Anion substitution example For SOCCR

Updated June 12, 2018

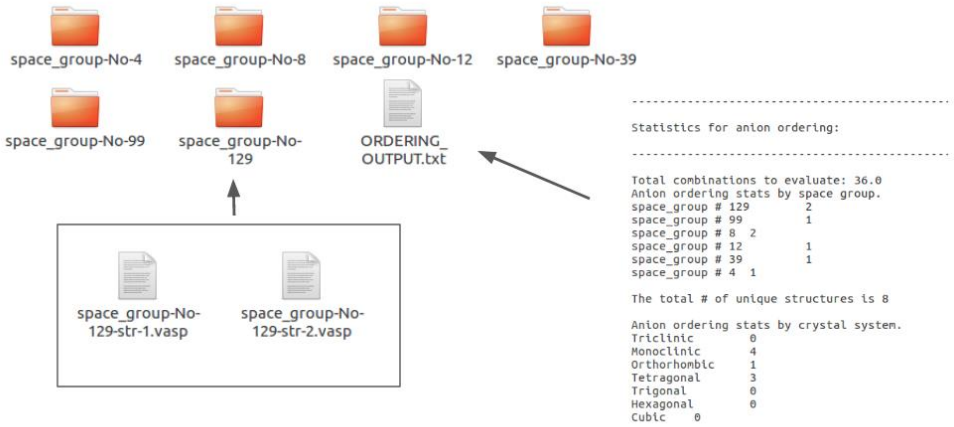
1. Download program from the github: https://github.com/MTD-group/MTDGderivative-structures
    a. Installation instructions found in the README.pdf
2. Setting up files for program requires an input file and the POSCAR of the parent structure
    a. POSCAR can be named anything, just needs to be in the VASP format
        i. POSCAR for $KNaNbF_6$ in *P*4/*nmm* provided
    b. Input file explained:

Input file

cenat = Na — Focal atom which anion ordering occurs. Usually the cation
cations = Na K Nb — Cations in parent structure
anions = F — Anions in parent structure
ionsub = O — Element you want in derivate structure
CALC_MODE = 2 — 1 is for cation substitution, 2 is for anion substitution, 3 is for random distribution of all sites without any consideration of ordering
subratio = 1/6 — Ratio of substitution (So 1/6 of F will be replaced with O)
P1_EVAL = .FALSE. — Doesn't report structures that have *P*1 symmetry
NPROCS = 1 — Number of processors to use. Parallel computing allowed
maxbin = 250 — Max# of structures to analyze per symmetry (Improves Scalability)
RAND_SEARCH = .FALSE. — Instead of iterating through every possible structure, it randomly sample a max number of structures that improves scalability of program. Assumes the majority of unique structures are captured. Helps a lot with large unit cells.
#max_rand = 800 — Max number of structures to sample if RAND_SEARCH Used
#SAVEIN = /home/?? — If you want to save the results in a folder different from where it was ran

3. Running the program
    a. Type following into terminal: $SOCCR -i input-file.ini -p POSCAR
    b. Using provided input files and 1 processor, this should take ~1 second
4. Output from program
    a. The unique structures are separated into folders by space group and they are in VASP format

b. ORDERING_OUTPUT.txt file has useful statistics about the results. Image of outputs are shown below



5. How to plot
    a. In the same folder as plotting.py and the ORDERING_OUTPUT.txt file use $python plotting.py
    b. Results shown below