M-SPARC

User guide

Material Physics & Mechanics Group, Georgia Tech
PI: Phanish Suryanarayana
Contributors
Citation
Acknowledgements

Introduction

Matlab-Simulation Package for Ab-initio Real-space Calculations (M-SPARC) is a real-space code for performing electronic structure calculations based on Kohn-Sham Density Functional Theory (DFT). Its primary purpose is the rapid development and testing of new algorithms and methods within DFT. The main features of M-SPARC v1.0.0 include

- Applicable to isolated systems such as molecules as well as extended systems such as crystals, surfaces, and wires.
- Local, semilocal, and nonlocal (including hybrid) exchange-correlation functionals.
- Standard ONCV pseudopotentials, including nonlinear core corrections.
- Calculation of ground state energy, atomic forces, and stress tensor.
- Structural relaxation and ab initio molecular dynamics (NVE, NVT, and NPT).
- Spin polarized and unpolarized calculations.

Introduction

- Spin-orbit coupling.
- Dispersion interactions through DFT-D3, vdW-DF1, and vdW-DF2.

Contributors

- **Phanish Suryanarayana** (PI)
  - **Qimen Xu**: Code infrastructure, SCF, Energy, Force, LDA
  - **Abhiraj Sharma**: Code infrastructure, k-points, PBE, Stress, Non-orthogonal, Relaxation
  - **Boqin Zhang**: vdW-DF, DFT-D3, SCAN
  - **Xin Jing**: Hybrid, SOC
  - **Shashikant Kumar**: Testing framework, NLCC

Citation

If you publish work using/regarding M-SPARC, please cite some of the following articles, particularly those that are most relevant to your work:

- General: https://doi.org/10.1016/j.softx.2020.100423
- Non-orthogonal systems:
  https://doi.org/10.1016/j.cplett.2018.04.018
- Linear solvers: https://doi.org/10.1016/j.cpc.2018.07.007,
  https://doi.org/10.1016/j.jcp.2015.11.018
- Stress tensor/pressure: https://doi.org/10.1063/1.5057355
- Atomic forces: https://doi.org/10.1016/j.cpc.2016.09.020,
  https://doi.org/10.1016/j.cpc.2017.02.019
- Mixing: https://doi.org/10.1016/j.cplett.2016.01.033,
  https://doi.org/10.1016/j.cplett.2015.06.029,
  https://doi.org/10.1016/j.cplett.2019.136983

Acknowledgments

Installation

Prerequisite: Matlab
No installation required.

Input files

The required input files to run a simulation with M-SPARC are
- ".inpt" file – User options and parameters.
- ".ion" file – Atomic information.

It is required that the ".inpt" and ".ion" files are located in the same directory and share the same name. A detailed description of the input options is provided in this document. Examples of input files can be found in the directory `M-SPARC/tests`.

In addition, M-SPARC requires pseudopotential files of psp8 format which can be generated by D. R. Hamann's open-source pseudopotential code ONCVPSP. A large number of accurate and efficient pseudopotentials are already provided within the package. For access to more pseudopotentials, the user is referred to the SG15 ONCV potentials. Using the ONCVPSP input files included in the SG15 ONCV potentials, one can easily convert the SG15 ONCV potentials from upf format to psp8 format. Paths to the pseudopotential files are specified in the ".ion" file.

Execution

M-SPARC can be executed in MATLAB by calling the `msparc` function (which is located under `src/` directory). It is required that the ".inpt" and ".ion" files are located in the same directory and share the same name. For example, to run a simulation with input files as "filename.inpt" and "filename.ion" in the `src/` directory, use the following command:

```
S = msparc('filename');
```

In many cases, we would not want to put the input files inside the `src/` directory. In such cases, we need to provide the path to the input file name, without any extension. As an example, one can run a test located in `M-SPARC/tests/Example_tests/`. First go to `src/` directory. Run a DC silicon system by:

```
S = msparc('../tests/Example_tests/Si8_kpt');
```

Execution

The result is printed to output file "Si8_kpt.out", located in the same directory as the input files. If the file "Si8_kpt.out" is already present, the result will be printed to "Si8_kpt.out_1" instead. The max number of ".out" files allowed with the same name is 100. Once this number is reached, the result will instead overwrite the "Si8_kpt.out" file. One can compare the result with the reference out file named "Si8_kpt.refout".

In the `tests/` directory, we also provide a sample script file `run_examples.m`, which launches four example tests one by one. To run these examples, simply change directory to `tests/examples/` directory, and run:

```
run_examples
```

Execution

Note that in this case, we're trying to call the `msparc` function from a different directory. This is achieved by using the MATLAB function `addpath` to add the `src/` directory to search path.
One can also run M-SPARC using the MATLAB parallel pool over k-points/spin by providing a second argument, `parallel_switch`, when running M-SPARC:

```
S = msparc('filename',parallel_switch);
```

If `parallel_switch` = 1, M-SPARC will start using the parallel pool, and if `parallel_switch` = 0, M-SPARC will not use the parallel pool, which is the default.
A suite of test systems is provided in the `tests/` directory. The test systems are arranged in a hierarchal systems of directories. Input and reference output files for each test system is stored in separate folders with the same name. A python script named 'test.py' is also provided to launch the tests on a cluster. Details on how to use the Python script can be found in `Readme` file in the `tests/` folder.

Output

Upon successful execution of the "S = msparc('filename');" command, an output structure is returned and stored in S. The structure S contains detailed information that can be useful for post-processing and debugging. Information such as the input parameters, densities, wavefunctions, eigenvalues, and all electronic ground-state properties calculated are stored in the output structure.

Apart from the output structure returned, depending on the calculations performed, some output files will be created in the same location as the input files too.

Single point calculations

- ".out" file – Contains general information about the test, including input parameters, SCF convergence progress, ground state properties and timing information.

- ".static" file – Contains the atomic positions and atomic forces if the user chooses to print these information..

Output

Structrual relaxation calculations

- ".out" file – See above.
- ".geopt" file – Contains the atomic positions and atomic forces for each relaxation step. This file is created only when the unit cell is fixed. For cell relaxation a '.cellopt' file is created instead.
- ".cellopt" file – Contains the cell information (lattice vectors, cell lengths, volume) and stresses for each relaxation step. Only created for cell relaxation.
- ".restart" file – Contains information necessary to perform a restarted structural relaxation calculation.

Molecular dynamics (MD) calculations

- ".out" file – See above.

Output

- ".aimd" file – Contains the atomic positions, atomic velocities, atomic forces, electronic temperature, ionic temperature and total energy for each MD step.
- ".restart" file – Contains information necessary to perform a restarted MD calculation.

Input file options

System

CELL | LATVEC_SCALE | LATVEC | FD_GRID | MESH_SPACING | ECUT | BC | FD_ORDER | EXCHANGE_CORRELATION | SPIN_TYP | KPOINT_GRID | KPOINT_SHIFT | ELEC_TEMP_TYPE | ELEC_TEMP | SMEARING | NSTATES | D3_FLAG | D3_RTHR | D3_CN_THR| VDWDF_GEN_KERNEL | EXX_RANGE_FOCK | EXX_RANGE_PBE | ATOM_TYPE | PSEUDO_POT | N_TYPE_ATOM | COORD | COORD_FRAC | RELAX | SPIN

Input file options

SCF

CHEB_DEGREE | RHO_TRIGGER | MAXIT_SCF | TOL_SCF | SCF_FORCE_ACC |
SCF_ENERGY_ACC | TOL_LANCZOS | MIXING_VARIABLE | MIXING_HISTORY |
MIXING_PARAMETER | MIXING_PARAMETER_SIMPLE |
MIXING_PARAMETER_MAG | MIXING_PARAMETER_SIMPLE_MAG |
PULAY_FREQUENCY | PULAY_RESTART | MIXING_PRECOND |
MIXING_PRECOND_MAG | TOL_PRECOND | PRECOND_KERKER_KTF |
PRECOND_KERKER_THRESH | PRECOND_KERKER_KTF_MAG |
PRECOND_KERKER_THRESH_MAG | TOL_FOCK | MAXIT_FOCK | MINIT_FOCK |
TOL_SCF_INIT | ACE_FLAG | EXX_METHOD | EXX_ACE_VALENCE_STATES |
EXX_DOWNSAMPLING | EXX_DIVERGENCE

Input file options

Electrostatics
`TOL_POISSON` | `TOL_PSEUDOCHARGE` | `REFERENCE_CUTOFF`

Stress calculation
`CALC_STRESS` | `CALC_PRES`

MD
`MD_FLAG` | `MD_METHOD` | `MD_NSTEP` | `MD_TIMESTEP` | `ION_TEMP` |
`ION_ELEC_EQT` | `RESTART_FLAG`

Input file options

Structural relaxation

`RELAX_FLAG` | `RELAX_METHOD` | `RELAX_NITER` | `TOL_RELAX` |
`TOL_RELAX_CELL` | `RELAX_MAXDIAL` | `NLCG_SIGMA` | `L_HISTORY` |
`L_FINIT_STP` | `L_MAXMOV` | `L_AUTOSCALE` | `L_LINEOPT` | `L_ICURV` |
`FIRE_DT` | `FIRE_MASS` | `FIRE_MAXMOV` | `RESTART_FLAG`

Print options

`PRINT_ATOMS` | `PRINT_FORCES` | `PRINT_MDOUT` | `PRINT_RELAXOUT` |
`PRINT_RESTART` | `PRINT_RESTART_FQ` | `PRINT_VELS` | `OUTPUT_FILE`

System: .inpt file

CELL

Type	Unit
Double	Bohr

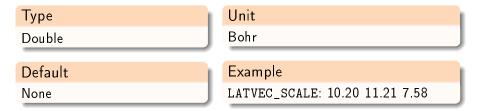| Default	Example
None	CELL: 10.20 11.21 7.58

Description

A set of three whitespace delimited values specifying the cell lengths in the lattice vector (`LATVEC`) directions, respectively.

Remark

Note that `CELL` ignores the lengths of the lattice vectors specified in the .inpt file and only treats them as unit vectors. `LATVEC_SCALE` and `CELL` cannot be specified simultaneously.

LATVEC_SCALE

Type	Unit
Double	Bohr

Default	Example
None	`LATVEC_SCALE`: 10.20 11.21 7.58

Description

A set of three whitespace delimited values specifying the scaling factors in the lattice vectors (`LATVEC`), respectively.

LATVEC_SCALE

> **Remark**
>
> The difference between `LATVEC_SCALE` and `CELL` is that `CELL` treats the lattice vectors as unit vectors, whereas `LATVEC_SCALE` scales the lattice vectors directly as specified by the user. `LATVEC_SCALE` and `CELL` cannot be specified simultaneously.

LATVEC

Type
Double array

Unit
No unit

Default
1.0 0.0 0.0
0.0 1.0 0.0
0.0 0.0 1.0

Example
LATVEC:
0.5 0.5 0.0
0.0 0.5 0.5
0.5 0.0 0.5

Description
A set of three vectors in row major order specifying the lattice vectors of the simulation domain (CELL).

FD_GRID

Type
Integer

Unit
No unit

Default
None

Example
`FD_GRID: 26 26 30`

Description
A set of three whitespace delimited values specifying the number of finite-difference intervals in the lattice vector (`LATVEC`) directions, respectively.

FD_GRID

> **Remark**
>
> The convergence of results with respect to spatial discretization needs to be verified. `ECUT`, `MESH_SPACING`, `FD_GRID` cannot be specified simultaneously.

MESH_SPACING

Type	Unit
Double	Bohr

Default	Example
None	MESH_SPACING: 0.4

Description

Mesh spacing of the finite-difference grid.

Remark

The exact mesh-size will be determined by the size of the domain.
MESH_SPACING, FD_GRID, ECUT cannot be specified simultaneously.

ECUT

Type	Unit
Double	Ha

Default	Example
None	ECUT: 30

Description

Equivalent plane-wave energy cutoff, based on which `MESH_SPACING` will be automatically calculated.

Remark

This is not exact, but rather an estimate. `ECUT`, `MESH_SPACING`, `FD_GRID` cannot be specified simultaneously.

BC

Type
Character

Unit
No unit

Default
None

Example
BC: P D D

Description
A set of three whitespace delimited characters specifying the boundary conditions in the lattice vector directions, respectively. P represents periodic boundary conditions and D represents Dirichlet boundary conditions.

FD_ORDER

Type	Unit
Integer	No unit

Default	Example
12	`FD_ORDER: 12`

Description

Order of the finite-difference approximation.

Remark

Restricted to even integers since central finite-differences are employed. The default value of 12 has been found to be an efficient choice for most systems.

Type

String

Unit

No unit

Default

No Default

Example

`EXCHANGE_CORRELATION: LDA_PW`

Description

Choice of exchange-correlation functional. Options are `LDA_PW` (Perdew-Wang LDA), `LDA_PZ` (Purdew-Zunger LDA), `GGA_PBE` (PBE GGA), `GGA_RPBE` (PBE RGGA), `GGA_PBEsol` (PBE GGAsol), `vdWDF1` (van der Waals Density Functional developed by Dion et al.), `vdWDF2` (vdW Density Functional modified by Lee et al), and `SCAN` (SCAN metaGGA).

EXCHANGE_CORRELATION

> **Remark**
>
> For spin-polarized calculation (`SPIN_TYP` $= 1$), `LDA_PZ` is not available.
> Before using `vdWDF1` or `vdWDF2`, please read the description and remark of
> `VDWDF_GEN_KERNEL`.

SPIN_TYP

Type
0 or 1

Unit
No unit

Default
0

Example
SPIN_TYP: 1

Description
SPIN_TYP: 0 performs spin unpolarized calculation.
SPIN_TYP: 1 performs unconstrained collinear spin-polarized calculation.

Remark
SPIN_TYP can only take values 0 and 1.

KPOINT_GRID

Type
Integer array

Unit
No unit

Default
1 1 1

Example
`KPOINT_GRID: 2 3 4`

Description
Number of k-points in each direction of the Monkhorst-Pack grid for Brillouin zone integration.

Remark
Time-reversal symmetry is assumed to hold.

KPOINT_SHIFT

Type
Double array

Unit
No unit

Default
0.0 0.0 0.0

Example
KPOINT_SHIFT: 0.5 0.5 0.5

Description
Shift of k-points in each direction of the k-point lattice.

Remark
The shift is in reduced coordinates. The default zero shift corresponds to the Monkhorst-Pack grid.

ELEC_TEMP_TYPE

Type
String

Unit
No unit

Default
gaussian

Example
ELEC_TEMP_TYPE: fd

Description
Function used for the smearing (electronic temperature). Options are:
fermi-dirac (or fd), gaussian.

Remark
Use ELEC_TEMP or SMEARING to set smearing value.

ELEC_TEMP

Type	Unit
Double	Kelvin

Default	Example
1160.452	`ELEC_TEMP: 315.773`

Description

Electronic temperature.

Remark

This is equivalent to setting `SMEARING` (0.001 Ha = 315.773 Kelvin).

SMEARING

Type
Double

Unit
Ha

Default
0.003675 for `gaussian`
0.007350 for `fermi-dirac`

Example
SMEARING: 0.001

Description
Value of smearing.

Remark
Equivalent to setting `ELEC_TEMP` (0.001 Ha = 315.773 Kelvin).

NSTATES

Type

Integer

Unit

No unit

Default

$N_e/2 \times 1.2 + 5$

Example

NSTATES: 24

Description

The number of Kohn-Sham states/orbitals.

Remark

This number should not be smaller than half of the total number of valence electrons (N_e) in the system. Note that the number of additional states required increases with increasing values of ELEC_TEMP/SMEARING.

D3_FLAG

Type	Unit
0 or 1	No unit

Default	Example
0	`D3_FLAG: 1`

Description

Flag for adding Grimme's DFT-D3 correction on the result

Remark

Only active when using GGA-PBE, GGA-RPBE and GGA-PBEsol.

D3_RTHR

Type	Unit
Double	Bohr^2

Default	Example
1600	`D3_RTHR: 9000`

Description

Square of cut-off radius for calculating DFT-D3 correction between two atoms

Remark

Only applicable when DFT-D3 correction `D3_FLAG` is used.

D3_CN_THR

Type	Unit
Double	$Bohr^2$

Default	Example
625	D3_CN_THR: 1600

Description

Square of cut-off radius for calculating CN value of every atom and DFT-D3 correction between three atoms

Remark

Only applicable when DFT-D3 correction D3_FLAG is used.

VDWDF_GEN_KERNEL

Type
0 or 1

Unit
No Unit

Default
0

Example
VDWDF_GEN_KERNEL: 1

Description
Flag for computing the kernel functions, its 2nd derivatives and the 2nd derivative of spline functions. If 0 is set, the program will read the functions (vdWDF_kernel.mat, vdWDF_d2Phidk2.mat and vdWDF_D2yDx2.mat in \vdW\vdWDF folder) directly; if 1 is set, the program will generate these functions before computation.

EXX_RANGE_FOCK

Type	Unit
Double	No unit

Default	Example
0.1587	`EXX_RANGE_FOCK: 0.106`

Description

Short range screen parameter of hartree-fock operator in HSE functional.

Remark

Default is using VASP's value. Different code has different parameters. Be careful with the results.

EXX_RANGE_PBE

Type	Unit
Double	No unit

Default	Example
0.1587	`EXX_RANGE_PBE: 0.106`

Description

Short range screen parameter of PBE in HSE functional.

Remark

Default is using VASP's value. Different code has different parameters. Be careful with the results.

System: .ion file

ATOM_TYPE

Type

String

Unit

No unit

Default

None

Example

`ATOM_TYPE: Fe`

Description

Atomic type symbol.

Remark

The atomic type symbol can be attached with a number, e.g., Fe1 and Fe2. This feature is useful if one needs to provide two different pseudopotential files (`PSEUDO_POT`) for the same element.

PSEUDO_POT

Type
String

Unit
No unit

Default
None

Example
PSEUDO_POT: ../psp/Fe.psp8

Description
Path to the pseudopotential file.

Remark
The default directory for the pseudopotential files is the same as the input files. For example, if a pseudopotential Fe.psp8 is put in the same directory as the input files, one can simply specify PSEUDO_POT: Fe.psp8.

N_TYPE_ATOM

Type
Integer

Unit
No unit

Default
None

Example
N_TYPE_ATOM: 2

Description
The number of atoms of a `ATOM_TYPE` specified immediately before this variable.

Remark
For a system with different types of atoms, one has to specify the number of atoms for every type.

COORD

Type	Unit
Double	Bohr

Default	Example
None	COORD:
	0.0 0.0 0.0
	2.5 2.5 2.5

Description

The Cartesian coordinates of atoms of a ATOM_TYPE specified before this variable. If the coordinates are outside the fundamental domain (see CELL and LATVEC) in the periodic directions (see BC), it will be automatically mapped back to the domain.

COORD

> **Remark**
>
> For a system with different types of atoms, one has to specify the coordinates for every `ATOM_TYPE`. One can also specify the coordinates of the atoms using `COORD_FRAC`.

COORD_FRAC

Type	Unit
Double	None

Default	Example
None	COORD_FRAC:
	0.5 0.5 0.0
	0.0 0.5 0.5

Description

The fractional coordinates of atoms of a `ATOM_TYPE` specified before this variable. `COORD_FRAC`$(i, j) \times$ `CELL`(j), $(j = 1, 2, 3)$ gives the coordinate of the ith atom along the jth `LATVEC` direction. If the coordinates are outside the fundamental domain (see `CELL` and `LATVEC`) in the periodic directions (see `BC`), it will be automatically mapped back to the domain.

COORD_FRAC

> **Remark**
>
> For a system with different types of atoms, one has to specify the coordinates for every `ATOM_TYPE`. One can also specify the coordinates of the atoms using `COORD`.

RELAX

Type
Integer

Unit
No unit

Default
1 1 1

Example
```
RELAX:
1 0 1
0 1 0
```

Description
Atomic coordinate with the corresponding `RELAX` value 0 is held fixed during relaxation/MD.

SPIN

Type
Double

Unit
No unit

Default
0.0

Example
SPIN:
1.0
-1.0

Description
Specifies the net initial spin on each atom for a spin-polarized calculation.

SCF

CHEB_DEGREE

Type
Integer

Unit
No unit

Default
Automatically set.

Example
`CHEB_DEGEE: 25`

Description
Degree of polynomial used for Chebyshev filtering.

Remark
For larger mesh-sizes, smaller values of `CHEB_DEGREE` are generally more efficient, and vice-versa.

RHO_TRIGGER

Type
Integer

Unit
No unit

Default
4

Example
RHO_TRIGGER: 5

Description
The number of times Chebyshev filtering is repeated before updating the electron density in the very first SCF iteration.

Remark
Values smaller than the default value of 4 can result in a significant increase in the number of SCF iterations. Larger values can sometimes reduce the number of SCF iterations.

MAXIT_SCF

Type
Integer

Unit
No unit

Default
100

Example
`MAXIT_SCF: 50`

Description
Maximum number of SCF iterations.

Remark
Larger values than the default of 100 may be required for highly inhomogeneous systems, particularly when small values of `SMEARING`/`ELEC_TEMP` are chosen.

TOL_SCF

Type
Double

Unit
No unit

Default
Automatically set for 10^{-3} Ha/Bohr accuracy in force

Example
TOL_SCF: 1e-5

Description
The tolerance on the normalized residual of the effective potential or the electron density for convergence of the SCF iteration.

TOL_SCF

> **Remark**
>
> Only one of `TOL_SCF`, `SCF_ENERGY_ACC`, or `SCF_FORCE_ACC` can be specified.

SCF_FORCE_ACC

Type	Unit
Double	Ha/Bohr

Default	Example
None	`SCF_FORCE_ACC: 1e-4`

Description

The tolerance on the atomic forces for convergence of the SCF iteration.

Remark

Only one of `SCF_FORCE_ACC`, `TOL_SCF` or `SCF_ENERGY_ACC` can be specified.

SCF_ENERGY_ACC

Type	Unit
Double	Ha/atom

Default	Example
None	`SCF_ENERGY_ACC: 1e-5`

Description

The tolerance on the free energy for the convergence of the SCF iteration.

Remark

Only one of `SCF_ENERGY_ACC`, `SCF_FORCE_ACC`, or `TOL_SCF` can be specified.

TOL_LANCZOS

Type	Unit
Double	No unit

Default	Example
1e-2	`TOL_LANCZOS: 1e-3`

Description

The tolerance within the Lanczos algorithm for calculating the extremal eigenvalues of the Hamiltonian, required as part of the CheFSI method.

Remark

Typically, the Lanczos tolerance does not need to be very strict.

MIXING_VARIABLE

Type	Unit
String	No unit

Default	Example
`potential`	`MIXING_VARIABLE: density`

Description

This specifies whether potential or density mixing is performed in the SCF iteration. Available options are: `potential` and `density`.

MIXING_HISTORY

Type	Unit
Integer	No unit

Default	Example
7	`MIXING_HISTORY: 40`

Description

The mixing history used in Pulay mixing.

Remark

Too small values of `MIXING_HISTORY` can result in poor SCF convergence.

MIXING_PARAMETER

Type	Unit
Double	No unit

Default	Example
0.3	`MIXING_PARAMETER: 0.1`

Description

The value of the relaxation parameter used in Pulay/simple mixing.

Remark

Values larger than the default value of 0.3 can be used for insulating systems, whereas smaller values are generally required for metallic systems, particularly at small values of SMEARING or ELEC_TEMP.

MIXING_PARAMETER_SIMPLE

Type

Double

Unit

No unit

Default

Automatically set to the same as MIXING_PARAMETER

Example

MIXING_PARAMETER_SIMPLE:
0.1

Description

The value of the relaxation parameter used in the simple mixing step in the periodic Pulay scheme.

MIXING_PARAMETER_MAG

Type

Double

Unit

No unit

Default

Automatically set to the same as `MIXING_PARAMETER`.

Example

`MIXING_PARAMETER_MAG`: 4.0

Description

The mixing parameter for the magnetization density in Pulay mixing for spin-polarized calculations.

MIXING_PARAMETER_MAG

> **Remark**
>
> For spin-polarized calculations, when SCF has difficulty to converge, increasing the mixing parameter to magnetization density might help. For example, setting it to 4.0, while turning off the preconditioner applied to the magnetization density (by setting `MIXING_PRECOND_MAG` to 'none') is a good choice.

MIXING_PARAMETER_SIMPLE_MAG

Type
Double

Unit
No unit

Default
Automatically set to the same as `MIXING_PARAMETER_MAG`

Example
```
MIXING_PARAMETER_SIMPLE_MAG:
4.0
```

Description
The value of the relaxation parameter for the magnetization density used in the simple mixing step in the periodic Pulay scheme for spin-polarized calculations.

PULAY_FREQUENCY

Type	Unit
Integer	No unit

Default	Example
1	`PULAY_FREQUENCY: 4`

Description

The frequency of Pulay mixing in Periodic Pulay.

Remark

The default value of 1 corresponds to Pulay mixing.

PULAY_RESTART

Type
Integer

Unit
No unit

Default
0

Example
PULAY_RESTART: 1

Description
The flag for restarting the 'Periodic Pulay' mixing. If set to 0, the restarted Pulay method is turned off.

MIXING_PRECOND

Type	Unit
String	No unit

Default	Example
none	`MIXING_PRECOND: kerker`

Description

This specifies the preconditioner used in the SCF iteration. Available options are: `none`, `kerker`, `resta` and `truncated_kerker`.

MIXING_PRECOND_MAG

Type	Unit
String	No unit

Default	Example
none	`MIXING_PRECOND_MAG: kerker`

Description

This specifies the preconditioner used for the magnetization density in the SCF iteration for spin-polarized calculations. Available options are: `none`, `kerker`.

TOL_PRECOND

Type	Unit
Double	No unit

Default	Example
$h^2 \times 0.001$	`TOL_PRECOND: 1e-4`

Description

The tolerance on the relative residual for the linear systems arising during the real-space preconditioning of the SCF.

Remark

The linear systems do not need to be solved very accurately. h is the mesh spacing.

PRECOND_KERKER_KTF

Type
Double

Unit
Bohr^{-1}

Default
1.0

Example
PRECOND_KERKER_KTF: 0.8

Description
The Thomas-Fermi screening length appearing in the kerker and truncated_kerker preconditioners (MIXING_PRECOND).

PRECOND_KERKER_THRESH

Type
Double

Unit
No unit

Default
0.25

Example
PRECOND_KERKER_THRESH: 0.1

Description
The threshold for the `truncated_kerker` preconditioner (`MIXING_PRECOND`).

Remark
This threshold will be scaled by the `MIXING_PARAMETER`. If the threshold is set to 0, the `kerker` preconditioner is recovered.

PRECOND_KERKER_KTF_MAG

Type	Unit
Double	Bohr^{-1}

Default	Example
1.0	`PRECOND_KERKER_KTF_MAG: 0.8`

Description

The Thomas-Fermi screening length appearing in the `kerker` preconditioner for the magnetization density (`MIXING_PRECOND_MAG`).

PRECOND_KERKER_THRESH_MAG

Type	Unit
Double	No unit

Default	Example
0.1	`PRECOND_KERKER_THRESH_MAG: 0.0`

Description

The threshold for the `kerker` preconditioner the magnetization density (`MIXING_PRECOND_MAG`).

Remark

This threshold will be scaled by the `MIXING_PARAMETER_MAG`. If the threshold is set to 0, the original `kerker` preconditioner is recovered.

TOL_FOCK

Type
Double

Unit
No unit

Default
0.2*TOL_SCF

Example
TOL_FOCK: 1e-6

Description
The tolerance on the Hartree-Fock outer loop, measured by the exact exchange energy difference per atom in 2 consecutive outer loops.

Remark
Only active when using hybrid functionals, like PBE0 and HSE.

MAXIT_FOCK

Type
Integer

Unit
No unit

Default
20

Example
MAXIT_FOCK: 50

Description
The maximum number of iterations for Hartree-Fock outer loop.

Remark
Only active when using hybrid functionals, like PBE0 and HSE.

MINIT_FOCK

Type
Integer

Unit
No unit

Default
2

Example
MINIT_FOCK: 3

Description
The minimum number of iterations for Hartree-Fock outer loop.

Remark
Only active when using hybrid functionals, like PBE0 and HSE.

TOL_SCF_INIT

Type
Double

Unit
No unit

Default
max(TOL_FOCK×10, 0.001)

Example
TOL_SCF_INIT: 1e-6

Description
The initial SCF tolerance for PBE iteration when using hybrid functionals.

Remark
Only active when using hybrid functionals, like PBE0 and HSE. Change the TOL_SCF_INIT to change the initial guess for Hartree Fock outer loop.

ACE_FLAG

Type	Unit
Integer	No unit

Default	Example
1	`ACE_FLAG: 0`

Description
Use ACE operator to accelarte the hybrid calculation.

Remark
Without ACE operator, the hybrid calculation will be way slower than with it on depending on the system size.

EXX_METHOD

Type
String

Unit
No unit

Default
`FOURIER_SPACE`

Example
`EXX_METHOD: REAL_SPACE`

Description
Methods to solve Poisson's equation in Exact Exchange part. Options include using FFT to solve it in Fourier space and using linear solver, like CG, to solve in Real space.

Remark

Only active when using hybrid functionals for molecule simulation, like PBE0 and HSE. `FOURIER_SPACE` method is much faster than `REAL_SPACE` method.

Type
Integer

Unit
No unit

Default
0

Example
EXX_MEM: 1

Description
Number of Poisson's equations to be solved in each process at a time when creating exact exchange operator or ACE operator. It could be any non negative integer.

Remark

If set to 0, user could get fastest speed but with highest memory requirement. If set to positive integer, user could run large systems with relatively small memory but slower speed.

EXX_ACE_VALENCE_STATES

Type	Unit
Integer	No unit

Default	Example
3	`EXX_ACE_VALENCE_STATES: 1`

Description
Control of number of unoccupied states used to construct ACE operator.

Remark
Only active when using hybrid functionals with ACE operator.

EXX_DOWNSAMPLING

Type

Integer

Unit

No unit

Default

1 1 1

Example

EXX_DOWNSAMPLING: 1 2 3

Description

Down-sampling of k-points grids. There should be 3 nonnegative integers. 0 means using 0 k-point in that direction, requiring 0 is one of the k-point after time-reversal symmetry in that direction. Positive value should be a factor of the number of grid points in that direction.

EXX_DIVERGENCE

Type
String

Unit
No unit

Default
SPHERICAL

Example
EXX_DIVERGENCE: AUXILIARY

Description
Treatment of divergence in exact exchange. Options are SPHERICAL (spherical truncation), AUXILIARY (auxiliary function method) and ERFC (erfc screening).

Remark

For systems with cube-like geometry, both methods converge fast. For slab and wire, auxiliary function method is a better option. ERFC screening is the default option for HSE in bulk and molecule simulation.

Electrostatics

TOL_POISSON

Type
Integer

Unit
No unit

Default
`TOL_SCF`×0.01

Example
TOL_POISSON: 1e-6

Description
The tolerance on the norm of the relative residual for the Poisson equation.

Remark
The tolerance for poisson solver should not be worse than `TOL_SCF`, otherwise it might seriously affect the convergence of the SCF iteration.

TOL_PSEUDOCHARGE

Type
Double

Unit
No unit

Default
TOL_SCF×0.01

Example
TOL_PSEUDOCHARGE: 1e-6

Description
The normalized error in the net enclosed charge for the pseudocharge density of each atom.

REFERENCE_CUTOFF

Type
Double

Unit
Bohr

Default
0.5

Example
REFERENCE_CUTOFF: 1.0

Description
The cutoff radius of the reference potential used for evaluating the electrostatic correction arising from overlapping pseudocharge densities.

Remark
This number should be smaller than half the smallest interatomic distance.

Stress calculation

CALC_STRESS

Type	Unit
0 or 1	No unit

Default	Example
0	`CALC_STRESS: 1`

Description

Flag for calculation of the Hellmann-Feynman stress tensor (in cartesian coordinates).

CALC_PRES

Type	Unit
0 or 1	No unit

Default	Example
0	`CALC_PRES: 1`

Description

Flag for calculation of the pressure.

Remark

Pressure is directly calculated, without calculation of the stress tensor.

MD

MD_FLAG

Type	Unit
0 or 1	No unit

Default	Example
0	`MD_FLAG: 1`

Description

MD simulations are performed if the flag is set to 1.

Remark

`MD_FLAG` and `RELAX_FLAG` both cannot be set to 1.

MD_METHOD

Type
String

Unit
No unit

Default
NVE

Example
`MD_METHOD: NVE`

Description
Type of MD to be performed.

Remark
Only NVE (microcanonical ensemble) is supported.

Type
Integer

Unit
No unit

Default
0

Example
MD_NSTEP: 100

Description

Specifies the number of MD steps.

Remark

If MD_NSTEP $= N$, the MD runs from 0 to $(N-1)\times$ MD_TIMESTEP fs.

MD_TIMESTEP

Type	Unit
Double	Femtosecond

Default	Example
1	`MD_TIMESTEP: 0.1`

Description

MD time step.

Remark

Total MD time is given by: `MD_TIMESTEP` $\times$ `MD_NSTEP`.

ION_TEMP

Type	Unit
Double	Kelvin

Default	Example
No Default	ION_TEMP: 315

Description

Starting ionic temperature in MD, used to generate initial velocity distribution.

Remark

Must be specified if MD_FLAG is set to 1.

ION_ELEC_EQT

Type
Integer

Unit
No unit

Default
1

Example
ION_ELEC_EQT: 0

Description
Flag that determines whether the `ELEC_TEMP` will be set equal to `ION_TEMP` during MD.

Remark
If the flag is set to 0, the values of `ELEC_TEMP` and `ION_TEMP` need to be identical.

RESTART_FLAG

Type	Unit
0 or 1	No unit

Default	Example
0	`RESTART_FLAG: 0`

Description

Flag for restarting molecular dynamics and structural relaxation.

Remark

Restarts from the previous configuration which is stored in a .restart file.

Structural relaxation

RELAX_FLAG

Type	Unit
0 or 1	No unit

Default	Example
0	`RELAX_FLAG: 1`

Description

Flag for performing structural relaxation. 0 means no structural relaxation. 1 represents relaxation of atom positions. 2 represents optimization of volume with the fractional coordinates of the atoms fixed.

Remark

This flag should not be specified if `MD_FLAG` is set to 1.

RELAX_METHOD

Type
String

Unit
No unit

Default
LBFGS

Example
RELAX_METHOD: NLCG

Description
Specifies the algorithm for structural relaxation. The choices are 'LBFGS' (limited-memory BFGS), 'NLCG' (Non-linear conjugate gradient), and 'FIRE' (Fast inertial relaxation engine).

Remark
LBFGS is typically the best choice.

RELAX_NITER

Type
Integer

Unit
No unit

Default
100

Example
`RELAX_NITER`: 25

Description
Specifies the maximum number of iterations for the structural relaxation (`RELAX_FLAG`).

Remark
If `RESTART_FLAG` is set to 1, then relaxation will restart from the last atomic configuration and run for maximum of `RELAX_NITER` iterations.

TOL_RELAX

Type
Double

Unit
Ha/Bohr

Default
5e-4

Example
TOL_RELAX: 1e-3

Description
Specifies the tolerance for termination of the structural relaxation. The tolerance is defined on the maximum force component (in absolute sense) over all atoms.

TOL_RELAX_CELL

Type	Unit
Double	GPa

Default	Example
1e-2	`TOL_RELAX: 1e-3`

Description

Specifies the tolerance for termination of the cell relaxation. The tolerance is defined on the maximum principle stress component.

RELAX_MAXDIAL

Type	Unit
Double	No unit

Default	Example
1.2	`RELAX_MAXDIAL: 1.4`

Description

The maximum scaling of the volume allowed with respect to the initial volume defined by `CELL` and `LATVEC`. This will determine the upper-bound and lower-bound in the bisection method (Brent's method) for the volume optimization.

NLCG_SIGMA

Type
Double

Unit
No unit

Default
0.5

Example
`NLCG_SIGMA: 1`

Description
Parameter in the secant method used to control the step length in NLCG (`RELAX_METHOD`).

Remark
Default value works well in most cases.

L_HISTORY

Type	Unit
Integer	No unit

Default	Example
20	`L_HISTORY: 15`

Description

Size of history in LBFGS (`RELAX_METHOD`).

Remark

Default value works well in most cases.

L_FINIT_STP

Type	Unit
Double	Bohr

Default	Example
5e-3	`L_FINIT_STP: 0.01`

Description

Step length for line optimizer in LBFGS (`RELAX_METHOD`).

Remark

Default value works well in most cases.

L_MAXMOV

Type
Double

Unit
Bohr

Default
0.2

Example
`L_MAXMOV: 1.0`

Description
The maximum allowed step size in LBFGS (`RELAX_METHOD`).

Remark
Default value works well in most cases.

L_AUTOSCALE

Type
Integer

Unit
No unit

Default
1

Example
`L_AUTOSCALE: 0`

Description
Flag for automatically determining the inverse curvature that is used to determine the direction for next iteration in LBFGS (`RELAX_METHOD`).

Remark
Default works well in most cases.

L_LINEOPT

Type
Integer

Unit
No unit

Default
1

Example
L_LINEOPT: 0

Description
Flag for atomic force based line minimization in LBFGS (`RELAX_METHOD`).

Remark
Required only if `L_AUTOSCALE` is 0.

L_ICURV

Type	Unit
Double	No unit

Default	Example
1.0	`L_ICURV: 0.1`

Description

Initial inverse curvature, used to construct the inverse Hessian matrix in LBFGS (`RELAX_METHOD`).

Remark

Needed only if `L_AUTOSCALE` is 0. Default value works well in most cases.

FIRE_DT

Type	Unit
Double	Femto second

Default	Example
1	`FIRE_DT`: 0.1

Description

Time step used in FIRE (`RELAX_METHOD`).

Remark

Default value works well in most cases.

FIRE_MASS

Type
Double

Unit
Atomic mass unit

Default
1.0

Example
FIRE_MASS: 2.5

Description
Pseudomass used in FIRE (RELAX_METHOD).

Remark
Default value works well in most cases.

FIRE_MAXMOV

Type	Unit
Double	Bohr

Default	Example
0.2	`FIRE_MAXMOV: 1.0`

Description

Maximum movement for any atom in FIRE (`RELAX_METHOD`).

Remark

Default value works well in most cases.

Print options

PRINT_ATOMS

Type	Unit
0 or 1	No unit

Default	Example
0	`PRINT_ATOMS: 1`

Description

Flag for writing the atomic positions. For ground-state calculations, atom positions are printed to a '.static' output file. For structural relaxation calculations, atom positions are printed to a '.geopt' file. For MD calculations, atom positions are printed to a '.aimd' file.
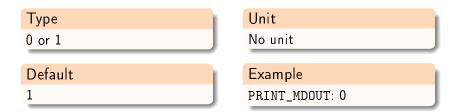
PRINT_FORCES

Type
0 or 1

Unit
No unit

Default
0

Example
PRINT_FORCES: 1

Description
Flag for writing the atomic forces. For ground-state calculations, forces are printed to a '.static' output file. For structural relaxation calculations, forces are printed to a '.geopt' file. For MD calculations, forces are printed to a '.aimd' file.

PRINT_MDOUT

Type	Unit
0 or 1	No unit

Default	Example
1	`PRINT_MDOUT: 0`

Description

Flag for printing the the MD output into the .aimd file.

PRINT_RELAXOUT

Type	Unit
0 or 1	No unit

Default	Example
1	`PRINT_RELAXOUT: 0`

Description
Flag for printing the structural relaxation data in a .geopt file.

Remark
Required only if `RELAX_FLAG` is set to 1.

PRINT_RESTART

Type	Unit
0 or 1	No unit

Default	Example
1	`PRINT_RESTART: 0`

Description

Flag for writing the .restart file, used to restart MD and structural relaxation simulations.

Remark

Relevant only if either `MD_FLAG` is 1 or `RELAX_FLAG` is 1.

PRINT_RESTART_FQ

Type
Integer

Unit
No unit

Default
1

Example
PRINT_RESTART_FQ: 10

Description
Frequency at which .restart file is written in MD and structural optimization simulations.

Remark
Relevant only if either MD_FLAG is 1 or RELAX_FLAG is 1.

PRINT_VELS

Type	Unit
0 or 1	No unit

Default	Example
1	`PRINT_VELS: 0`

Description

Flag for printing the ion velocities in an MD simulation into the .aimd file.

Remark

Relevant only if `MD_FLAG` is set to 1.

OUTPUT_FILE

Type
String

Unit
No unit

Default
Same as the input file name

Example
`OUTPUT_FILE`: myfname

Description
The name of the output files. The output files are attached with a suffix ('.out','.static','.geopt' and '.aimd').

Remark

If an output file with the same name already exist, the results will be written to a file with a number attached, e.g., 'myfname.out_1'. The maximum number of output files with the same name allowed is 100. After that the output files will be overwritten in succession.