3. Usage of programs

3.1. getCluster

Find symmetry-independent clusters (CE / GICE / VLCE / CS-VLCE)

- **Input files**

  POSCAR.cluster, POSCAR.lattice

  gcvc-cluster.in

- **Output files**

  clucar, dimcar, info-cluster, cogcar

> **Note**
>
> 1. The cell size of POSCAR.* should be larger than twice of maximum distance of used clusters due to the periodic boundary condition.

3.2. getCluster-couple

Find independent coupling clusters between base and virtual lattices (VLCE)

- **Input files**

  POSCAR.cluster

  dimcar.conf-couple, dimcar.disp-couple, cogcar.conf-couple, cogcar.disp-couple

  gcvc-cluster-couple.in

- **Output files**

  clucar.couple, dimcar.couple, info-cluster.couple

- **Output**

  Progress status of calculation

> **Note**
>
> 1. dimcar.*-couple and cogcar.*-couple files are obtained by getCluster for conf and disp, and rename output files.
> 2. Coupling of conf-cluster and disp-cluster is not necessarily the same as cluster used for configuration lattice or cluster used for disp lattice.

3.3. getBasis-conf

Calculate basis functions of given structure (CE / GICE / VLCE / CS-VLCE)

- **Input files**

  POSCAR.cluster, POSCAR.spin

  clucar, dimcar

  gcvc-basis-conf.in

- **Output files**

  basecar.conf

- **Output**

  Value of basis functions (Each line for one cluster with multiple basis index)

> **Note**
>
> 1. The 1st line of output is ALWAYS basis function for empty cluster.

3.4. getBasis-disp

Calculate basis functions for displacement on virtual lattice (VLCE / CS-VLCE)

- **Input files**

  POSCAR.cluster, POSCAR.spin, POSCAR.disp

  clucar, dimcar (for virtual lattice)

  gcvc-basis-disp.in

- **Output files**

  basecar.disp

- **Output**

  Value of basis functions

❗ Note

1. Correlation for empty cluster is NOT written in output.
2. Sequence of atomic position in POSCAR.spin and POSCAR.disp should be the same.
3. Number of lines in output should be [(number of lines in dimcar) * (number of calculate direction)]

3.5. getBasis-couple

Calculate basis functions for coupling between base and virtual lattices (VLCE)

- **Input files**

  POSCAR.cluster, POSCAR.spin, POSCAR.disp
  dimcar.conf-couple, dimcar.disp-couple, dimcar.couple
  clucar.conf-couple, clucar.disp-couple, clucar.couple
  gcvc-basis-couple.in

- **Output**

  Value of coupling basis functions

❗ Note

1. Correlation for empty cluster is NOT written in output.

3.6. getGraphLap

Calculate topological invariant based on graph approach

- **Input files**

  POSCAR.cluster, POSCAR.spin
  clucar, dimcar
  gcvc-graph.in

- **Output files**

  basecar.conf
  dosA : DOS for eigenvalues of A and module of A (Ready for plot with gaussian weight).

- **Output**

  Cluster function, trace of power of A (adjacent matrix), graph energy and number of connected graph for A and module of A.

3.7. gs-graph-can

Calculate topological information for all possible configuration

- **Input files**

  POSCAR.cluster
  dimcar, ecicar, mccar.conf
  gcvc-gs-graph-can.in

- **Output files**

  graphcar : Structure index and Topological informations
  indexcar : Structure index information (For debug)
  gscar : Energy & spin variables
  gsunit : POSCAR file for gs-cell. Sequence of spins in gscar is consistent with gsunit.
  poscar.gs : (gs-conf-can only & NUMSTR > 0 only) POSCAR for structures having 1-nth lowest enegy (n = NUMSTR)
  corrcar : (WRITECORR = 1 only) Correlations (excluding empty) for all configurations

- **Output**

  Final energy and topological information (for GS and check. They should be the same value) for A and module A

❗ Note

1. Format of graphcar: [str index, (Num connected links, graph energy, Tr[module{A}^2], Tr[module{A}^3) for each module]
2. Since graph energy and number of connected graphs are non-extensive, their values for original cell
   (i.e., values before dividing cell, corresponding to number of atoms in POSCAR.cluster) are output.
3. Unit of ECI in ecicar should be consistent with the system size in POSCAR.cluster
   (e.g.: If POSCAR.cluster contains 256 atoms, ECI should be in [eV / 256-atom])

4. Unit of energy in gscar is [eV / (numAtoms in POSCAR.cluster / (gscellsize-x * gscellsize-y * gscellsize-z))]

3.8. randMat

Calculate DOS for Random Matrix

- **Input files**
  NO file is needed. Instead, 3 arguments are required.

- **Arguments[0-2]**
  [0]: Number of basis functions (i.e., Dimension of configuration space)
  [1]: Number of sampling point
  [2]: Maximum order of moment for DOS

- **Output**
  Normalized covariance matrix, Eigenvalues, Moment of covariance matrix.

3.9. conf-space

Analyzing statistical information for configuration space

- **Input files**
  gcvc-conf-space.in
  data file (Each line: sampling data, Each row: coordination for configuration (or some other) space)

- **Argument[0]**
  [0]: Filename of data file (Note: double quotation is NOT needed)

- **Output**
  Normalized covariance matrix, Eigenvalues, Moment of covariance matrix.

3.10. opt-basis

Find optimal set of clusters giving lowest cross-validation score

CV score is optimized using simulated annealing with MC simulation.

- **Input files**
  CORRELATION, enecar
  gcvc-opt-basis.in

- **Output**
  Progress of simulation (Temperature, CV)
  Optimal cluster indices and CV score

Note

1. When applying optimized set of clusters to lsfit and obtain unexpected magnitude of ECIs, correlations for pure elements in CORRELATION file are omitted (but at least correlations for ONE pure element should be included in CORRELATION file!), and then retry opt-basis.

3.11. lsfit

Main program to obtain effective interactions by least-square fitting

- **Input files**
  enecar, CORRELATION
  gcvc-lsfit.in

- **Output files**
  ecicar : cluster index and ECI (for multiple basis index)

- **Output**
  DFT and CE-predicted energy, ECI, Cross validation, LS-fitted error, and Determinant of covariance matrix

Note

1. Unit of ecicar is the same as enecar.

3.12. setMC-conf

Construct coordination information for each lattice points

- **Input files**

clucar, dimcar

- **Output files**

  mccar.conf

Note

1. clucar.conf and dimcar.conf are obtained by getCluster, and rename output files.

3.13. str-select, str-select-can

Find optimal set of DFT input structures based on MWV (Maximum Weigted Validation)

str-select finds structures for WHOLE compositions
str-select-can finds structures for SINGLE composition

- **Input files**

  POSCAR.cluster
  dimcar, ecicar, mccar.conf
  CORRELATION
  gcvc-ss.in (for str-select), gcvc-ss-can.in (for str-select-can)
  see gcvc-ss.in, gcvc-ss-can.in

- **Output files**

  poscar.mwv : POSCAR files for selected structures.
  mwvunit : POSCAR file for mwv-cell. Sequence of spins is consistent with SPINCE.

- **Output**

  Temperature and maximum MWV

Note

1. Larger MWV value means better set of DFT input structures, so try to obtain larger MWV by changing calculation condition. In order to obtain maximum value of MWV, simulated annealing algorism is employed.
2. Use ecicar optimized by opt-basis, where ECI should be [eV / (number of atoms in POSCAR.cluster)].
3. CORRELATION contains correlations (including selected and not-selected) only for structures whose energies are used to obtain ecicar.
4. If the optimized ECI (obtained by opt-basis or etc.) has for instance, (ternary case)
   1-NN pair (1,1) (2,2) (but basis (1,2) is NOT included in optimized ECI),
   then you should write ecicar as follows:
   [line for the above pair (here, cluster index is 2 for instance).]:
   2 -0.1 0.0 0.8
   This means that basis index (1,2) which is NOT included in optimized ECI should have “zero” effective interaction, which should be explicitly written in ecicar. This description should be required to tell the program linear-dependence of basis functions.
5. Correlations whose variance is zero are automatically omitted in this program to estimate MWV.

3.14. cmc-graph

Main program for getting topological information with configurational MC simulation (canonical)

- **Input files**

  POSCAR.cluster, POSCAR.spin
  mccar.conf, dimcar, ecicar
  gcvc-cmc-graph.in
  see gcvc-cmc-graph.in

- **Output files**

  graph-cmc : Main results of graph info
  (Tabbed by each figure: 1st-row: Number of connected graph, 2nd: Graph Energy, 3,4-row: trace of AA and AAA)
  A: Adjacency matrix for defined basis functions
  info-cmc : Main results of MC (Energy, basis func., etc)
  poscar.cmc : Atomic position for final MC step
  corr.cmc : Basis functions for each MC-step at annealing temperature (1st row is always 1.0 (=corr of empty))
  energy.cmc : Energy and squared energy for each MC-step at annealing temperature
  Unit of energy is [eV / atoms in MC-cell] for cmc-conf, and [eV / atoms in MC-divide-cell] for cmc-conf-div
  t-energy.ave, t-corr.ave (cmc-conf only) : Averaged energy and correlations

at each temperature

- **Output**

  Temperature and MC step

> **Note**
>
> 1. Unit of ECI in ecicar should be consistent with the system size in POSCAR.spin
>   (e.g.: If POSCAR.spin contains 256 atoms, ECI should be in [eV / 256-atom])
> 2. Unit of energy in info-cmc is :
>   [eV / numAtoms in POSCAR.cluster] for cmc-conf
>   [eV / numAtoms in MC divided-cell] for cmc-conf-div

3.15. wang-landau

Main program for Wang-Landau simulation

- **Input files**

  POSCAR.cluster, POSCAR.spin
  mccar.conf, dimcar, ecicar
  gcvc-wl.in
  see gcvc-wl.in

- **Output files**

  info-wl : Simulation results of Wang-Landau
  poscar.wl : Atomic position for final MC step
  ln-g.all : Log of density of states, log(g(E)) for all iteration.
  ln-g.wl : log(g(E)) for final iteration, normalized to sum[g(E)] = R^N.
  macro-state.wl : POSCARs for macrostates.

- **Output**

  F value and WL step

3.16. f-wang-landau

Main program for calculating free energy from log(g(E)) obtained by WL simulation

- **Input files**

  ln-g.wl
  gcvc-f-wl.in
  see gcvc-f-wl.in

- **Output**

  Temperature and Helmholtz free energy

3.17. wang-landau-2d

Main program forWang-Landau simulation to construct two-dimensional configurational DOS

- **Input files**

  POSCAR.cluster, POSCAR.spin
  mccar.conf, dimcar, ecicar
  gcvc-wl-2d.in
  see gcvc-wl-2d.in

- **Output files**

  info-wl : Simulation results of Wang-Landau
  poscar.wl : Atomic position for final MC step
  ln-g.all : Log of density of states, log(g(x,y)) for all iteration.
  ln-g.wl : log(g(x,y)) for final iteration, normalized to sum[g(x,y)] = R^N.

> **Note**
>
> 1. Cluster indices used for wang-landau-2d are specified in ecicar file. The ecicar file should contain only empty, and user-selected TWO clusters. You can choose point cluster as one of the selected two clusters.
> 2. Unlike wang-landau program, the wang-landau-2d provides two-dimensional configurational DOS.
> 3. Value of ECIs in ecicar file are NOT used in wang-landau-2d: Resultant two-dimensional CDOS is estimated without any weight of energy.

3.18. cmc-conf, cmc-conf-div

Main program for configurational MC simulation (canonical)

cmc-conf is MC simulation for MC-cell size same as size of POSCAR.spin
cmc-conf-div is MC simulation for MC-cell size by dividing cell of POSCAR.spin
(i.e., MC simulation for smaller cell size)

- **Input files**

  POSCAR.cluster, POSCAR.spin
  mccar.conf, dimcar, ecicar
  gcvc-cmc-conf.in (for cmc-conf), gcvc-cmc-conf-div.in (for cmc-conf-div)
  see gcvc-cmc-conf.in, gcvc-cmc-conf-div.in

- **Output files**

  info-cmc : Main results of MC (Energy, basis func., etc)
  poscar.cmc : Atomic position for final MC step
  corr.cmc : Basis functions for each MC-step at annealing temperature (1st row is always 1.0 (=corr of empty))
  energy.cmc : Energy and squared energy for each MC-step at annealing temperature
  Unit of energy is [eV / atoms in MC-cell] for cmc-conf, and [eV / atoms in MC-divide-cell] for cmc-conf-div
  t-energy.ave, t-corr.ave (cmc-conf only) : Averaged energy and correlations at each temperature

- **Output**

  Temperature and MC step

Note

1. Unit of ECI in ecicar should be consistent with the system size in POSCAR.spin
   (e.g.: If POSCAR.spin contains 256 atoms, ECI should be in [eV / 256-atom])
2. Unit of energy in info-cmc is :
   [eV / numAtoms in POSCAR.cluster] for cmc-conf
   [eV / numAtoms in MC divided-cell] for cmc-conf-div
3. For cmc-conf-div, POSCAR.spin should be created so that each resultant MC-divided-cells specified by “MCCELLSIZE” should have the same atomic configuration. (Otherwise, cmc-conf-div program show errors and stops automatically)

3.19. cmc-conf-fix

Special canonical MC simulation under fixed-lattice-points condition (Advanced)

MC simulation where occupation of specified lattice points are kept fixed to initial configuration
This program is used for special calculation in SRO-disorder expansion theory.

- **Input files**

  POSCAR.cluster, POSCAR.spin
  mccar.conf, dimcar, ecicar
  gcvc-cmc-conf-fix.in

- **Output files**

  info-cmc : Main results of MC (Energy, basis func., etc)
  poscar.cmc : Atomic position for final MC step
  corr.cmc : Basis functions for each MC-step at annealing temperature

- **Output**

  Temperature and MC step

3.20. cmc-conf-pmb

Special canonical MC simulation: Ensemble average separated by specified cluster with correlation below/above specified value

This program is used for special calculation in SRO-disorder expansion theory.

- **Input files**

  POSCAR.cluster, POSCAR.spin
  mccar.conf, dimcar, ecicar
  gcvc-cmc-conf-pmb.in

- **Output files**

  info-cmc : Main results of MC (Averaged correlation with specified cluster with correlation below/above specified value)
  poscar.cmc : Atomic position for final MC step

- **Output**

  Temperature and MC step

3.21. gcmc-conf

Main program for semi-grand canonical MC simulation (canonical)

- **Input files**

  POSCAR.cluster, POSCAR.spin
  mccar.conf, dimcar, ecicar
  gcvc-gcmc-conf.in

- **Output files**

  info-cmc : Main results of MC (Energy, basis func., etc)
  poscar.cmc : Atomic position for final MC step
  corr.cmc : Basis functions for each MC-step at annealing temperature (1st row is always 1.0 (=corr of empty))
  energy.cmc : Energy and squared energy for each MC-step at annealing temperature
  Unit of energy is [eV / atoms in MC-cell] for cmc-conf, and [eV / atoms in MC-divide-cell] for cmc-conf-div
  t-energy.ave, t-corr.ave (cmc-conf only) : Averaged energy and correlations at each temperature

- **Output**

  Temperature and MC step

Note

1. Unit of ECI in ecicar should be consistent with the system size in POSCAR.spin
   (e.g.: If POSCAR.spin contains 256 atoms, ECI should be in [eV / 256-atom])
2. Unit of energy in info-cmc is :
   [eV / numAtoms in POSCAR.cluster] for cmc-conf
   [eV / numAtoms in MC divided-cell] for cmc-conf-div
3. For cmc-conf-div, POSCAR.spin should be created so that each resultant MC-divided-cells specified by “MCCELLSIZE” should have the same atomic configuration. (Otherwise, cmc-conf-div program show errors and stops automatically)

3.22. stat-cluster

Cluster statistical information from MC simulation data

- **Input files**

  corr.cmc
  gcvc-stat-cluster.in

- **Output files**

  prob-dist : (SIGMA >= 0.0 only) Probability distribution of basis functions for selected clusters. (1-row: basis func. 2-row: distribution)
  bd-dist : (SIGMA2 >= 0.0 only) Bivariate probability distribution of basis function for selected two clusters.
  (1- and 2-row: basis func. for two clusters 3-row: distribution)

- **Output**

  (STATFLAG = 1 only) Cluster statistical information (Average, variance, covariance, correlation coefficient)

3.23. find-r

Find optimal R value and calculate corresponding ensemble-averaged pair correlation

This program is used for lattice statistical informatics theory

- **Input files**

  gcvc-find-r.in

- **Output**

  Set of [Trial R, ave-corr(pmb), ave-e(pmb)], optimal R, and set of [Temperature, ensemble-averaged pair correlation, averaged energy], and [T, averaged-property]

3.24. gs-conf, gs-conf-can

Performing ground-state search

gs-conf performs ground-state search for WHOLE compositions
gs-conf-can performs ground-state search for SINGLE composition

- **Input files**

  POSCAR.cluster
  dimcar, ecicar, mccar.conf
  gcvc-gs-conf.in (for gs-conf), gcvc-gs-conf-can.in (for gs-conf-can)
  see gcvc-gs-conf.in, gcvc-gs-conf-can.in

- **Output files**

gscar : Energy & spin variables
gsunit : POSCAR file for gs-cell. Sequence of spins in gscar is consistent with gsunit.
poscar.gs : (gs-conf-can only & NUMSTR > 0 only) POSCAR for structures having 1-nth lowest enegy (n = NUMSTR)
corrcar : (WRITECORR = 1 only) Correlations (excluding empty) for all configurations

- **Output**

  Final energy (for GS and check. They should be the same value)

Note

1. Unit of ECI in ecicar should be consistent with the system size in POSCAR.cluster
(e.g.: If POSCAR.cluster contains 256 atoms, ECI should be in [eV / 256-atom])
2. Unit of energy in gscar is [eV / (numAtoms in POSCAR.cluster / (gscellsize-x * gscellsize-y * gscellsize-z))]

3.25. equiv-str

Search for all symmetry-equivalent structures to POSCAR.spin

- **Input files**

  POSCAR.cluster, POSCAR.spin
  dimcar, mccar.conf
  gcvc-es.in

- **Output files**

  escar : Spin variables (for equivalent structures). Sequence of spin is consistent with POSCAR.cluster.

- **Output**

  Correlation for POSCAR.spin, for final configuration (obtained by es-simulation), and for final configuration (by solely calculation of correlation with spinarray)

Note

1. In order to raise accuracy for searching equivalent structures, dimcar.conf (and generated mccar.conf by setMC code) should contain as many cluster type as possible. Therefore, when performing getCluster code, “TRUNCATION” should be as large as possible to get clucar and dimcar.

3.26. qr-search, qrm-search, qrma-search, qrl-search

Search for quasi-random (or any structure with given correlations) structure(s)

qr-search is a regular and fastest program. You can use this program except for special reason. Enables calculation of LP-norm.
qrm-search is a special program to search multiple structures whose average of correlations are as close to as those in qrccar file.
qrma-search searches set of structures whose correlation “ratio” becomes closer to specified value.
qrl-search searches single structure, where composition of specified area(s) (in POSCAR.canarea) is kept fixed at initial value.

- **Input files**

  POSCAR.cluster
  POSCAR.spin (qrm-search: READPOS = 1 & qrl-search only)
  POSCAR.canarea (qrl-search only)
  dimcar, mccar.conf, qrccar
  gcvc-qr.in (qr-search)
  gcvc-qrm.in (qrm-search)
  gcvc-qra.in (qrma-search)
  gcvc-qrl.in (qrl-search)

- **Output files**

  info-qr: (qr-search) Minimum distance and correlation for selected struture.
  info-qrm or info-qrl: Minimum distance and correlation for selected struture(s), and average correlation of selected structures (for multiple search)
  poscar.qr or poscar.qrm : POSCAR for selected structure(s)

- **Output**

  Tempearture, minimum-distance in the current status
  Multiplied factor, R (qrma-search only)

Note

1. Algorism is based on finding structure(s) so that correlation distance between selected structure(s) and those in qrccar become minimum. In order to obtain minimum distance, simulated annealing algorism is employed.
2. When multiple structure search (NUMSTR > 1), qrm-search finds best "multiple" sturctures whose "average" correlations are close to those written in qrccar.
3. If you want to search quasi-random structure, write correlations of clusters for ideally-random structure in qrccar file.
4. Format of qrccar file can be obtained from part of for instance, "info-cmc" file:

   ```
   [cluster index] [correlations (with multiple basis index)]
   [cluster index] [correlations (with multiple basis index)]
   ...
   ```

   which MUST INCLUDE empty (index=0). Correlation of empty should be equal to 1.0. This file tells program the correlation of random structure to be searched.
5. Composition for qrccar and "COMPOSITION" in gcvc-qr.in, gcvc-qra.in or gcvc-qrl.in should be consistent.
6. (qrma-search only) DO NOT write correlation of point cluster in qrccar file.
7. (qrl-search only) POSCAR.canarea specifies sublattice-like index, where composition within set of lattice points
   with the same sublattice-like index is kept fixed at initial value during the simulation.
   Therefore, POSCAR.canarea can differ from POSCAR.lattice.

3.27. convert-axis

Convert POSCAR with original axis to user-defined axis

- **Input files**

  POSCAR
  gcvc-ca.in

- **Output**

  POSCAR with user-defined axis written in gcvc-ca.in

Note

1. Triplet product of axis vectors in gcvc-ca.in should be positive. (This is convensional for crystallography with right-hand rule).

3.28. trans-pos

Perform translation and/or magnification of internal positions in POSCAR

- **Input files**

  POSCAR
  gcvc-tp.in

- **Output**

  POSCAR after translation and magnification.

3.29. sortPoscarDisp

Sort sequence of internal position in POSCAR.disp same as POSCAR.spin

- **Input files**

  POSCAR.disp POSCAR.spin

- **Output**

  Sorted poscar, which can be used for new POSCAR.disp