OpenMX Calculations: A Case Study of Hydrogen Molecule

Truong Vinh Truong Duy

Ozaki Laboratory

Research Center for Integrated Science

Japan Advanced Institute of Science and Technology (JAIST)

Dec 15, 2011

Most slides are provided by Prof. Ozaki

Outline

- Introduction to OpenMX
- Mathematical Structure of Kohn-Sham Equation
- Total Energy and Force
- Hydrogen Molecule and the Input File
- Calculation Flowchart
  - Input
  - Main Computation Loop
  - Output
- Result
- Conclusion

OpenMX Open source package for Material eXplorer

http://www.openmx-square.org

Basic methods:

Functional:	LDA, GGA, LDA+U
Basis function:	Localized pseudoatomic basis funtions
Treatment of core:	Norm-conserving peudopotentials (PP)
Poisson solver:	Fast Fourier transform (FFT)
Integration method:	Regular mesh + projector expansion method
Eigenvalue solver:	Householder+QR methods
Geometry opt.:	BFGS and eigenvector following (EF) methods
Parallelization:	MPI+OpenMP

Extensions:

O(N) eigenvalue solver:	Krylov subspace, divide-conqure methods
Non-collinear DFT:	Two-component spinor with constrained schemes
Relativistic effects:	Fully relativistic PP with spin-orbit coupling
Electronic polarization:	Berry phase scheme
Electronic transport:	Ballistic transport by NEGF
Wannier function (WF):	Maximally localized WF and mapping to a TB model
……	

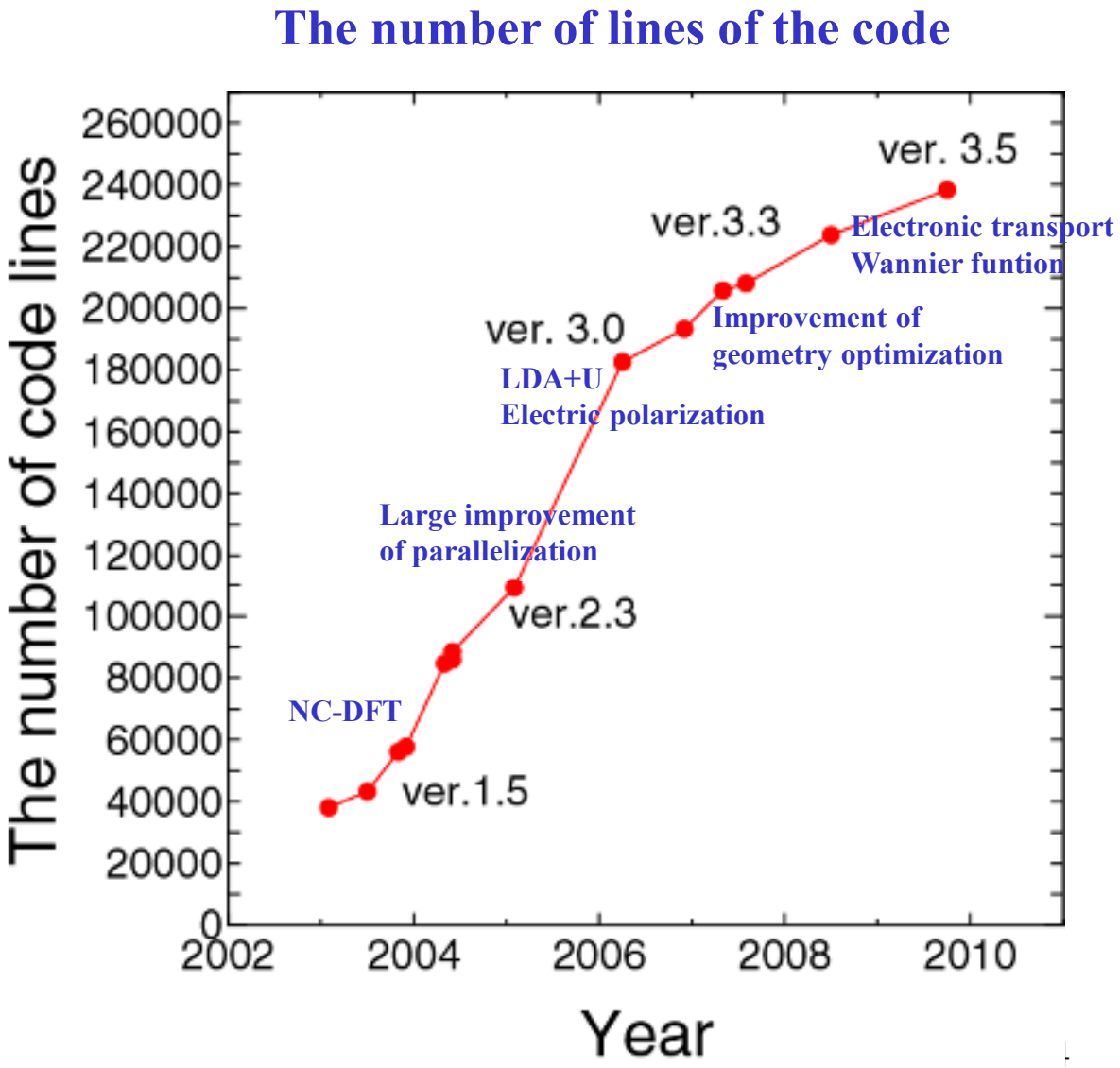
Short history of OpenMX

The development of the code has been started from the middle of 2000.

The first public release was done at 2003 January, and fifteen releases were made until now.

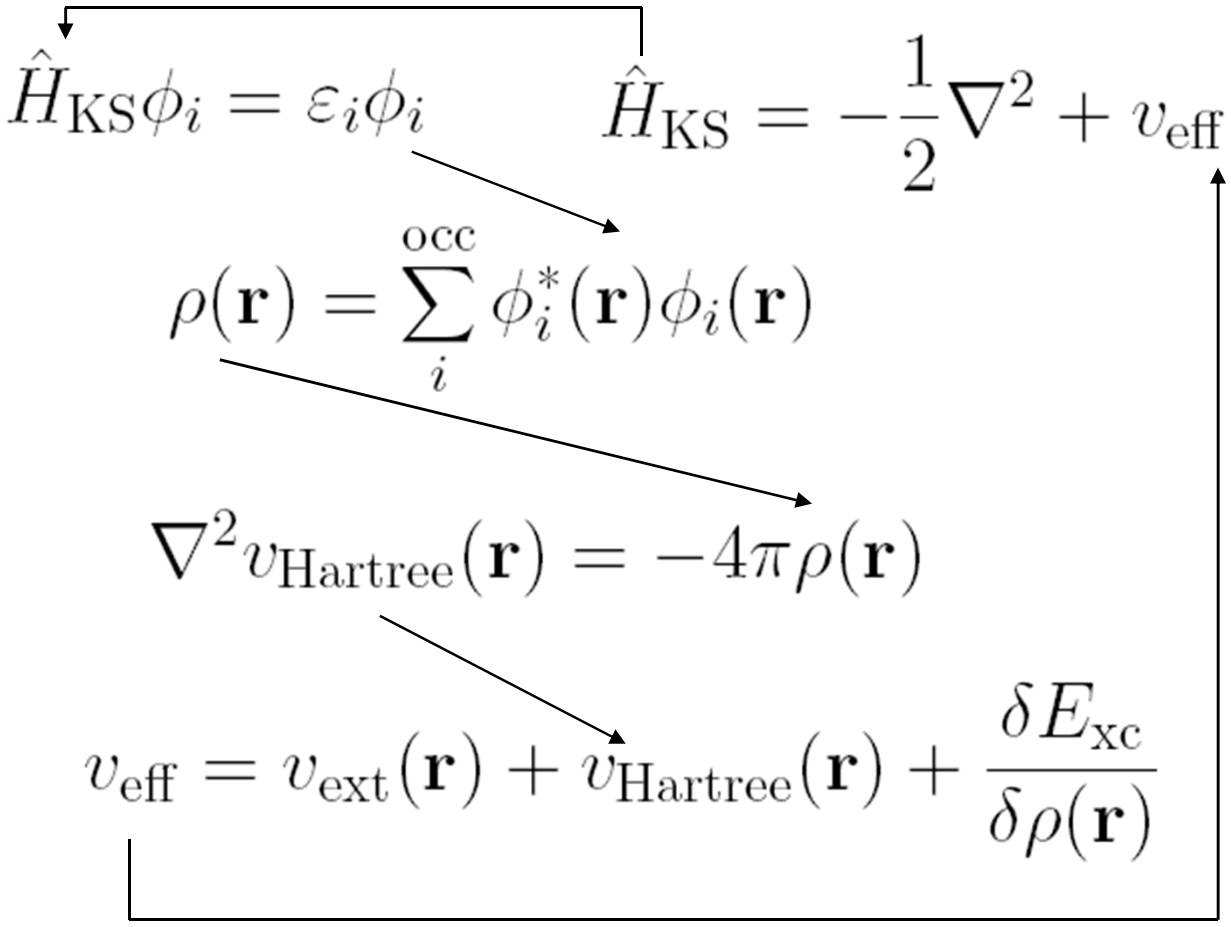
The code has been steadily developed as shown in the figure, and the community itself has been also growing.

About 100 related papers have been published.



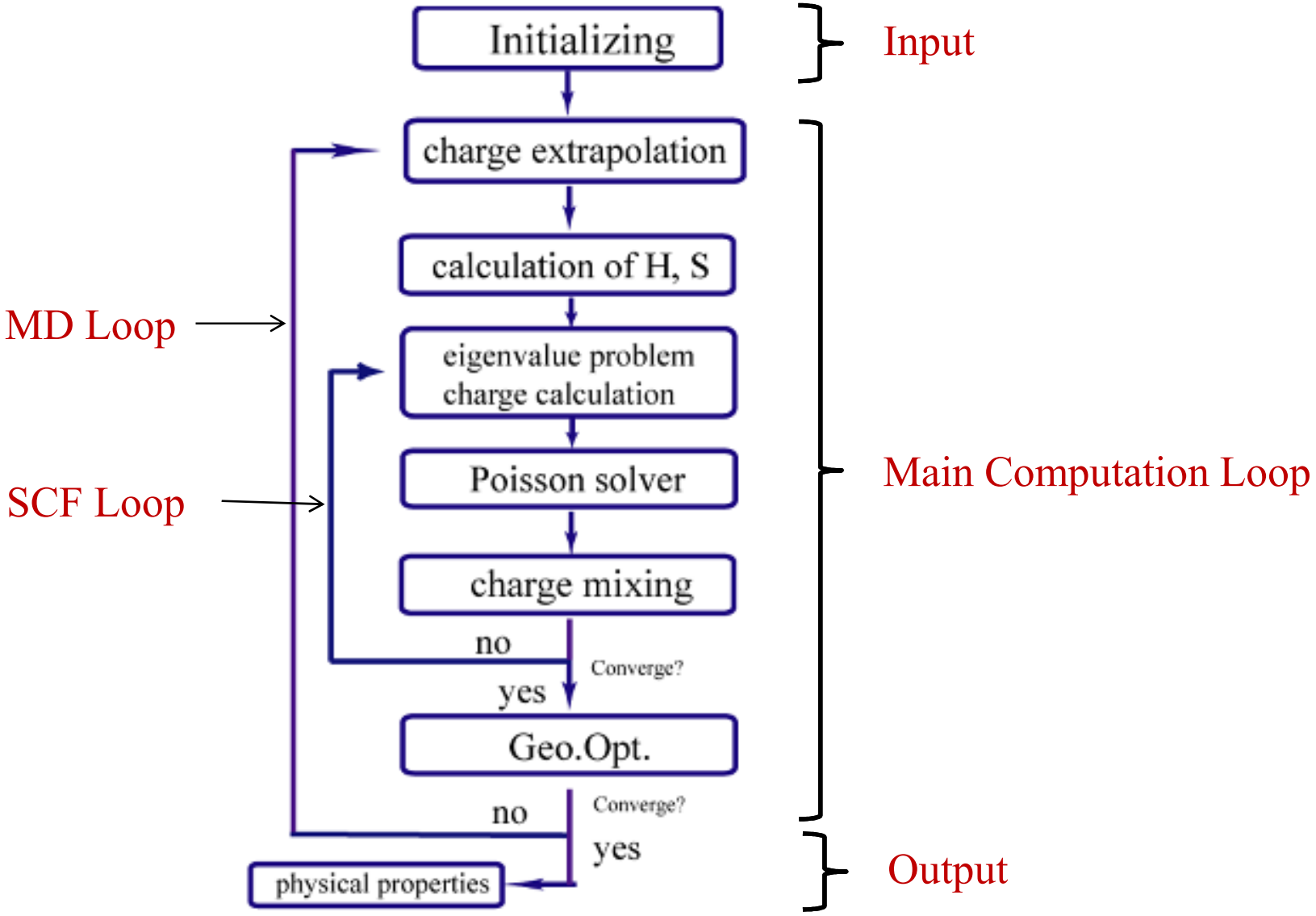
Mathematical Structure of KS Eq.

3D coupled non-linear differential equations have to be solved self-consistently.



Input charge = Output charge → Self-consistent condition

Flowchart of Calculation



Total Energy #1

The total energy is given by

$$E_{\mathrm{tot}} = E_{\mathrm{kin}} + E_{\mathrm{na}} + E_{\mathrm{ec}}^{(\mathrm{NL})} + E_{\delta\mathrm{ee}} + E_{\mathrm{xc}} + E_{\mathrm{scc}}.$$

Each term is evaluated by using a different numerical grid.

Spherical coordinate in momentum space

The kinetic energy

$$E_{\mathrm{kin}} = \sum_{\sigma} \sum_{\mathrm{n}}^{N} \sum_{i\alpha,j\beta} \rho_{\sigma,i\alpha j\beta}^{(\mathbf{R}_{\mathrm{n}})} h_{i\alpha j\beta,\mathrm{kin}}^{(\mathbf{R}_{\mathrm{n}})}.$$

The neutral atom energy

$$\begin{aligned} E_{\mathrm{na}} &= \int dr^3 n(\mathbf{r}) \sum_I V_{\mathrm{na},I}(\mathbf{r} - \tau_I), \\ &= \sum_{\sigma} \sum_{\mathrm{n}}^{N} \sum_{i\alpha,j\beta} \rho_{\sigma,i\alpha j\beta}^{(\mathbf{R}_{\mathrm{n}})} \sum_I \langle \phi_{i\alpha}(\mathbf{r} - \tau_i) | V_{\mathrm{na},I}(\mathbf{r} - \tau_I) | \phi_{j\beta}(\mathbf{r} - \tau_j - \mathbf{R}_{\mathrm{n}}) \rangle, \end{aligned}$$

The non-local electron-core Coulomb energy

$$E_{\mathrm{ec}}^{(\mathrm{NL})} = \sum_{\sigma} \sum_{\mathrm{n}}^{N} \sum_{i\alpha,j\beta} \rho_{\sigma,i\alpha j\beta}^{(\mathbf{R}_{\mathrm{n}})} \langle \phi_{i\alpha}(\mathbf{r} - \tau_i) | \sum_I V_{\mathrm{NL},I}(\mathbf{r} - \tau_I) | \phi_{j\beta}(\mathbf{r} - \tau_j - \mathbf{R}_{\mathrm{n}}) \rangle.$$

Total Energy #2

The total energy is given by

$$E_{\rm tot} = E_{\rm kin} + E_{\rm na} + E_{\rm ec}^{\rm (NL)} + E_{\delta\rm ee} + E_{\rm xc} + E_{\rm scc}.$$

Each term is evaluated by using a different numerical grid.

Real space regular mesh

Difference charge Hartree energy

$$E_{\delta\rm ee} = \frac{1}{2}\int dr^3 \delta n(\mathbf{r})\delta V_{\rm H}(\mathbf{r}),$$

The exchange-correlation energy

$$E_{\rm xc} = \int dr^3 \{n_\uparrow(\mathbf{r}) + n_\downarrow(\mathbf{r}) + n_{\rm pcc}(\mathbf{r})\}\epsilon_{\rm xc}(n_\uparrow + \frac{1}{2}n_{\rm pcc}, n_\downarrow + \frac{1}{2}n_{\rm pcc}),$$

Real space fine mesh

Screened core-core repulsion energy

$$E_{\rm scc} = \frac{1}{2}\sum_{I,J}\left[\frac{Z_I Z_J}{|\tau_I - \tau_J|} - \int dr^3 n_I^{\rm (a)}(\mathbf{r})V_{{\rm H},J}^{\rm (a)}(\mathbf{r})\right].$$

Forces

$$\mathbf{F}_i = -\frac{\partial E_{\rm tot}}{\partial \mathbf{R}_i} = -\frac{\partial E_{\rm kin}}{\partial \mathbf{R}_i} - \frac{\partial E_{\rm na}}{\partial \mathbf{R}_i} - \frac{\partial E_{\rm scc}}{\partial \mathbf{R}_i} - \frac{\partial E_{\delta\rm ee}}{\partial \mathbf{R}_i} - \frac{\partial E_{\rm xc}}{\partial \mathbf{R}_i} - \frac{\partial E_{\rm cc}}{\partial \mathbf{R}_i}$$

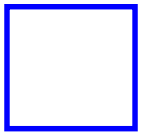
$$\frac{\partial E_{\delta\rm ee}}{\partial \mathbf{R}_k} = \sum_{\rm p} \frac{\partial n(\mathbf{r}_{\rm p})}{\partial \mathbf{R}_k} \frac{\partial E_{\delta\rm ee}}{\partial n(\mathbf{r}_{\rm p})} + \sum_{\rm p} \frac{\partial n^{\rm a}(\mathbf{r}_{\rm p})}{\partial \mathbf{R}_k} \frac{\partial E_{\delta\rm ee}}{\partial n^{\rm a}(\mathbf{r}_{\rm p})}.$$

$$\begin{aligned}
\frac{\partial E_{\delta\rm ee}}{\partial n(\mathbf{r}_{\rm p})} &= \frac{1}{2}\Delta V\{\delta V_{\rm H}(\mathbf{r}_{\rm p}) + \sum_{\rm q} \delta n(\mathbf{r}_{\rm q}) \frac{\partial \delta V_{\rm H}(\mathbf{r}_{\rm q})}{\partial n(\mathbf{r}_{\rm p})}\}, \\
&= \frac{1}{2}\Delta V\{\delta V_{\rm H}(\mathbf{r}_{\rm p}) + \frac{4\pi}{N_{\rm rsg}} \sum_{\rm G} \frac{1}{|\mathbf{G}|^2} \sum_{\rm q} \delta n(\mathbf{r}_{\rm q}) e^{i\mathbf{G}\cdot(\mathbf{r}_{\rm q}-\mathbf{r}_{\rm p})}\}, \\
&= \Delta V \delta V_{\rm H}(\mathbf{r}_{\rm p}).
\end{aligned}$$

$$\begin{aligned}
\frac{\partial E_{\delta\rm ee}}{\partial n^{\rm a}(\mathbf{r}_{\rm p})} &= -\frac{1}{2}\Delta V\{\delta V_{\rm H}(\mathbf{r}_{\rm p}) - \sum_{\rm q} \delta n(\mathbf{r}_{\rm q}) \frac{\partial \delta V_{\rm H}(\mathbf{r}_{\rm q})}{\partial n^{\rm a}(\mathbf{r}_{\rm p})}\}, \\
&= -\frac{1}{2}\Delta V\{\delta V_{\rm H}(\mathbf{r}_{\rm p}) + \frac{4\pi}{N_{\rm rsg}} \sum_{\rm G} \frac{1}{|\mathbf{G}|^2} \sum_{\rm q} \delta n(\mathbf{r}_{\rm q}) e^{i\mathbf{G}\cdot(\mathbf{r}_{\rm q}-\mathbf{r}_{\rm p})}\}, \\
&= -\Delta V \delta V_{\rm H}(\mathbf{r}_{\rm p}).
\end{aligned}$$

$$\begin{aligned}
\frac{\partial n(\mathbf{r}_{\rm p})}{\partial \mathbf{R}_k} &= \sum_{i\alpha,j\beta} \sum_{\nu} \{\frac{\partial c^*_{i\alpha,\nu}}{\partial \mathbf{R}_k} c_{j\beta,\nu} \chi_{i\alpha}(\mathbf{r}) \chi_{j\beta}(\mathbf{r}) + c^*_{i\alpha,\nu} \frac{\partial c_{j\beta,\nu}}{\partial \mathbf{R}_k} \chi_{i\alpha}(\mathbf{r}_{\rm p}) \chi_{j\beta}(\mathbf{r}_{\rm p})\} \\
&\quad + 2 \sum_{\alpha,j\beta} \rho_{k\alpha,j\beta} \frac{\partial \chi_{k\alpha}(\mathbf{r}_{\rm p})}{\partial \mathbf{R}_k} \chi_{j\beta}(\mathbf{r}_{\rm p}).
\end{aligned}$$

$$\begin{aligned}
\frac{\partial E_{\rm xc}}{\partial \mathbf{R}_k} &= \sum_{\rm p} \frac{\partial n(\mathbf{r}_{\rm p})}{\partial \mathbf{R}_k} \frac{\partial E_{\rm xc}}{\partial n(\mathbf{r}_{\rm p})}, \\
&= \Delta V \sum_{\rm p} \frac{\partial n(\mathbf{r}_{\rm p})}{\partial \mathbf{R}_k} v_{\rm xc}(n(\mathbf{r}_{\rm p})).
\end{aligned}$$

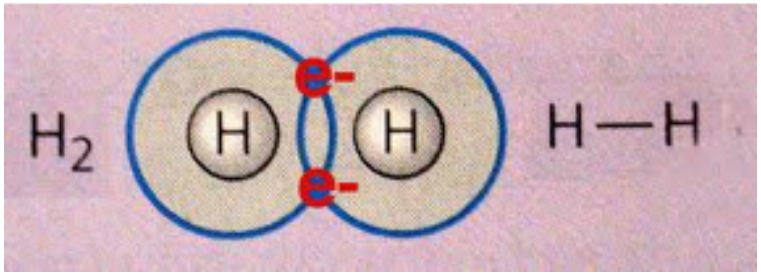


Easy calc.

See the left

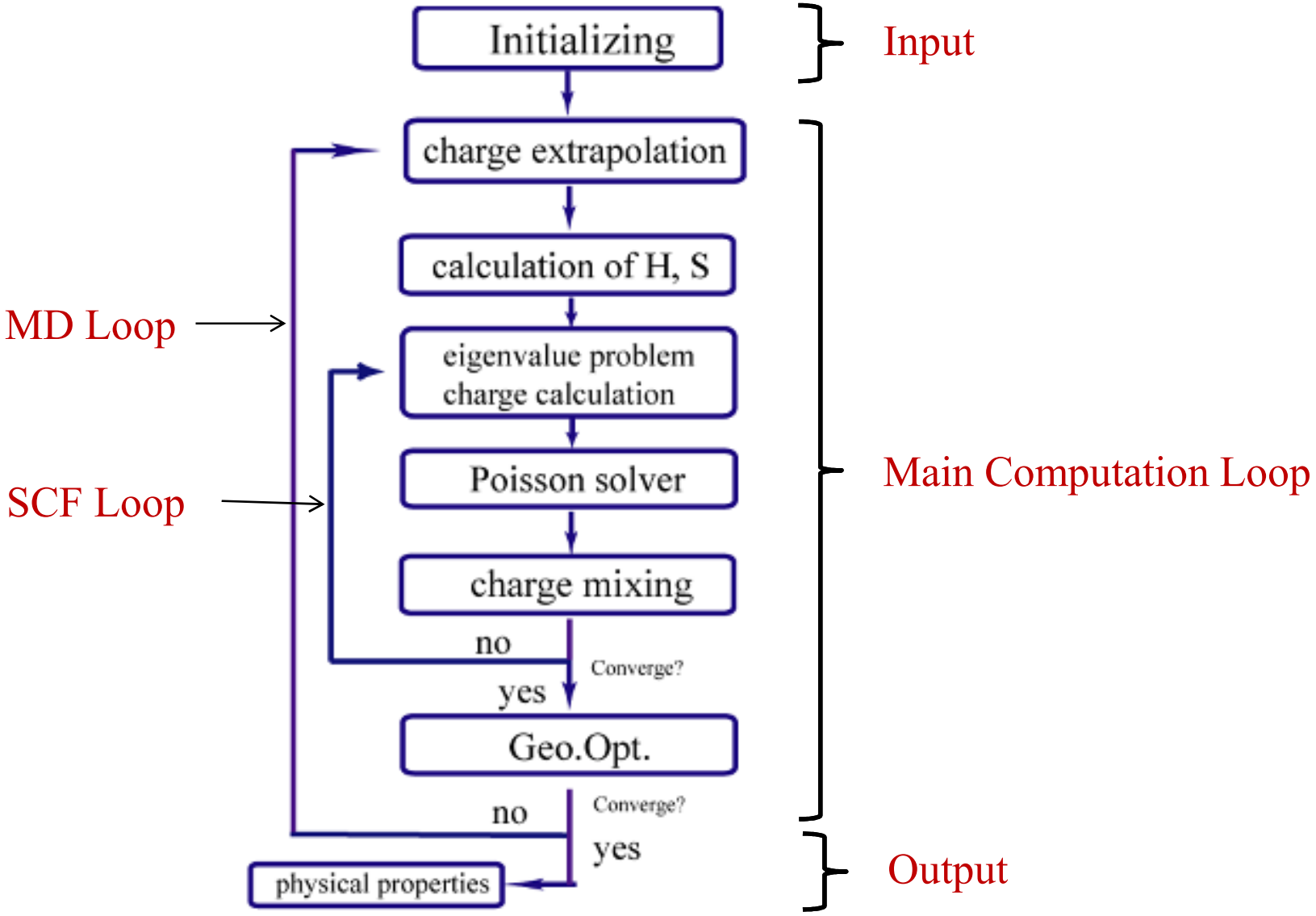
Forces are always analytic at any grid fineness and at zero temperature, even if numerical basis functions and numerical grids.

Hydrogen Molecule

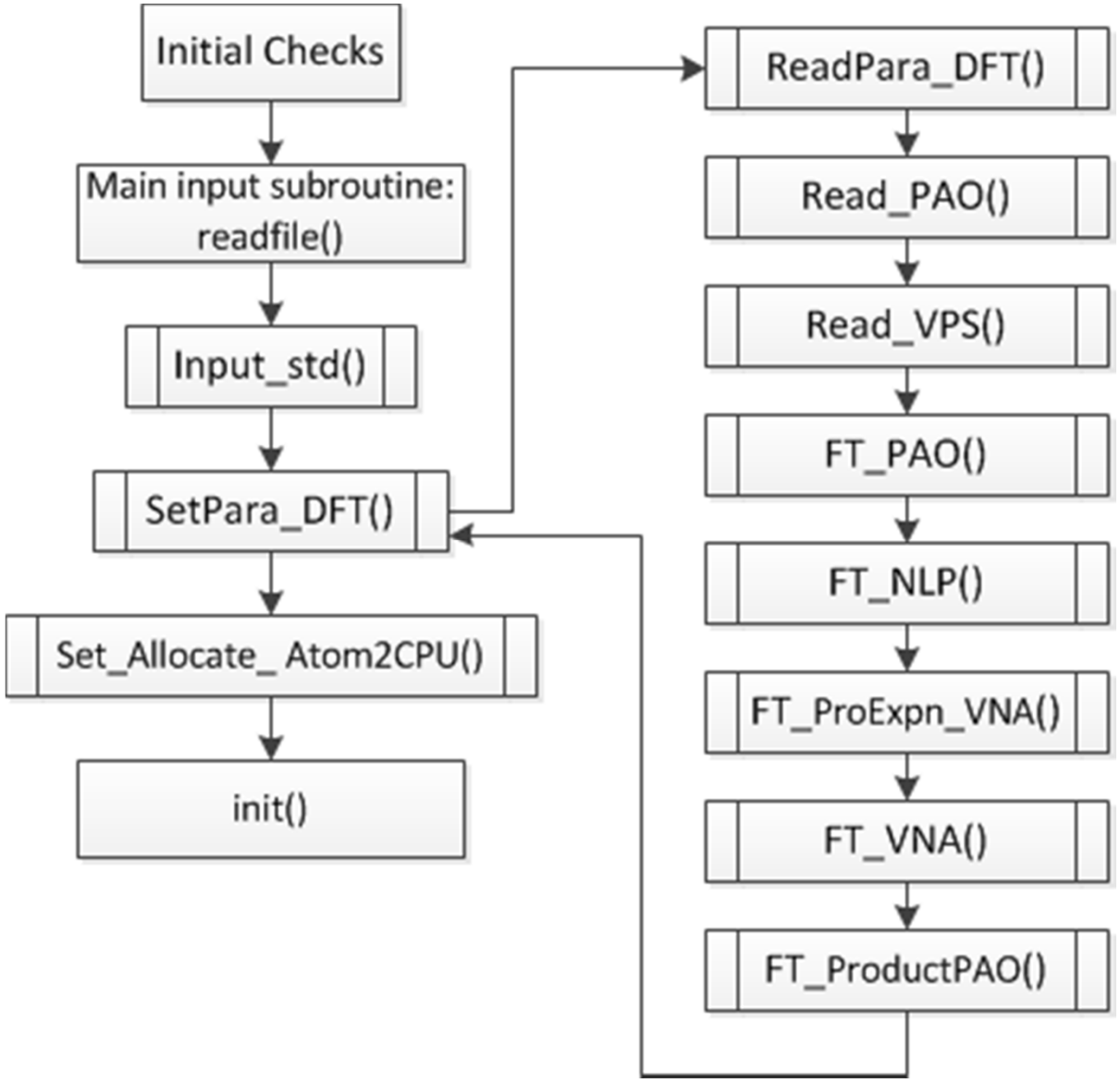


- **Version of OpenMX: the latest version of 3.6.**
- **Input file: openmx3.6/work/force_example/H2_LDA.dat.**
- **H H5.0-s1 H_CA11**
- **scf.XcType LDA**
- **scf.energycutoff 100.0**
- **scf.EigenvalueSolver cluster**
- **Execution hosts: Cray XT5 of JAIST.**
- **The number of MPI processes: 1 and 2. Results are presented for the first case only.**
- **Program profiler: Craypat and HPCToolkit.**

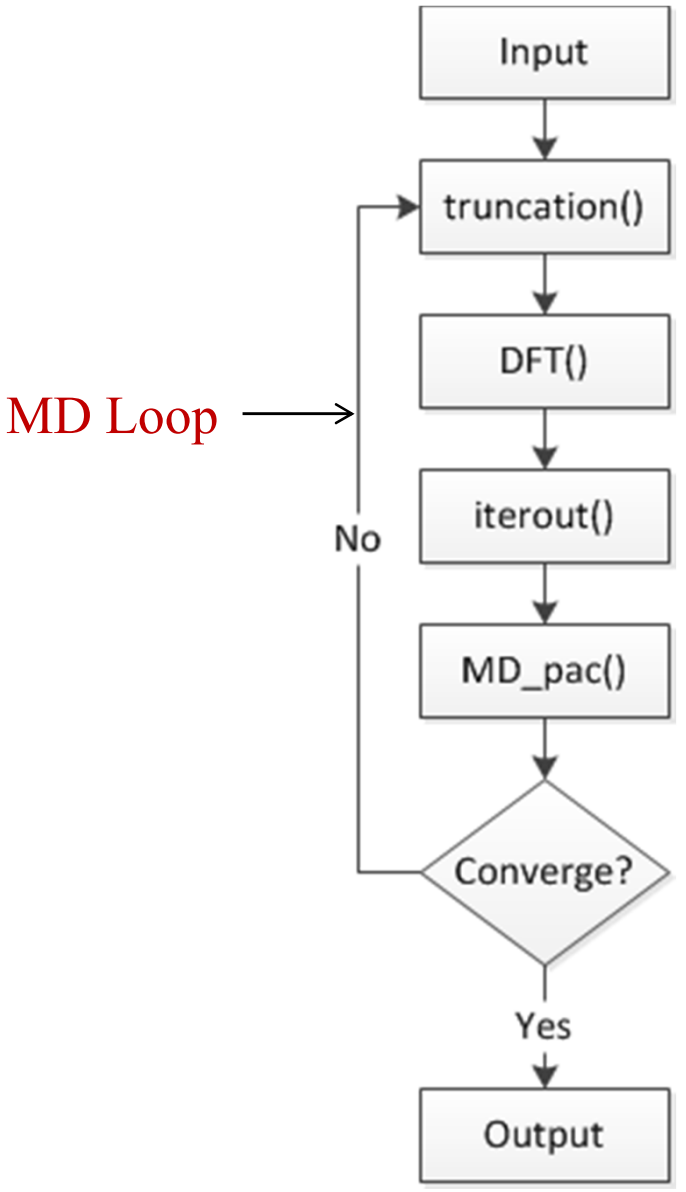
Flowchart of Calculation



Input


Initial Checks
Main input subroutine: readfile()
Input_std()
SetPara_DFT()
Set_Allocate_ Atom2CPU()
init()
ReadPara_DFT()
Read_PAO()
Read_VPS()
FT_PAO()
FT_NLP()
FT_ProExpn_VNA()
FT_VNA()
FT_ProductPAO()

Main Loop: MD



Divide the system into smaller systems and set grid data

SCF loop is performed within DFT

Output xyz-coordinates at each MD step to files

Perform MD simulations and geometry optimization

Main Loop: SCF

Calculate the overlap matrix *OLP* and the matrix for kinetic operator *H0* for *T*

Calculate the matrices *HNL*, *DS_NL,* and derivatives of nonlocal potentials

Calculate the matrices *HVNA* , *HVNA2*, *DS_VNA* and derivatives of VNA projector expansion

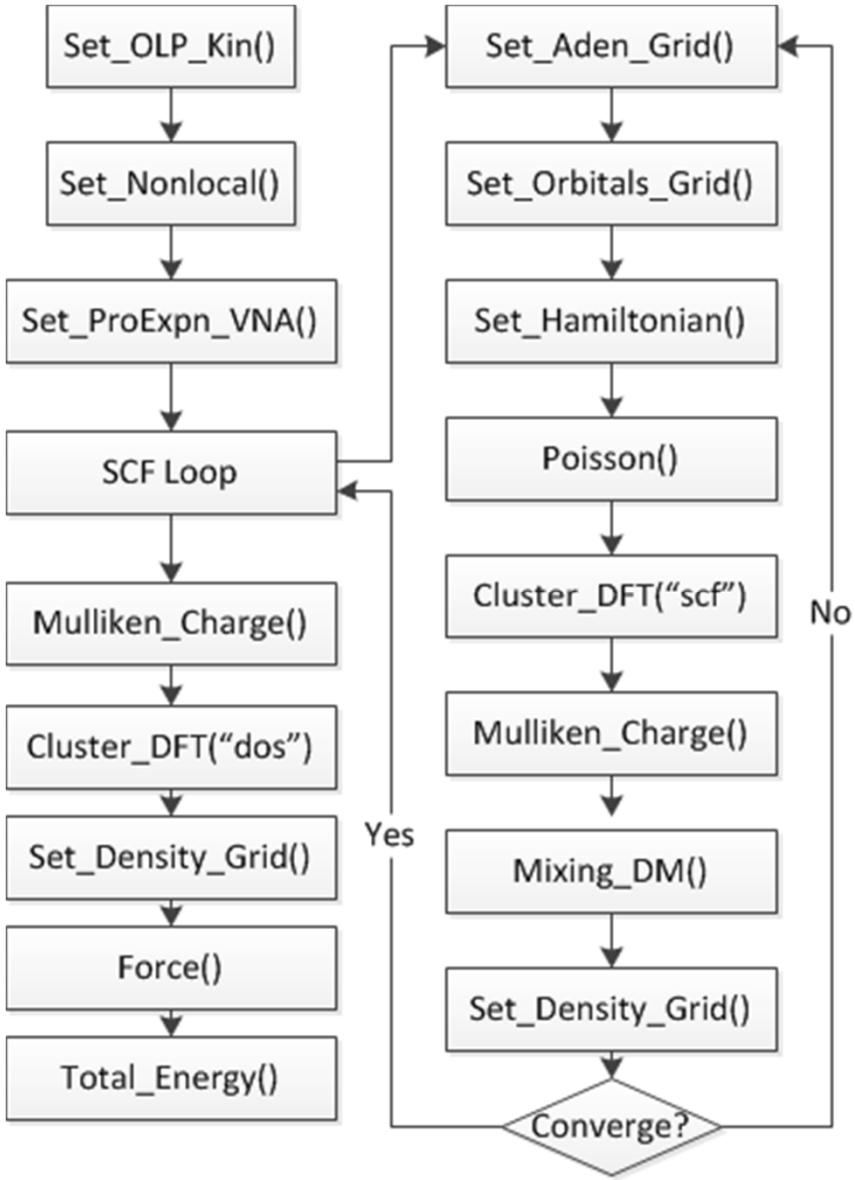
Calculate Mulliken charge

Perform cluster calculation in terms of the density of state

Calculate charge density on grid by one-particle wave function *Density_Grid*, *Corbs_Grid*

Force calculation

Total energy calculation



Calculate elec. den. *Density_Grid*, by atom. den *Adensity_Grid*, by partial core correction den. *PCCDensity_Grid*

Calculate values of basic orbitals on grids *Orbs_Grid*

Calculate the KS matrix *H* = *H0+HNL+ dVH+Vxc + HVNA*

Solve Poisson's equation

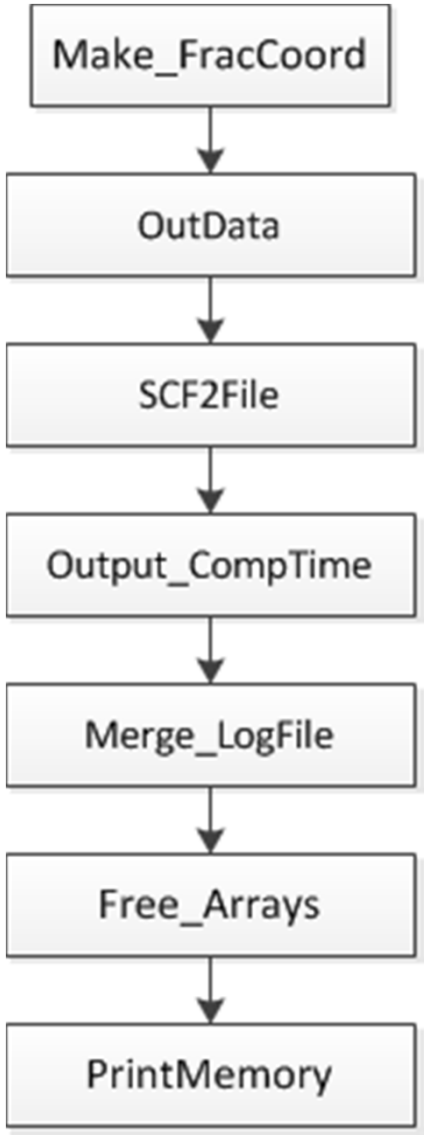
Perform cluster calculation and solve the eigenvalue problem with full overlap matrix *S*

Calculate Mulliken charge

Mix current and old density matrices *ResidualDM* and *iResidualDM*

Calculate charge density on grid by one-particle wave function *Density_Grid*, *Corbs_Grid*

Output


Make_FracCoord
OutData
SCF2File
Output_CompTime
Merge_LogFile
Free_Arrays
PrintMemory

Results

```
****************** MD= 1  SCF= 1 ******************
  Mc_AN=  1 Gc_AN=  1 ocupcy_u= 0.500000000000 ocupcy_d= 0.500000000000
  Mc_AN=  2 Gc_AN=  2 ocupcy_u= 0.500000000000 ocupcy_d= 0.500000000000
<Cluster>  Solving the eigenvalue problem...
  Eigenvalues of OLP   1   0.240978994092259
  Eigenvalues of OLP   2   1.759020904003656
  Eigenvalues of Kohn-Sham  1 -0.445643207069 -0.445643207069
  Eigenvalues of Kohn-Sham  2  0.187811903205  0.187811903205
ChemP= 0.000000000000 TZ= 2.000000000000 Num_state= 2.000000000000
  ChemP= 0.000000000000
  HOMO =  1
    1    H  MulP   0.5000  0.5000 sum   1.0000
    2    H  MulP   0.5000  0.5000 sum   1.0000
 Sum of MulP: up   =     1.00000 down         =     1.00000
            total=     2.00000 ideal(neutral)=     2.00000
<DFT>  Total Spin Moment (muB) =  0.000000000000
<DFT>  Mixing_weight= 0.000100000000
<DFT>  Uele   =   -0.891286414137  dUele     =   1.000000000000
<DFT>  NormRD =    1.000000000000  Criterion =   0.000000000100
```

Results

```
****************** MD= 1  SCF= 2 ******************
<Poisson>  Poisson's equation using FFT...
<Set_Hamiltonian>  Hamiltonian matrix for VNA+dVH+Vxc...
<Cluster>  Solving the eigenvalue problem...
  Eigenvalues of Kohn-Sham  1 -0.424793245755 -0.424793245755
  Eigenvalues of Kohn-Sham  2  0.206931358315  0.206931358315
ChemP= 0.000000000000 TZ= 2.000000000000 Num_state= 2.000000000000
  ChemP= 0.000000000000
  HOMO =  1
    1    H  MulP   0.5000  0.5000 sum   1.0000
    2    H  MulP   0.5000  0.5000 sum   1.0000
 Sum of MulP: up   =     1.00000 down         =     1.00000
              total=     2.00000 ideal(neutral)=     2.00000
<DFT>  Total Spin Moment (muB) =  0.000000000000
<DFT>  Mixing_weight= 0.000100000000
<DFT>  Uele   =   -0.849586491510  dUele     =   0.041699922627
<DFT>  NormRD =    0.000000000000  Criterion =   0.000000000100
```

Results

```
******************** MD= 1  SCF= 3 ********************
<Poisson>  Poisson's equation using FFT...
<Set_Hamiltonian>  Hamiltonian matrix for VNA+dVH+Vxc...
<Cluster>  Solving the eigenvalue problem...
  Eigenvalues of Kohn-Sham  1 -0.424793245755 -0.424793245755
  Eigenvalues of Kohn-Sham  2  0.206931358315  0.206931358315
ChemP= 0.000000000000 TZ= 2.000000000000 Num_state= 2.000000000000
  ChemP= 0.000000000000
  HOMO =  1
    1    H  MulP   0.5000  0.5000 sum   1.0000
    2    H  MulP   0.5000  0.5000 sum   1.0000
 Sum of MulP: up   =     1.00000 down         =     1.00000
              total=     2.00000 ideal(neutral)=     2.00000
<DFT>  Total Spin Moment (muB) =  0.000000000000
<DFT>  Mixing_weight= 0.000100000000
<DFT>  Uele   =   -0.849586491510  dUele     =   0.000000000000
<DFT>  NormRD =    0.000000000000  Criterion =   0.000000000100
```

Results

```
<MD= 1>  Force calculation
  Force calculation #1
<Force>  force(1) myid= 0  Mc_AN= 1 Gc_AN= 1   0.016934198825  0.012697431438 -0.021158048413
<Force>  force(1) myid= 0  Mc_AN= 2 Gc_AN= 2  -0.016934198825 -0.012697431438  0.021158048413
  Force calculation #2
<Force>  force(2) myid= 0  Mc_AN= 1 Gc_AN= 1   0.192004840585  0.144003630439 -0.240006050731
<Force>  force(2) myid= 0  Mc_AN= 2 Gc_AN= 2  -0.192004840585 -0.144003630439  0.240006050731
  Force calculation #3
<Force>  force(3) myid= 0  Mc_AN= 1 Gc_AN= 1  -0.088256376434 -0.066180645714  0.110312051520
<Force>  force(3) myid= 0  Mc_AN= 2 Gc_AN= 2   0.088256376434  0.066180645714 -0.110312051520
  Force calculation #4
<Force>  force(4B) myid= 0  Mc_AN= 1 Gc_AN= 1  -0.345886901541 -0.259415167684  0.432358618455
<Force>  force(4B) myid= 0  Mc_AN= 2 Gc_AN= 2   0.345886901541  0.259415167684 -0.432358618455
  Force calculation #5
<Force>  force(5) myid= 0  Mc_AN= 1 Gc_AN= 1   0.078269566025  0.058702174519 -0.097836957531
<Force>  force(5) myid= 0  Mc_AN= 2 Gc_AN= 2  -0.078269566025 -0.058702174519  0.097836957531
```

Results

```
<MD= 1>  Total Energy
  Force calculation #6
<Total_Ene>  force(6) myid= 0  Mc_AN= 1 Gc_AN= 1   0.316816150895  0.237612113171 -
0.396020188619
<Total_Ene>  force(6) myid= 0  Mc_AN= 2 Gc_AN= 2  -0.316816150895 -0.237612113171
0.396020188619
<Calc_EH0> A spe= 0 1D-grids=127 3D-grids=19050
<Calc_EH0> B spe= 0 1D-grids=127 3D-grids=19050
  Force calculation #7
<Total_Ene>  force(7) myid= 0  Mc_AN= 1 Gc_AN= 1  -0.071184495454 -0.053388371590
0.088980619317
<Total_Ene>  force(7) myid= 0  Mc_AN= 2 Gc_AN= 2   0.071184495454  0.053388371590 -
0.088980619317
<Total_Ene>  force(8) myid= 0  Mc_AN= 1 Gc_AN= 1  -0.026297677244 -0.019723255253
0.032872093133
<Total_Ene>  force(8) myid= 0  Mc_AN= 2 Gc_AN= 2   0.026297676122  0.019723259771 -
0.032872093024
<Total_Ene>  force(t) myid= 0 Gc_AN= 1   0.072399305658  0.054307909326 -0.090497862870
<Total_Ene>  force(t) myid= 0 Gc_AN= 2  -0.072399306780 -0.054307904808  0.090497862979
```

Results

```
*****************************************************
            Total Energy (Hartree) at MD = 1
*****************************************************

 Uele  =     -0.849586491510

 Ukin  =      0.867294836322
 UH0   =     -1.126624865574
 UH1   =      0.002131136124
 Una   =     -1.143875598977
 Unl   =     0.153152502541
 Uxc0  =     -0.297105686293
 Uxc1  =     -0.297105686293
 Ucore =      0.748369642435
 Uhub  =      0.000000000000
 Ucs   =      0.000000000000
 Uzs   =      0.000000000000
 Uzo   =      0.000000000000
 Uef   =     0.000000000000
 UvdW  =      0.000000000000
 Utot  =   -1.09376371971380
```

Uele: band energy
Ukin: kinetic energy
UH0: electric part of screened Coulomb energy
UH1: difference electron-electron Coulomb energy
Una: neutral atom potential energy
Unl: non-local potential energy
Uxc0: exchange-correlation energy for alpha spin
Uxc1: exchange-correlation energy for beta spin
Ucore: core-core Coulomb energy
Uhub: LDA+U energy
Ucs: constraint energy for spin orientation
Uzs: Zeeman term for spin magnetic moment
Uzo: Zeeman term for orbital magnetic moment
Uef: electric energy by electric field
UvdW: semi-empirical vdW energy

Note:

Utot = Ukin+UH0+UH1+Una+Unl+Uxc0+Uxc1+Ucore+Uhub+Ucs+Uzs+Uzo+Uef+UvdW

Conclusion

- A look at OpenMX calltree
- Some important subroutines
  - Input
  - MD loop and SCF loop
  - Output
- Many more things to learn!
  - Go deeper at each subroutine
  - Domain decomposition in truncation()
  - Performance model
  - etc.