mol.out{1,2,3,4} | sed "s/\.*num>//g" >> mol.in
                     then one adds
                     $d2edr2 steps=-1 $end
                     and executes the program once again to process the partial
                     results.
```

$nmr

```
standard=%f,%f,...
               Shielding constants for each atomic number to use as
               standards in NMR chemical shift calculations. One may
               specify values for atoms present in the molecule only,
               for example:
                 standard(1)=31.3915
                 standard(6)=181.9921
               which is computed for Si(CH3)4 can be used for a series
               of hydrocarbons under study.
points=%f,%f,%f,
       %f,%f,%f,
       ...
               An option to add points in space (with x,y,z coordinates in
               angstroms) for which magnetic shielding tensors will be
               computed in addition to the nuclear centers.
```

$molecule

This input group has special format with ordered data:

```
charge=%d          - charge of molecule
mult=%d            - spin multiplicity
               next should follow one of the two keywords:
cartesian          - cartesian coordinates will be used
z-matrix           - distance/angle/torsion coordinates will be used
               in "cartesian" case the input consists basically
               of four numbers four each atom:
%d %f %f %f        - atomic number, x, y, z coordinate in Angstroms.
               next an atomic mass may be specified, if using different
               isotopes:
mass=%f
               this is repeated for each atom in the system
               in the case of "z-matrix" input one uses the format
%d %d %f %d %f %d %f mass=%f
               to specify the atomic number and three internal coordinates.
               Different combinations of distances/angles/torsions are
               possible:
               q_i j r_ij  k a_ijk  l b_ijkl - distance, angle, and torsion;
               q_i j r_ij  k a_ijk -l a_ijl  - distance and two angles
               q_i j r_ij -k r_ik   l b_ijkl - two distances and torsion
               q_i j r_ij -k r_ik  -l r_il   - three distances
               the mass specification is optional as above.
               Distances are in Angstroms, angles and torsions are in degrees.
               One has the possibility to mix both formats:
               The first atoms can be entered in cartesian format,
               followed by a z-matrix input of the remaining ones.
               This is handy for modifying stuctures.
               If z-matrix input is used alone, the first atom should be
               specified only by its atomic number, the second using
               only the distance, the third using only
               two internal coordinates.
set=%s         This optional keyword can be used to specify the basis set
               for the atoms whose coordinates follow after it.
               For example:
                $molecule
                 z-matrix
                 set=B2
                   8
                 set=B1
                   1    1    0.96252725
                   1    1    0.96252725    2    102.943872
                $end
              Here the oxygen atom has a 'B2' set,
              while the hydrogens have 'B1' instead.
              The names of the basis sets should match those found in the
              basis set file.
```

Input Examples

O2 triplet riMP2 relativistic optimization

```
$control
 task=optimize
 theory=riMP2
 basis=basis4.in
 four=1
$end
$optimize tol=1e-5 $end
$molecule
 mult=3
 z-matrix
 set=L1
   8
   8    1    1.25017467
$end
```

UO2F2 relativistic PBE0 energy, L2 basis set, print orbitals.

```
$control
 task=energy
 theory=riDFT
 basis=basis4.in
 four=1
 print=charges+bonds+vectors+molden
$end
$scf iter=100 conv=1e-6 $end
$dft functional=PBE1 $end
$molecule
 z-matrix
 set=L2
  92
   9    1    2.06303279
   9    1    2.06303314    2    111.520778
   8    1    1.78032783    2     92.742191   -3     92.742194
   8    1    1.78032789    3     92.742132   -2     92.742273
$end
```

An Alder-ene transition state, Hessian calculation, non-relativistic, broken-spin.

```
$control four=0 task=hessian theory=DFT basis=/usr/local/bin/basis1.bas
 print=molden $end
$guess mix=1 $end
$dft functional=PBE $end
$scf iterations=300 procedure=NR restrict=0 $end
$System mem=200 disk=500 $end
$optimize steps=100 tol=5.0e-5 saddle=1 $end
$molecule cartesian
 set=L11
   6     1.01645147   2.27475539  -1.17198568
   6     1.26836007   0.73274534  -1.20793578
   6     0.00036881   0.04712739  -0.78455705
   6    -0.53254131   0.44631950   0.51342874
   6     0.03013686   1.75607120   1.09663365
   6     1.30035831   2.19100351   0.35311076
   6     2.18823044   0.95367038   0.03721674
   6    -0.45429900  -1.11180116  -1.37134200
   6     0.29098246  -2.55531997   0.05742784
   6    -0.77258597  -3.13993827   0.97600174
   6    -0.70967090  -4.60154851   1.41170063
   6     0.30288137  -2.15726484   1.35126070
   6    -2.13023072  -2.63751701   0.84773650
   7    -3.21748670  -2.23395371   0.74449811
   1    -1.62578923   0.35429176   0.59059479
   1     1.70315349   0.28849564  -2.11497929
   1     3.19956901   1.25176254  -0.27500973
   1     2.25005172   0.14754958   0.78351579
   1     1.77417764   3.05898792   0.83540403
   1    -0.72612808   2.55533966   1.01978110
   1     0.24517225   1.62811164   2.17064395
   1     0.02860384   2.63529310  -1.49685787
   1     1.81155051   2.81623684  -1.70517878
   1     0.00384537  -1.45948992  -2.30161010
   1    -1.47730592  -1.44969023  -1.18841008
   1     0.97208725  -3.03813540  -0.63813662
   1     0.99592296  -2.06556461   2.18381219
   1    -0.19178868  -0.48914999   1.16296277
   1    -1.15185452  -4.73150344   2.41162233
   1     0.33873199  -4.92936057   1.44779838
   1    -1.26038281  -5.24826177   0.71104928
$end
```