**Manual for domain walls(DWs) structures construction using Python scripts**

**Contents**

Cartesian\_direct.py : Convert Cartesian coordinate system to direct coordinate system

Direct\_cartesian.py : Convert direct coordinate system to Cartesian coordinate system

O\_90.py : construct 90° domain wall structure of orthorhombic phase

O\_180\_001.py : construct 180° domain wall structure of orthorhombic phase along 001 direction

O\_180\_100.py : construct 180° domain wall structure of orthorhombic phase along 100 direction

R\_71.py : construct 71° domain wall structure of rhombohedral phase

R\_109.py : construct 109° domain wall structure of rhombohedral phase

R\_180\_A\_Ising.py : construct 109° domain wall structure of rhombohedral phase

T\_90.py : construct 90° domain wall structure of tetragonal phase

T\_180\_A\_ising.py : construct A centered Ising type 180° domain wall structure of tetragonal phase

T\_180\_A\_ising\_Bloch\_Anti.py : construct A centered mixed Ising-Bloch type 180° domain wall structure of tetragonal phase(Bloch component: antiparall )

T\_180\_B\_Ising.py : construct B centered Ising type 180° domain wall structure of tetragonal phase

T\_180\_B\_Ising\_Bloch\_Anti.py : construct B centered mixed Ising-Bloch type 180° domain wall structure of tetragonal phase(Bloch component: antiparall )

T\_180\_nm0 : construct (nm0)-oriented building block for DW structure construction

**Procedure**

First, prepare the POSCAR of building block for DWs structure construction.

Second, run corresponding script to construct DWs structure.

Note: for all scripts, the input file is “POSCAR.0” and output file is “POSCAR”. The size of supercell can be tuned by variable n in subfunctions of in the main function.

**Examples**

Example1: (100)-oriented A-centered(or B-centered) Ising(or Ising-Bloch) type 180° DWs structure

Scripts: T\_180\_A\_ising.py, T\_180\_A\_ising\_Bloch\_Anti.py, T\_180\_B\_Ising.py, T\_180\_B\_ising\_Bloch\_Anti.py

Take T\_180\_A\_ising.py as an example.

First, prepare the POSCAR of a unit cell, name the file as “POSCAR.0”. The “POSCAR.0” are as follows:

PbTiO3

1.00000000000000

3.8672948179140714 0.0000000000000000 0.0000000000000000

0.0000000000000000 3.8672948179140714 0.0000000000000000

0.0000000000000000 0.0000000000000000 4.0321616359809260

O Ti Pb

3 1 1

Direct

0 0.5 0.398257956

0.5 0 0.398257956

0.5 0.5 0.911755967

0.5 0.5 0.466524961

0 0 0

Then, construct the n×1×1 supercell containing DWs using “Python3 T\_180\_A\_Ising.py”. The output file is ”POSCAR”. The size of supercell can be tuned by n, which are in the main function at the last part of the script, as shown in Fig. 2.

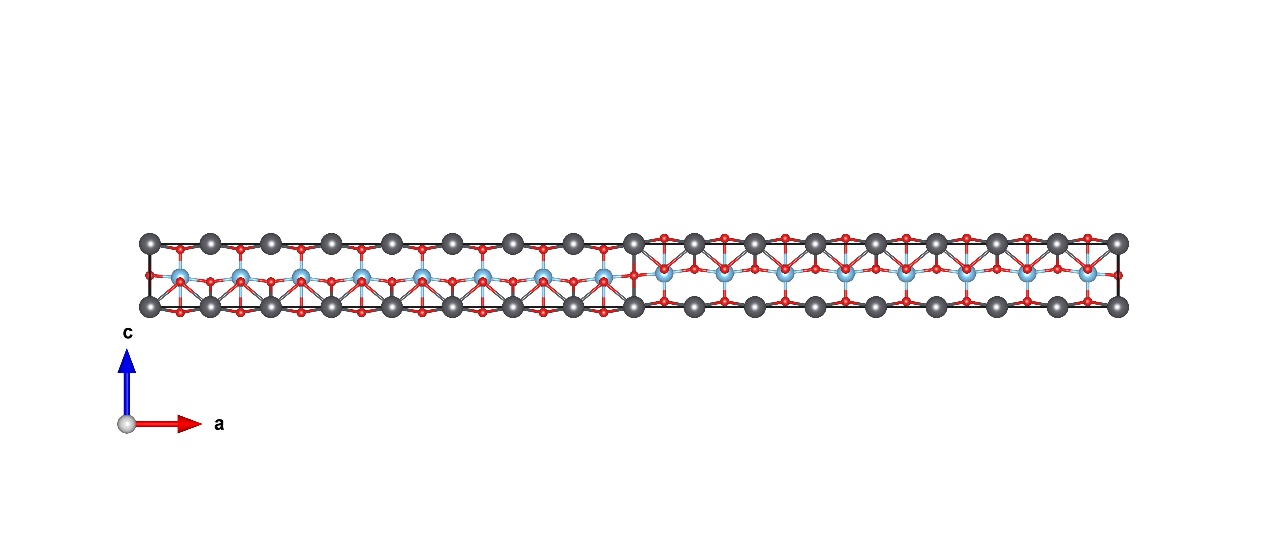


Fig. 1. A centered Ising type 180° domain wall structure of tetragonal phase

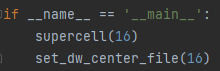


Fig. 2. Main function of T\_180\_A\_Ising.py script. The size of n×1×1 supercell can be tuned by variable n in two subfunctions. (here, n=16)

Example2: Tetragonal PbTiO3 : 180° (140)-oriented Ising type DWs structure.

Script : T\_180\_nm0 and T\_180\_A\_ising.py

First, prepare the POSCAR of a unit cell, name the file as “POSCAR.0”.

Second, obtain the POSCAR of the building block using “python3 T\_180\_nm0”, then name the file as “POSCAR.0”. The building block are shown in Fig. 3. The main functoin are shown in Fig. 4.

Third, construct the n×1×1 supercell containing DWs using “Python3 T\_180\_A\_Ising.py”.

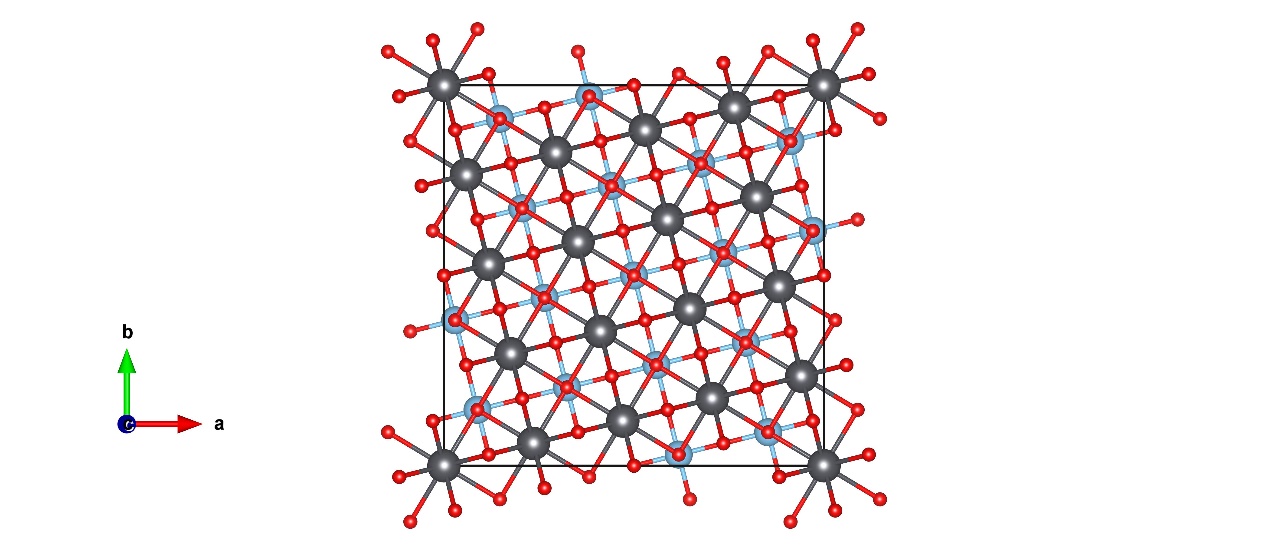


Fig. 3. Building block for (140)-oriented DWs structure construction of tetragonal phase

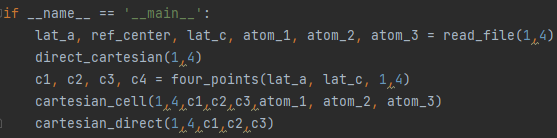


Fig. 4. Main function of T\_180\_nm0.py script. The orientation of building block can be tuned by variable n, m in five subfunctions.(here, n=1, m=4)

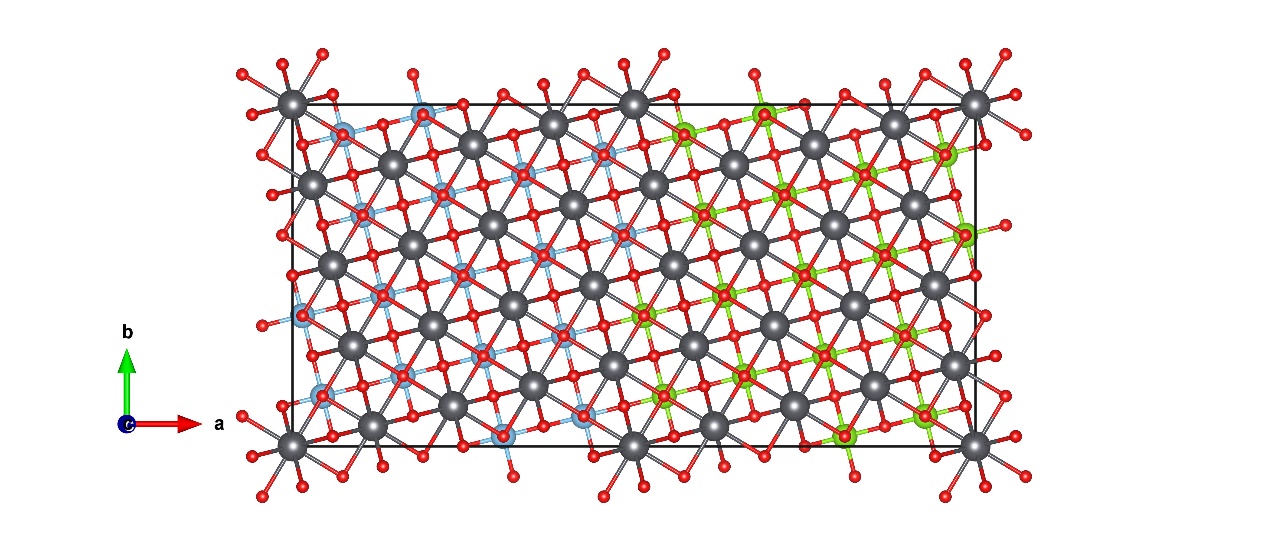


Fig. 5. 2×1×1 (140)-oriented DWs structure construction of tetragonal phase. Left part: polaization is along +z; Right part: polarization is along -z.

Example3: 90° domain wall of tetragonal PbTiO3

Script: T\_90.py

First, prepare the POSCAR of a unit cell, name the file as “POSCAR.0”.

Second, obtain the POSCAR file of the 90° DWs structure using “python3 T\_90.py”, as shown in Fig. 6.

Note: There are also some temporary files after running the T\_90.py script. Only the POSCAR is the final output file.

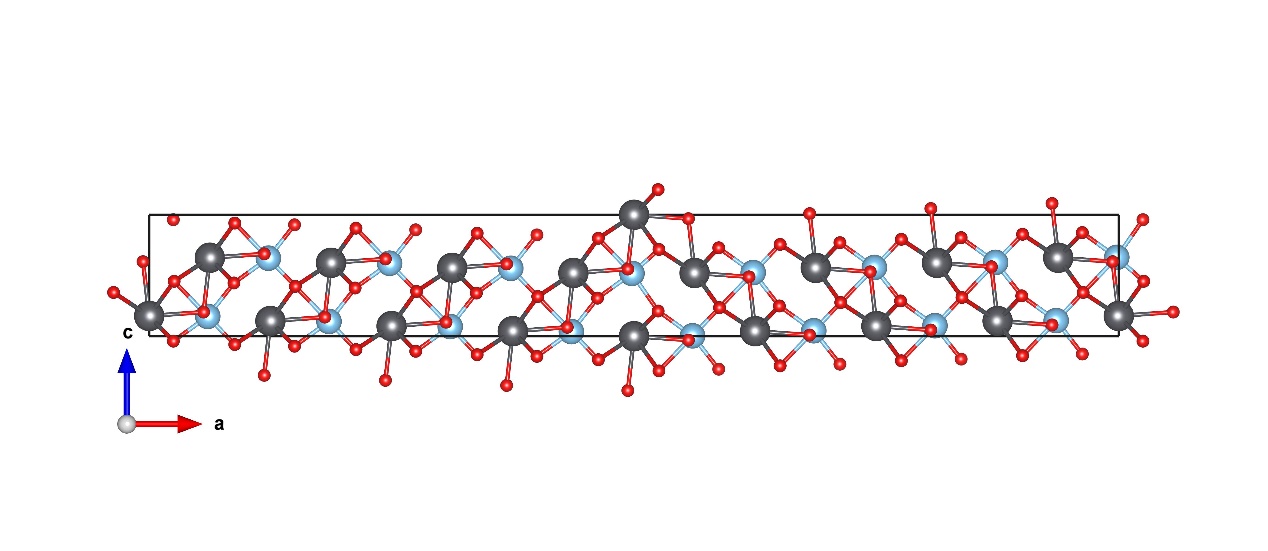


Fig. 6. 90° DWs of tetragonal ABO3 pervoskite

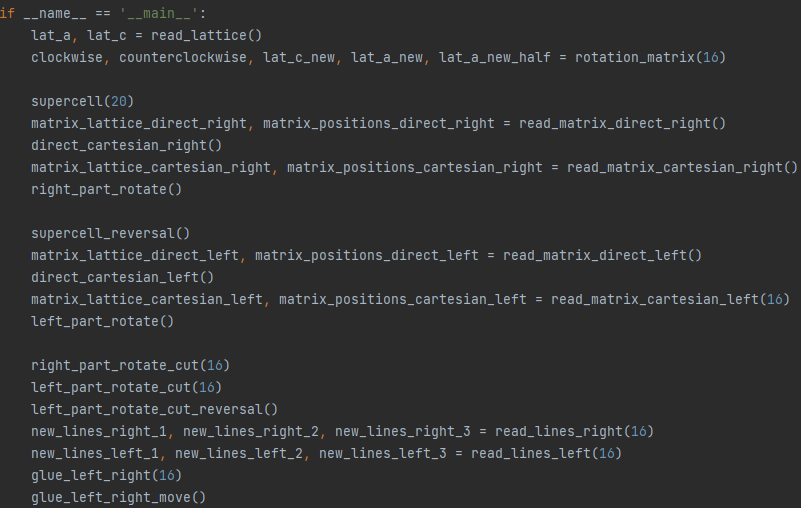


Fig. 7. Main function of T\_90.py script. The size of supercell can be tuned by variable n in seven subfunctions. (here, n=16)

Example4: 90° domain wall of orthorhombic BaTiO3

Script: O\_90.py

First, prepare the POSCAR of a 5-atom primitive cell, name the file as “POSCAR.0”, as shown in Fig. 7.

Second, obtain the POSCAR file of the 90° DWs structure using “python3 O\_90.py”, as shown in Fig. 8.

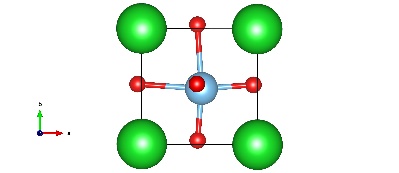


Fig. 7. 5-atom primitive cell

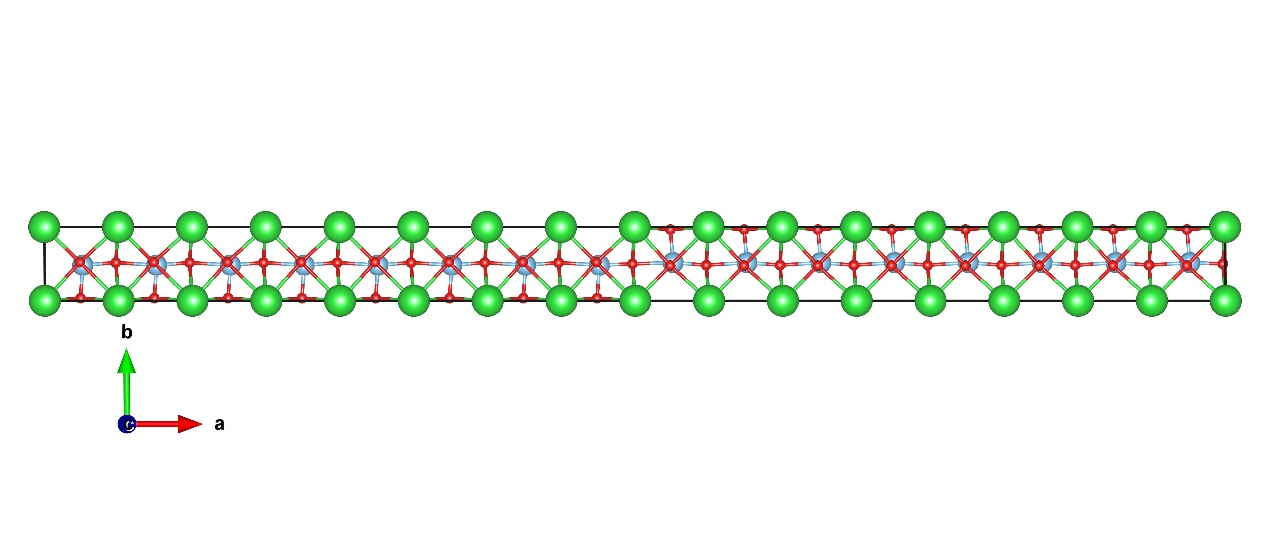


Fig. 8. 90° DWs of tetragonal ABO3 pervoskite

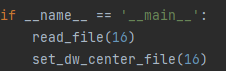


Fig. 9. Main function of O\_90.py script. The size of n×1×1 supercell can be tuned by variable n in two subfunctions. (here, n=16)

Example5: 180° domain wall of orthorhombic BaTiO3

Script: O\_180\_001.py, O\_180\_100.py

First, prepare the POSCAR of a 10-atom conventional cell, name the file as “POSCAR.0”, as shown in Fig. 10.

Second, obtain the POSCAR files of the (100)-oriented or (001)-oriented 180° DWs structure using “python3 O\_180\_100.py” or “python3 O\_180\_001.py”, as shown in Fig. 11.

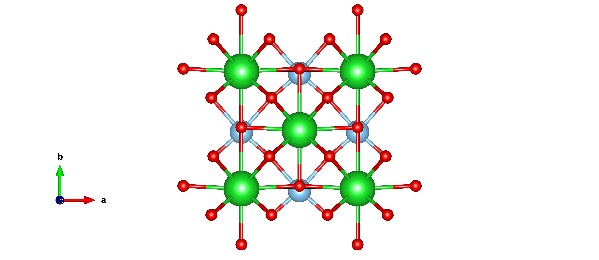


Fig. 10. 10-atom conventional cell

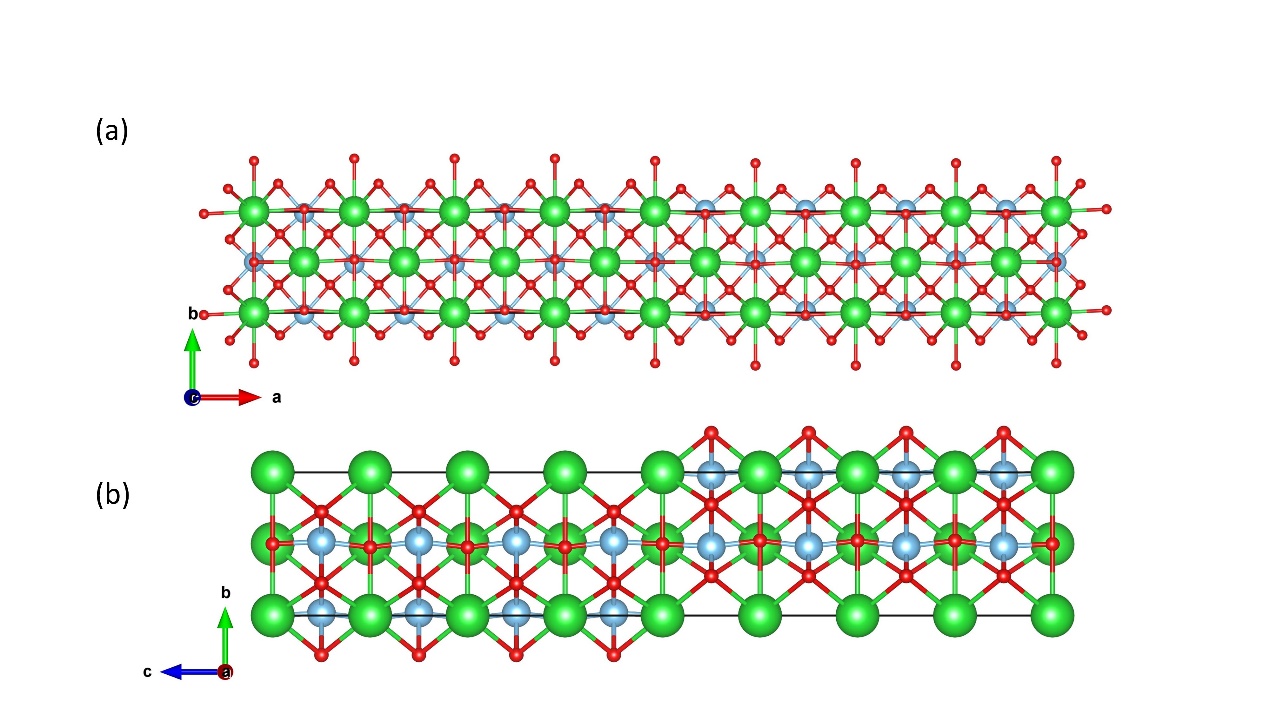


Fig. 11. (a) (100)-oriented and (b) (001)-oriented 180° DWs of orthorhombic ABO3 pervoskite

Example6: 109° domain wall of rhombohedral BaTiO3

Script : R\_109.py

First, prepare the POSCAR of a 5-atom primitive cell, name the file as “POSCAR.0”, as shown in Fig. 12.

Second, obtain the POSCAR files of the 109° DWs structure using “python3 R\_109.py” , as shown in Fig. 13.

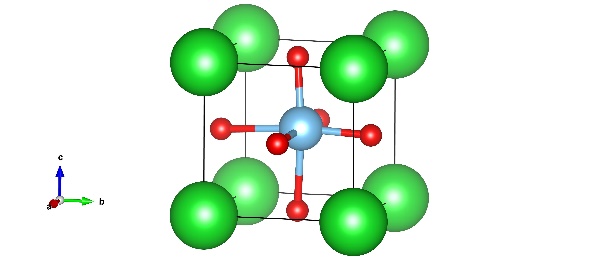


Fig. 12. 5-atom primitive cell of rhombohedral BaTiO3

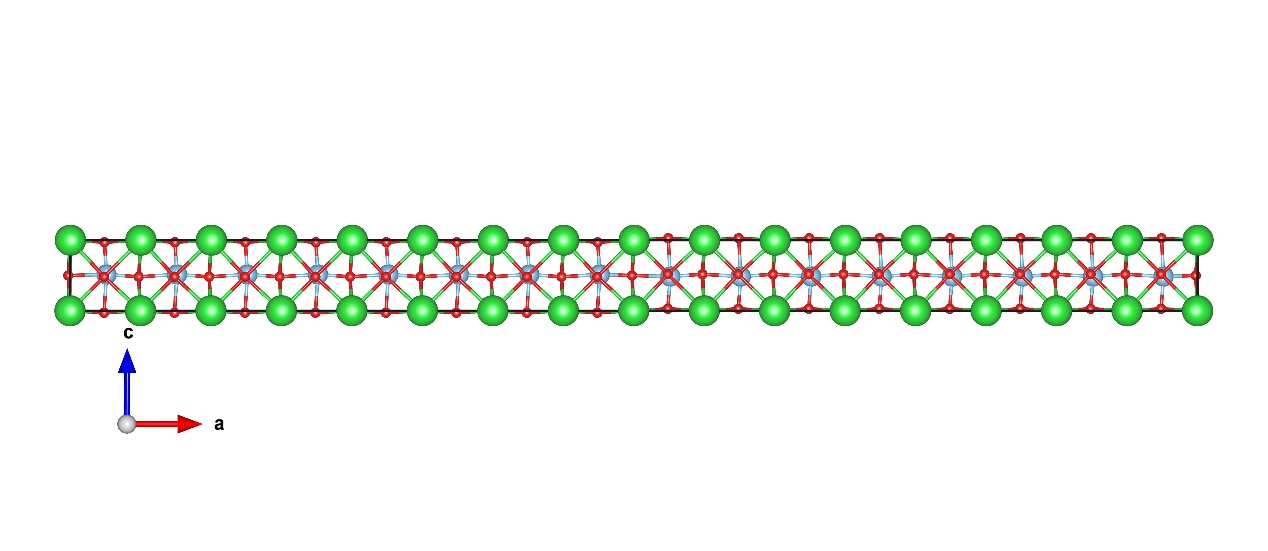


Fig. 13. 109° DWs of rhombohedral ABO3 pervoskite

Example7: 71° or 180° domain wall of rhombohedral BaTiO3

Script : R\_71.py and R\_180.py

First, prepare the POSCAR of a 10-atom building block, name the file as “POSCAR.0”. The building block can be obtained by VESTA using the transformation matrix“1 1 0 -1 1 0 0 0 1”, as shown in Fig. 14.

Second, obtain the POSCAR files of the 71° or 180° DWs structure using “python3 R\_71.py” or “python3 R\_180.py” , as shown in Fig. 15.

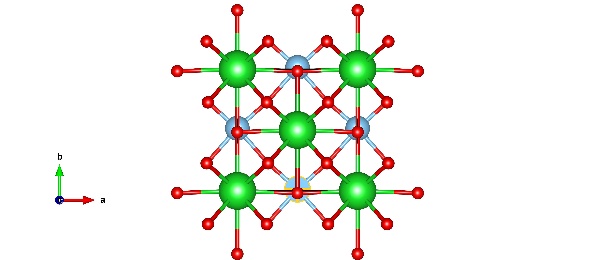


Fig. 14. 10-atom building block of rhombohedral BaTiO3

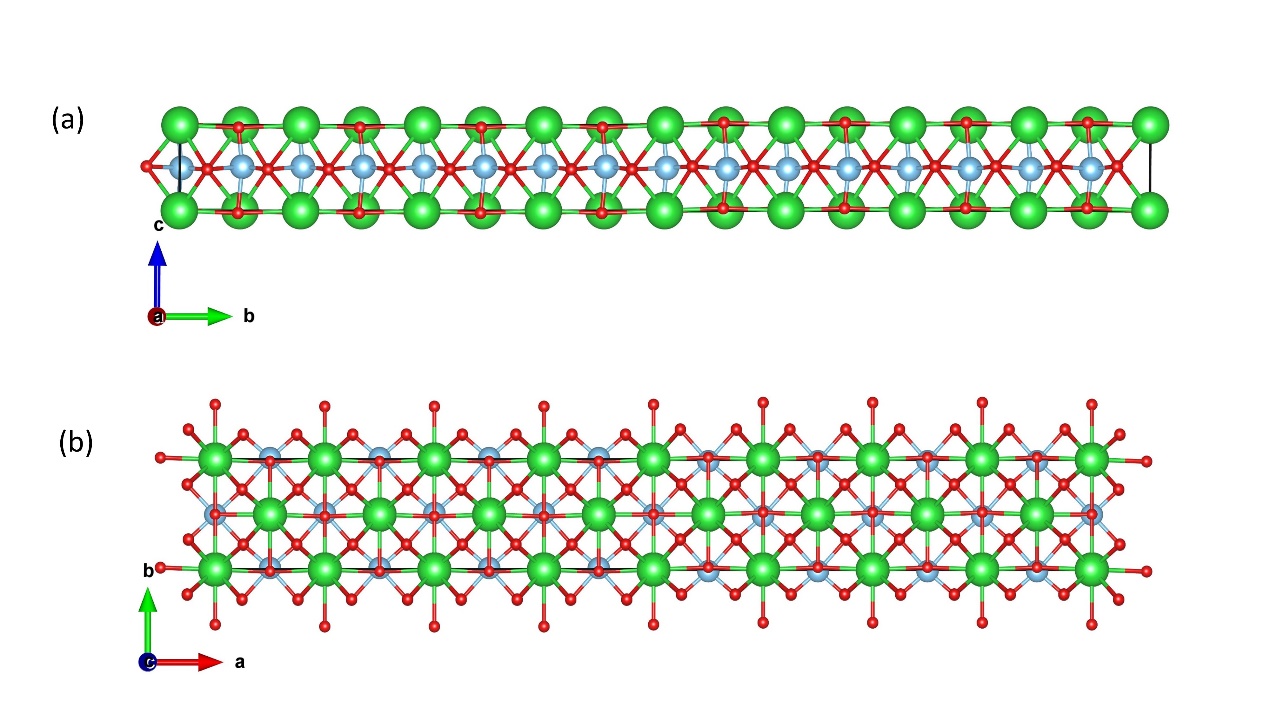


Fig. 15. (a)71° and (b)180° DWs of rhombohedral ABO3 pervoskite

Example7:

Cartesian\_direct.py : Convert Cartesian coordinate system to direct coordinate system using “Python3 Cartesian\_Direct.py”

Direct\_cartesian.py : Convert direct coordinate system to Cartesian coordinate system using “Python3 Direct\_Cartesian.py”