Example I: Coupled-cluster calculations for the water molecule

In this example we will calculate the ground state energy of the water molecule employing the CCSD (coupled-cluster with singles and doubles) parametrization of the electronic wavefunction:

$$\left|\Psi_{CCSD}\right\rangle = e^{T_1+T_2}\left|\Phi\right\rangle$$

and using Tensor Contraction Engine (TCE) module in NWChem and RHF reference $|\Phi>$.

Analytic gradients are not yet available for geometry optimization. However, using even analytical gradients say in CCSDT case is too expensive for most systems. The following strategy is "digestible"
1.) Use low-order methods (DFT,MP2,…) to locate possible energy minimas (isomers)
2.) Perform single point calculations for geometries obtained in point (1) using more accurate methods, for example CCSD(T) method.

Input file:

```
echo

geometry units bohr
H    0.000000000000000    1.079252144093028    1.474611055780858
O    0.000000000000000    0.000000000000000    0.000000000000000
H    0.000000000000000    1.079252144093028   -1.474611055780858
end
```

Geometry specification

```
basis
H library sto-3g
O library sto-3g
end
```

Basis set specification

```
scf
thresh 1.0e-10
tol2e 1.0e-10
singlet
rhf
end
```

SCF module provides reference, $|\Phi>$, for the CCSD calculations

```
tce
ccsd
end
```

In this line we define the CC approximation: in this case – CCSD approach

```
task tce energy
```

Output:

STEP1: SCF module which produces molecular orbitals further used to
generate molecular one- and two-electron integrals (these integrals
are requested by the CCSD modul)

iter	energy	gnorm	gmax	time
1	-74.9239094230	6.49E-01	5.52E-01	0.1
2	-74.9617970554	1.35E-01	9.96E-02	0.1
3	-74.9626616082	3.16E-03	2.92E-03	0.1
4	-74.9626630621	7.11E-06	6.49E-06	0.2
5	-74.9626630621	5.69E-13	4.35E-13	0.2

```
       Final RHF  results
       -----------------

       Total SCF energy =     -74.962663062148
  One-electron energy =    -122.219010644891
  Two-electron energy =      38.161498509339
Nuclear repulsion energy =       9.094849073405
```

STEP2: CCSD calculations, program iteratively solves so called CCSD equations
that define so-called cluster amplitudes used to calculate the CCSD
energy.

CCSD iterations

Iter	Residuum	Correlation	Cpu	Wall
1	0.0891232379551	-0.0358672469179	0.4	0.4
2	0.0317596201320	-0.0454068882657	0.4	0.4
3	0.0126828916023	-0.0483870059027	0.4	0.4
4	0.0053832778844	-0.0494370597647	0.4	0.4
5	0.0023954452285	-0.0498391184890	0.4	0.4
6	0.0011108272683	-0.0500021724029	0.4	0.4
7	0.0005330437725	-0.0500711904756	0.4	0.4
8	0.0002625570400	-0.0501014381364	0.4	0.4
9	0.0001317707641	-0.0501150974135	0.4	0.4
10	0.0000669953658	-0.0501214303300	0.4	0.4
11	0.0000343654836	-0.0501244348663	0.4	0.4
12	0.0000177357555	-0.0501258887096	0.4	0.4
13	0.0000091923943	-0.0501266039080	0.4	0.4
14	0.0000047789221	-0.0501269605251	0.4	0.4
15	0.0000024899940	-0.0501271402835	0.4	0.4
16	0.0000012995442	-0.0501272316751	0.4	0.4
17	0.0000006791083	-0.0501272784536	0.4	0.4
18	0.0000003552400	-0.0501273025229	0.4	0.4
19	0.0000001859739	-0.0501273149581	0.4	0.4
20	0.0000000974237	-0.0501273214031	0.4	0.4

Usually it takes few iterations to converged the CCSD equations

```
Iterations converged
CCSD correlation energy / hartree =         -0.050127321403066
CCSD total energy / hartree      =         -75.012790383550865
```

CCSD total energy = SCF energy (STEP1) + CCSD correlation energy

IMPORTANT: In many cases to achieve the desired level
of accuracy one needs to include energy correction due to
triply excited configurations: CCSD(T) method

$$E_{CCSD(T)} = E_{CCSD} + \delta_{CCSD(T)}$$

*The CCSD(T) correction $\delta_{CCSD(T)}$ can be
expressed in terms of converged CCSD amplitudes*

Changes in the input file:

tce
ccsd(t) ←———————— *Now, this line defines the CCSD(T) method*
end

Output:

STEP1: SCF calculations   (same as in the CCSD calculations)

STEP2: CCSD calculations   (same as in the CCSD calculations)

STEP3: CCSD(T) calculations

```
CCSD[T] correlation energy / hartree =         -0.050213173443135
CCSD[T] total energy / hartree       =         -75.012876235564164
CCSD(T) correlation energy / hartree =         -0.050202369883323
CCSD(T) total energy / hartree       =         -75.012865432004347
```

Other options available for ground and excited states:

ccsdt, ccsdtq, mp2, mp3, mp4, cisd, cisdt, cisdtq, eomccsd, eomccsdt,
eomccsdtq

Example II: Coupled-cluster calculations for the methyl radical

In this example we will calculate the ground state energy of the methyl radical using The CCSD(T) approach, Tensor Contraction Engine (TCE) module in NWChem, and UHF reference $|\Phi\rangle$.

Input file:

```
echo

geometry units angstrom
 symmetry c1
C  0.000   0.000  0.00
H  0.000   1.078  0.00
H  0.934  -0.539  0.00
H -0.934  -0.539  0.00
end

basis
H library cc-pVDZ
C library cc-pVDZ
end

scf
thresh 1.0e-10
tol2e 1.0e-10
doublet
uhf
end

tce
ccsd(t)
end

task tce energy
```

The cc-pVDZ basis set will be used

UHF reference describing dublet state is requested

Output:

STEP1: UHF calculations

iter	energy	gnorm	gmax	time
1	-39.5502480260	2.59D-01	1.59D-01	0.7
2	-39.5632540852	4.19D-02	1.60D-02	0.8
3	-39.5638686922	1.44D-03	5.59D-04	0.9
4	-39.5638693247	1.19D-06	5.99D-07	0.9
5	-39.5638693247	3.96D-12	9.41D-13	1.1

```
      Final UHF  results
      ------------------
```

```
        Total SCF energy =    -39.563869324676
    One electron energy =    -71.579679128774
    Two electron energy =     22.331785271555
Nuclear repulsion energy =      9.684024532543
```

STEP2: CCSD calculations based on the UHF reference

CCSD iterations
```
 ----------------------------------------------------------
 Iter        Residuum        Correlation    Cpu    Wall
 ----------------------------------------------------------
    1   0.1085002713208   -0.1316819355375    0.2    0.2
    2   0.0327770316627   -0.1485568757579    0.2    0.2
    3   0.0209747968750   -0.1532266417817    0.2    0.2
    4   0.0095442226038   -0.1532976563254    0.2    0.2
    5   0.0077324059600   -0.1530224716884    0.1    0.2
    6   0.0040662550801   -0.1537651898915    0.1    0.2
    7   0.0025109918683   -0.1539859907594    0.1    0.2
    8   0.0014493008153   -0.1540911006375    0.1    0.2
    9   0.0006504935953   -0.1541886515069    0.1    0.2
   10   0.0002980125325   -0.1541701459400    0.1    0.1
   11   0.0001548310993   -0.1541677205681    0.1    0.1
   12   0.0000660558607   -0.1541728231536    0.1    0.1
   13   0.0000414315756   -0.1541737633684    0.2    0.2
   14   0.0000177165137   -0.1541740907934    0.1    0.1
   15   0.0000106547087   -0.1541746548581    0.2    0.1
   16   0.0000039612316   -0.1541742746123    0.2    0.2
   17   0.0000022998365   -0.1541742216280    0.2    0.2
   18   0.0000008723337   -0.1541742464035    0.1    0.1
   19   0.0000005483497   -0.1541742321548    0.1    0.1
   20   0.0000002106069   -0.1541742300318    0.1    0.1
   21   0.0000001259082   -0.1541742295841    0.1    0.1
   22   0.0000000464890   -0.1541742256218    0.1    0.1
 ----------------------------------------------------------
 Iterations converged
 CCSD correlation energy / hartree =      -0.154174225621815
 CCSD total energy / hartree      =     -39.718043550297843
```

STEP3: Forming the CCSD(T) correction
```
 CCSD[T] correlation energy / hartree =     -0.156961771551652
 CCSD[T] total energy / hartree       =    -39.720831096227677
 CCSD(T) correlation energy / hartree =     -0.156917360108930
 CCSD(T) total energy / hartree       =    -39.720786684784954
```

Example III: Coupled-cluster calculations of the excitation energies of the water molecule

In this example we will calculate the excitation energies using the EOMCCSD (equation-of-motion coupled-cluster with singles and doubles) parametrization of the excited-state electronic wavefunction:

$$\left| \Psi_{EOMCCSD} \right\rangle = (R_0 + R_1 + R_2) e^{T_1 + T_2} \left| \Phi \right\rangle$$

and using Tensor Contraction Engine (TCE) module in NWChem.

```
Input:

echo

geometry units bohr
symmetry c1
H     1.474611052297904    0.000000000000000    0.863401706825835
O     0.000000000000000    0.000000000000000   -0.215850436155089
H    -1.474611052297904    0.000000000000000    0.863401706825835
end

basis
* library sto-3g
end

scf
thresh 1.0e-14
tol2e 0.0e-10
singlet
rhf
end

tce
ccsd
freeze core atomic
thresh 1.0d-12
nroots 1
end

task tce energy
```

Freeze defines core orbitals that are excluded in the calculations

convergence threshold for EOMCCSD procedure

nroots specifies the number of excited states to be determined.

```
Output:
```

STEP1: SCF calculations

STEP2: CCSD calculations

STEP3: EOMCCSD calculations

```
========================================
Excited-state calculation ( a    symmetry)
========================================

x1 file size    =             8

x2 file size    =           128

No. of initial right vectors    1

EOM-CCSD right-hand side iterations
----------------------------------------------------------------
     Residuum        Omega / hartree  Omega / eV   Cpu     Wall
----------------------------------------------------------------

Iteration   1 using    1 trial vectors
  0.3441834723239   0.4960382793056   13.49789    0.0      0.1

Iteration   2 using    2 trial vectors
  0.0569282647092   0.4422999231348   12.03560    0.0      0.1

Iteration   3 using    3 trial vectors
  0.0097995551546   0.4407319040407   11.99293    0.0      0.1

Iteration   4 using    4 trial vectors
  0.0019045964964   0.4406584247345   11.99093    0.0      0.1

Iteration   5 using    5 trial vectors
  0.0004988184955   0.4406563205074   11.99087    0.1      0.1

Iteration   6 using    6 trial vectors
  0.0000114045018   0.4406554471827   11.99085    0.0      0.1

Iteration   7 using    7 trial vectors
  0.0000000000000   0.4406555021961   11.99085    0.1      0.1
----------------------------------------------------------------
Iterations converged
```

To obtain the EOMCCSD excited-state energies one has to diagonalize non-hermitian operator (so-called similarity transformed Hamiltonian) $\overline{H} = e^{-(T_1+T_2)} H e^{(T_1+T_2)}$ in the space spanned by all singly- and doubly-excited configurations. The most efficient diagonalization procedure are based on Nakatsuji-Hirao algorithm,