disorder : a code for generating irreducible site-occupancy configurations

— interfaced to VASP

***disorder* manual**

What is *disorder* ?

disorder is an open source software designed for generating irreducible site-occupancy configurations (i.e., symmetrically inequivalent disordered crystal structures), which can be used for disordered doping, including substitution doping and vacancy doping. The ***disorder*** code works for arbitrary parent cells with any supercell expansion matrix, and for any number of atomic types with arbitrary stoichiometry. Most important, a linear scale of run time with the number of irreducible configurations is achieved, which is the best possible scaling for this type of problem.

Features

- Build supercell form arbitrary input cell with any supercell expansion matrix;
- Search space group operations of the supercell and identify its point group symbol;
- Construct the equivalent atomic matrix of the doping site;
- Generate irreducible configurations for any number of atomic types with arbitrary stoichiometry;
- Count the degeneracy (the number of equivalent configurations) of each irreducible configuration;
- Output the crystal structure file of each irreducible configuration.

Author & Contact

Ji-Chun Lian (jichunlian@hnu.edu.cn), Department of Applied Physics, School of Physics and Electronics, Hunan University, Changsha, 410082, China

If you have any questions, suggestions, and problems regarding ***disorder***, please feel free to contact me.

How to Cite ?

Please cite the following article when you use ***disorder***:

J.-C. Lian, H.-Y. Wu, W.-Q. Huang, W. Hu, and G.-F. Huang, Phys. Rev. B **102**, 134209 (2020).

If you want to understand the algorithm implemented in ***disorder***, you can read this reference carefully. After several updates, as an aside, the latest version of ***disorder*** possesses better time and space complexity than the original algorithm described in this reference, but the key concept is consistent.

Download

The latest version of ***disorder*** package interfaced to **VASP** is available on GitHub: https://github.com/jichunlian/disorder, and that interfaced to **PWmat** can be download from: http://www.pwmat.com/module-download.

Installation

$ unzip disorder-master

$ cd disorder-master

$ make disorder (Only ***disorder*** is compilied)

$ make supercell (Only ***supercell*** is compilied)

$ make or make all (Both ***disorder*** and ***supercell*** are compilied)

$ make clean (rm *.o *.mod disorder supercell)

export PATH=$PATH:disorder_installation_path/disorder-master/bin

(Add this to the ~/.bashrc file as you wish)

Note: Only the **ifort** compiler is supported.

Input files

SPOSCAR (Crystal structure file)

```
SPOSCAR
   1.0
    13.1323995589999996    0.0000000000000000    0.0000000000000000
     0.0000000000000000   13.1323995589999996    0.0000000000000000
     0.0000000000000000    0.0000000000000000   13.1323995589999996
  Te    Pb
  32    32
Direct
  0.0000000000000000  0.0000000000000000  0.0000000000000000
  0.5000000000000000  0.0000000000000000  0.0000000000000000
  0.0000000000000000  0.5000000000000000  0.0000000000000000
  0.5000000000000000  0.5000000000000000  0.0000000000000000
  0.0000000000000000  0.0000000000000000  0.5000000000000000
  0.5000000000000000  0.0000000000000000  0.5000000000000000
  0.0000000000000000  0.5000000000000000  0.5000000000000000
  0.5000000000000000  0.5000000000000000  0.5000000000000000
  0.0000000000000000  0.2500000000000000  0.2500000000000000
  0.5000000000000000  0.2500000000000000  0.2500000000000000
  0.0000000000000000  0.7500000000000000  0.2500000000000000
  0.5000000000000000  0.7500000000000000  0.2500000000000000
  0.0000000000000000  0.2500000000000000  0.7500000000000000
  0.5000000000000000  0.2500000000000000  0.7500000000000000
  0.0000000000000000  0.7500000000000000  0.7500000000000000
  0.5000000000000000  0.7500000000000000  0.7500000000000000
  0.2500000000000000  0.0000000000000000  0.2500000000000000
```

⋮

The **SPOSCAR** is the crystal structure file after cell expansion, its format is consistent with the **POSCAR** file of **VASP**. The ***disorder*** does not expanding the cell inputted from **SPOSCAR**, but we have provided an util (i.e., ***supercell***) for cell expansion, in which the non-diagonal supercell expansion matrix is also supported. Any other software can be also used for cell expansion, as long as the format of **SPOSCAR** file is correct.

INDSOD (Running control file)

INDSOD file contains **11** parameters to control the running of ***disorder***, its format and the detailed explanation of all parameters are shown in below and next few pages.

```
&input
  nsub = integer
  subs = integer, integer, …
  symb = character, character, …
  site = integer
  prec = real
  fast = logical
  lpro = logical
  lpos = logical
  leqa = logical
  lspg = logical
  lcfg = logical
/
```

```
&input
  nsub = 2
  subs = 2,70
  symb = Kw,O
  site = 2
  prec = 1D-5
! fast = .true.
  lpro = .true.
! lpos = .true.
! leqa = .true.
! lspg = .true.
! lcfg = .false.
/
```

Note:

The head "&input" and tail "/" are indispensable and immutable.

The orders of the above parameters are irrelevant.

The parameters with default values can be omitted.

The exclamation mark "!" can be used for annotations.

The annotated parameter is equivalent to using the default value.

All parameters are not case sensitive.

nsub-tag

nsub = 2 ~ 5; Default: nsub = 2

The multi-nary (binary, ternary, quaternary, and quinary) site-occupancy system is also supported in ***disorder***, i.e., nsub different types of atoms are allowed to occupy the atomic positions of the doping site. In practice, quinary system is quite enough, but you can modify the code (lines 35, 36, and 43 in **disorder.f90** file) to support a higher nsub.

subs-tag

subs = integer, integer, …; Default: None

The subs-tag is an integer array with a size of nsub. The elements in subs represent the numbers of doping atoms. The sum of subs needs to be equal to the number of atomic positions of the doping site. We recommend always putting the largest integer value last.

symb-tag

symb = character, character, …; Default: None

The symb-tag is an character string array with a size of nsub. The elements in symb represent the symbols of doping atoms, and corresponding to the elements in subs one by one. The vacancies are deemed as a special type of atoms. In ***disorder***, we define the symbol of vacancy as "**Kw**" (note that "**K**" is upper case and "**w**" is lower case), i.e., the first letter of Chinese Pinyin of vacancy (**K**ong **w**ei).

site-tag

site = integer; Default: 1

The site-tag indicates which type of atoms is selected as the doping site, while the atomic types are numbered in sequence according to their position in the **SPOSCAR** file.

prec-tag

prec = real; Default: 1D-5

The prec-tag determines how accurate the positions in the **SPOSCAR** file must be. The default is 10^{-5}, which is usually suffiently large even if the **SPOSCAR** file has been generated with a single precision program. Increasing prec means that the positions in the **SPOSCAR** file can be less accurate. The prec-tag plays an important role in searching the space group operations. The correctness of space group operations is completely determined by the **SPOSCAR** file and prec-tag, which are both controlled by user. Therefore, ***disorder*** does not guarantee the correctness of the space group operations.

More on prec-tag

On account of the "noise" in the structural parameters (lattice vectors and/or ionic positions), the determination of the symmetry of some structures will shows a strong dependence on the tolerance parameter **prec**. Therefore, we suggest that the point group symbol obtained by ***disorder*** should be compared with that obtained by other softwares to ensure the reliability of the results (note that the point group obtained by ***disorder*** is that of the supercell not the primitive cell). When the point group symbols obtained by ***disorder*** and other softwares are inconsistent, you can try to adjust the prec-tag or the structure file **SPOSCAR**, otherwise you may get the wrong result. Fortunately, for most structures, the default is fine.

fast-tag

fast = logical; Default: .false. (semi-automatic)

The fast-tag controls whether the *fast mode* is enabled in the process of eliminating duplicate configurations. The default value of fast-tag is .false., that is, the *fast mode* is not enabled (i.e., *standard mode* is enabled), but when certain conditions are met, the fast-tag will be automatically adjusted to .true.. In addition, even if fast = .true. is set, in some cases, the fast-tag will be automatically adjusted to .false.. In other words, the fast-tag is a semi-automatic parameter, and not completely controlled by the user. It will take a long time to fully explain the *fast mode*. But don't worry, we suggest that always set fast = .false., when you need to obtain the degeneracies of the irreducible configurations, and you can set fast = .true., when the degeneracies are not needed.

lpro-tag

lpro = logical; Default: .false.

The lpro-tag controls whether displays the progress bar in the process of eliminating duplicate configurations. For the small systems, the running time may very short, so the display of the progress bar has no practical significance. For the large systems, the running time is long enough, and the display of the progress bar can track the running progress of the program, but it will slightly slow down the speed of the program.

lpos-tag

lpos = logical; Default: .false.

The lpos-tag controls whether output the structure files **POSCAR-x…** of the irreducible configurations. About **POSCAR-x…** files, see the next section for details.

leqa-tag

leqa = logical; Default: .false.

The leqa-tag controls whether output the equivalent atomic matrix file **EQAMAT**. About **EQAMAT** file, see the next section for details.

lspg-tag

lspg = logical; Default: .false.

The lspg-tag controls whether output the space group operations file **SPGMAT**. About **SPGMAT** file, see the next section for details.

lcfg-tag

lcfg = logical; Default: .true.

The lpos-tag controls whether output the irreducible configurations file **CFGMAT**. About **CFGMAT** file, see the next section for details.

Output files

SPGMAT (Space group operations file)

```
 8 = 4 x 2

-1   0   0    0.000000    0.500000
 0  -1   0    0.000000    0.000000
 0   0  -1    0.000000    0.000000

 1   0   0    0.000000    0.500000
 0   0   1    0.000000    0.000000
 0  -1   0    0.000000    0.000000

 1   0   0    0.000000    0.500000
 0   0   1    0.000000    0.000000
 0   1   0    0.000000    0.000000

 1   0   0    0.000000    0.500000
 0   1   0    0.000000    0.000000
 0   0  -1    0.000000    0.000000

 1   0   0    0.000000    0.500000
 0   1   0    0.000000    0.000000
 0   0   1    0.000000    0.000000
```

The **SPGMAT** file stores the space group operations information of the supercell. It is not output by default. For most users, this file is useless. The **SPGMAT** file contains N_r 3 × 3 integer matrices (in the fractional coordinates), representing rotation operations. Each matrix is followed by N_t 3 × 1 decimal column vector (in the fractional coordinates), which represent translation operations. In addition, the first line of **SPGMAT** file is $N_o = N_r$ x N_t, where N_o is the number of space group operations.

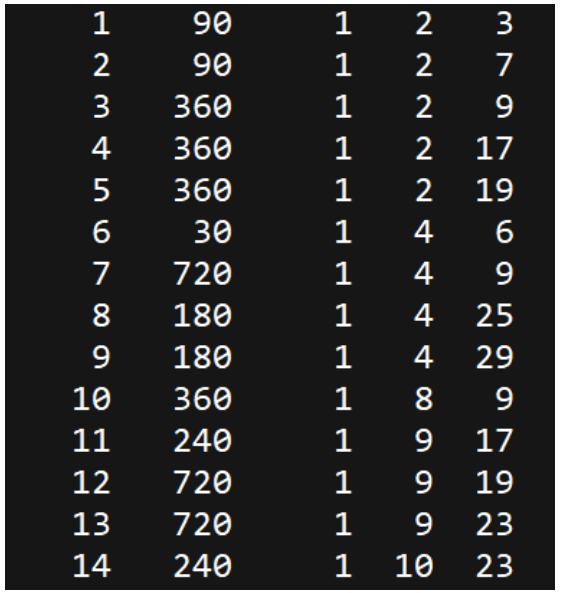
EQAMAT (Equivalent atomic matrix file)

The **EQAMAT** file stores the equivalent atomic matrix information of the doping site. It is not output by default. For most users, this file is useless. A set of atomic points labeled with consecutive integers are transformed into another set of atomic points by a space group operation. The labels of the transformed atomic points constitute one row elements of the equivalent atomic matrix, while the complete equivalent atomic matrix can be obtained by traversing all space group operations. Therefore, the number of rows corresponds to the number of space group operations, and the number of columns corresponds to the number of atomic positions of the doping site.

5	6	7	8	1	2	3	4	11	12	9	10
6	5	8	7	2	1	4	3	12	11	10	9
7	8	5	6	3	4	1	2	9	10	11	12
8	7	6	5	4	3	2	1	10	9	12	11
1	2	3	4	5	6	7	8	11	12	9	10
2	1	4	3	6	5	8	7	12	11	10	9
3	4	1	2	7	8	5	6	9	10	11	12
4	3	2	1	8	7	6	5	10	9	12	11
5	6	7	8	1	2	3	4	9	10	11	12
6	5	8	7	2	1	4	3	10	9	12	11
7	8	5	6	3	4	1	2	11	12	9	10
8	7	6	5	4	3	2	1	12	11	10	9
1	2	3	4	5	6	7	8	9	10	11	12
2	1	4	3	6	5	8	7	10	9	12	11
3	4	1	2	7	8	5	6	11	12	9	10
4	3	2	1	8	7	6	5	12	11	10	9

CFGMAT (Irreducible configurations file)

The **CFGMAT** file stores the information of the irreducible configurations and its degeneracies. It is output by default. The first column of the **CFGMAT** file is the serial number of each irreducible configuration, which has no practical significance. The second column is the degeneracy of each irreducible configuration, which represents the number of equivalent configurations (including itself), and the sum of the degeneracies should be equal to the total number of configurations. All the following columns are the position labels of the irreducible configurations.



1	90	1	2	3
2	90	1	2	7
3	360	1	2	9
4	360	1	2	17
5	360	1	2	19
6	30	1	4	6
7	720	1	4	9
8	180	1	4	25
9	180	1	4	29
10	360	1	8	9
11	240	1	9	17
12	720	1	9	19
13	720	1	9	23
14	240	1	10	23

POSCAR-x… (Crystal structure files)

The **POSCAR-x…** files, under the **poscar** folder, are the crystal structure files of all irreducible configurations, and its format is also consistent with the **POSCAR** file of **VASP**. The file name suffix "**-x…**" denotes the serial number, which corresponds to the irreducible configuration with the same serial number in **CFGMAT**.

```
hnu_ljc:1_SnxPb1-xTe$ ls poscar/
POSCAR-01  POSCAR-09  POSCAR-17  POSCAR-25  POSCAR-33  POSCAR-41  POSCAR-49  POSCAR-57
POSCAR-02  POSCAR-10  POSCAR-18  POSCAR-26  POSCAR-34  POSCAR-42  POSCAR-50  POSCAR-58
POSCAR-03  POSCAR-11  POSCAR-19  POSCAR-27  POSCAR-35  POSCAR-43  POSCAR-51  POSCAR-59
POSCAR-04  POSCAR-12  POSCAR-20  POSCAR-28  POSCAR-36  POSCAR-44  POSCAR-52  POSCAR-60
POSCAR-05  POSCAR-13  POSCAR-21  POSCAR-29  POSCAR-37  POSCAR-45  POSCAR-53  POSCAR-61
POSCAR-06  POSCAR-14  POSCAR-22  POSCAR-30  POSCAR-38  POSCAR-46  POSCAR-54  POSCAR-62
POSCAR-07  POSCAR-15  POSCAR-23  POSCAR-31  POSCAR-39  POSCAR-47  POSCAR-55  POSCAR-63
POSCAR-08  POSCAR-16  POSCAR-24  POSCAR-32  POSCAR-40  POSCAR-48  POSCAR-56  POSCAR-64
```

Running

We have provided three examples (i.e., 1_SnxPb1-xTe, 2_TiO2-VO and 3_Tellurene) for testing. Here, we will use the first example to introduce the running of ***disorder***, and the second example will be used to introduce the running of ***supercell***.

```
hnu_ljc:disorder-master$ ls examples/
1_SnxPb1-xTe  2_TiO2-VO  3_Tellurene
```

disorder

After the two input files **INDSOD** and **SPOSCAR** are ready, the running of ***disorder*** is quite simple, as shown in the right.

Note: If the executable binary file of ***disorder*** is under the environment variable path, just type “disorder” on the command line without including its path.

supercell

The ***supercell*** is an util used to expanding the small cell (it can be any cell, such as primitive cell, unit cell, and supercell). The input file of ***supercell*** is **POSCAR** (the structure file before cell expansion), and the output file is **SPOSCAR** (the structure file after cell expansion). Moreover, the supercell expansion matrix $\boldsymbol{M}$ (3 × 3 integer matrix) is inputed from the keyboard, of course, the input redirection (<) and pipe (|) can also be used to input $\boldsymbol{M}$.

The supercell expansion matrix $\boldsymbol{M}$ is defined by $\boldsymbol{A_s = A_i M^T}$ (be careful the transposition of $\boldsymbol{M}$), where $\boldsymbol{A_i} = (\boldsymbol{a_i}\ \boldsymbol{b_i}\ \boldsymbol{c_i})$ is the lattice basis vectors before cell expansion, and $\boldsymbol{A_s} = (\boldsymbol{a_s}\ \boldsymbol{b_s}\ \boldsymbol{c_s})$ is the lattice basis vectors after cell expansion. Be careful that the axes in **POSCAR** is defined by three row vectors, i.e., $(\boldsymbol{a_i}\ \boldsymbol{b_i}\ \boldsymbol{c_i})^T$.

Note: Because the coordinates is defined in the right-handed system, the determinant of $\boldsymbol{M}$ should be greater than 0.

If the small cell is an unit cell

```
hnu_ljc:2_TiO2-VO$ cp POSCAR_unit POSCAR
hnu_ljc:2_TiO2-VO$ ../../bin/supercell
3 0 0
0 3 0
0 0 1
```

If the small cell is a primitive cell

```
hnu_ljc:2_TiO2-VO$ cp POSCAR_prim POSCAR
hnu_ljc:2_TiO2-VO$ ../../bin/supercell
3 0 3
0 3 3
-1 -1 0
```

Use pipe (|) to input ***M***

```
hnu_ljc:2_TiO2-VO$ echo -e "3 0 3\n0 3 3\n -1 -1 0"|../../bin/supercell
hnu_ljc:2_TiO2-VO$ printf "3 0 3\n0 3 3\n -1 -1 0"|../../bin/supercell
```

Use input redirection (<) to input ***M***

```
hnu_ljc:2_TiO2-VO$ ../../bin/supercell < M
hnu_ljc:2_TiO2-VO$ cat M
  3  0  3
  0  3  3
 -1 -1  0
```

All of these commands get the same supercell.

Disclaimers

The ***disorder*** code does not guarantee the correctness of the results. We will not be responsible for any errors caused by any reason and any adverse effects on users or others.

The End !