

# A manual of using a template GenX script to setup and optimize surface structural models

Canrong Qiu

cqiu@alaska.edu

This is a manual explaining how to customize a template of the GenX script (genx folder/scripts/genx\_script\_standard\_RAXS\_CTR\_beta.py) to setup surface structure model. This script is specifically developed for considering sorbate adsorption on hematite (1-102) surface under different binding configurations, including monodentate, bidentate, tridentate and outer-sphere modes. Currently the local structure of a surface complex can be chosen for any sorbate with local structure of either a trigonal pyramid (Pb adsorption) or an octahedral (Sb adsorption) or tetrahedral (As adsorption).

## Inherent features of a hematite (1-102) surface

- 1) Two chemically distinct domains (corresponding to two termination patterns) with each having two symmetry related domains with half unit cell step height difference.
- 2) The surfaces with two termination patterns could have a different step height relationship depending on how the sample is prepared (CMP or annealed).
- 3) Surface functional groups appears in pair, which determines that any potential binding site will have a corresponding symmetry related site to be considered explicitly with its geometrical relationship to the parent site defined by a glide plane (b glide).

## Features of CTR/RAXR template script

- 1) You can fit both CTR and RAXR models
- 2) You can consider as many domains as you want with the surface termination pattern specified;
- 3) You can freely choose different combination of binding sites. It will be very handy if you have multiple sites occurring simultaneously (like inner-sphere and outer-sphere at the same time).
- 4) Different sites could be stacked up on the same domain or be distributed to different domains.
- 5) You can consider more than one type of sorbate
- 6) Sorbates could be distributed on different domains or on a single domain to consider for the forming of ternary complex species or clusters
- 7) Water pairs could be added on top easily by specifying the reference anchor atoms.
- 8) You may consider layered water/sorbate structure, which only affect the specular rod.
- 9) Automatically group symmetrically related atoms (including sorbate atoms) together based on the glide plane symmetry
- 10) You can choose to impose bond valence constraint during the model fitting
- 11) You can specify the protonation schemes of surface atoms
- 12) The fitting parameter tables could be automatically generated based on your model setup
- 13) Some model files (e density profiles, structure xyz file, CTR profiles, RAXR profiles) could be generated and saved on local disk in the post-run checking step as you wish. Those dumped files could be called by different functions available to do the associated graphing.

## How does bond valence constraint work in the script?

- 1) Pre-define a set of coordinated members
- 2) Specify the radius of the searching sphere (ideal bond length+offset Å by default).  
The searching radius is based on the associated ideal bond length, which is added by an offset value. For example, considering the pair of O-Pb, then the ideal bond length is 2.19 Å, then the searching radius will be 2.39 Å if the

offset is 0.2 Å. That way, the same oxygen atom will apply different searching radius if the counter ion is different (O-Fe Vs O-Pb).

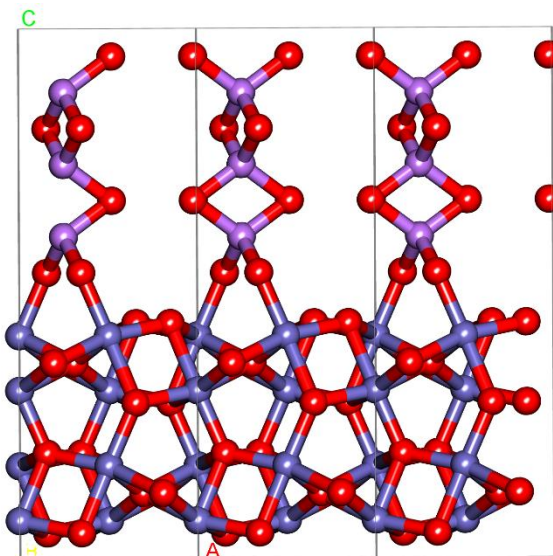
- 3) Calculate the bond valence sum of the atom under consideration within the searching sphere
  - a. In the case of cation-anion, of which both are coordinated members, use the bond valence-bond strength relationship for the calculation of bond valence
  - b. In the case of cation-anion, one of which does not belong to the coordinated members, calculate the bond valence from *step a* and then scale it by an arbitrary penalty factor.
  - c. In the case of cation-cation or anion-anion (like O-O), the calculation of the bond valence is based on arbitrarily assigning a value to the  $r_0$ , which is a big value (corresponding to a high bond valence value purposely as a penalty) if the inter-atomic distance is lower than the cutoff limit (the limit could be specified) or otherwise an extremely small value (the bond valence will be too small to make contributions to the total bond valence sum).
- 4) Calculate the numerical difference between the formal charge of the atom under consideration and the calculated total bond valence sum after counting the contribution from protonation or hydrogen bond. Such difference (higher than 1) will serve as a factor to scale the original FOM.
- 5) Bond valence consideration for symmetry site pair

Due to the intrinsic symmetry for  $r_{\text{cut}}$  hematite surface, any one site will have a symmetry related site associated with it. To make the modeling more effective, both sites are assigned to a single domain instead of being assigned to two equivalent domains. So such a domain with both sites assigned is actually an average structure comprising of two symmetry domains. To correctly calculate the bond valence, only one site should be counted (you may have one surface functional group being attached to both sorbates, which will not physically occur). The way that the script deal with this problem is to select a partial set of the interfacial atoms, which include all surface atoms but one set of sorbates (metal and the associated distal oxygens possibly). The selection of a partial sorbate set will result in one problem associating with the miscalculation of bond valence sum for the counterpart anchored sites, since the counterpart anchored oxygens will be treated as unbounded with sorbate in the search mode. The under-coordinated counterpart anchored oxygens may not be assigned any proton/hydrogen bond if the coordination number is already saturated according to the calculation performed before `sim` function, which is based on the complete structure. The way to fix that issue is by deleting the counterpart oxygens in the pool of atoms being considered for bond valence calculation. As a result, all atoms except for the counterpart set of sorbates as well as anchored oxygens will be considered in the following bond valence calculation. Due to the symmetry constraint, you should know the unconsidered set of atoms should show equivalent bond valence analysis results compared to the associated counterpart atoms.
- 6) It should be noted that the bond valence constraint should be employed only when the structure involves surface species stacking in  $z$  direction rather than in-plane directions. For example, you can consider a structure with inner-sphere species coexisting with outer-sphere or ternary complex species, but you should not consider two different types of inner-sphere species within the same unit cell in the bond valence constraint. If you consider more than one inner-sphere species within one unit cell, the bond valence constraint will become physically unreliable, and a right way for that is to separate two species and assign each one on a separated unit cell (multiple unit cells). Or you can just turn off the bond valence constraint, and optimize the average structure for a quick check of the proportion of each species, which should be further refined in the following multi-unit cell model within bond valence constraint turned on.

## Rules to add sorbates

- 1) Addition of water layers: you can specify even number of water molecules with a reference consisting of two atoms (then the reference point will be at the center point of those two atoms) or one atom position (then the reference point will be that position); the water positions will be calculated with an alpha angle in degree and a vertical shift in Å and a  $R$  constrained by the condition of  $|y_1 - y_2| = 0.5$ . Don't forget to allow for the  $dy$  movement for water pairs during fitting.
- 2) Addition of a metal in a Monodentate/Bidentate mode/tridentate mode/outer-sphere mode: with a local structure of trigonal pyramid for Pb sorbate or octahedral for Sb sorbate

- a. Monodentate: not tested yet, so the code could be buggy
  - b. Bidentate: you need two anchor atoms and one reference atom so that you can set the coordinated system properly.
    - i. Pb case: you need specify arguments of top\_angle (or edge length R under USE\_TOP\_ANGLE=False), phi (rotation angle), edge\_offset, and angle\_offset, edge\_offset2 (angle\_offset and edge\_offset2 are relevant to the distal oxygen)
    - ii. Sb case: you have only one freedom that is phi (rotation angle)
  - c. Tridentate: you need three anchor atoms but don't need reference atom here
    - i. Pb case: Not tested
    - ii. Sb case: you have no freedom in this case
  - d. Outersphere: You will need to specify one reference point, which is the starting point of one of the outersphere site. And the other symmetry related site will be calculated automatically by using intrinsic symmetry relationship.
  - e. Add distal oxygen wildly: you can choose to add distal oxygen wildly by setting ADD\_DISTAL\_LIGAND\_WILD=True; then for each distal oxygen pairs (two oxygens) you need r, theta and phi arguments and the position of the associated metal to calculate the ligand position.
- 3) The other way to assign the binding sites is by explicitly specifying the fractional coordinates of the sorbate as well as the distal oxygens. It is a preferred way for site assignment if you have a good guess of starting points for them.
- 4) Adding ternary complex species: you can add second species (same or different element type) on top of the first one by anchoring on the distal oxygens from the first species. You may stack several species on top of each other forming a cluster-like structure (like the structure model shown below).



### Step by step procedures to set up model

- ❖ Fire up GenX (Python genx\_gui.py)
- ❖ Format your background subtracted dataset so that it can be loaded using uaf\_CTR\_raxs\_2 loader. If you start with a data format (\*.lst) exported from HDF integrator (available inside TDL package), then you could use data\_formatting.py (saved in GenX folder\\supportive\_functions) to create a data file in a right format for the data loader mentioned above.
- ❖ Import the dataset, and copy the GenX template script (saved in GenX folder\\scripts\\genx\_script\_standard\_RAXS\_CTR\_beta.py) inside the Script panel. Then set parameters inside the main setup zone in the template script, and note that most of the variables could be left unchanged. The ones in red are variables you will need to specify for building a structure model (read following section for details).

- ❖ Print the table file (set the running mode to False, then a “table.tab” file will be made and saved in D:// by default after you hit simulation button) and import it to GenX file, and edit the fitting parameter (a lot of redundant items need to be deleted) table if necessary.
- ❖ Set the running mode back to True, and submit the file to super computer system for model optimization. Way to go!!!!

#### #####Explanation of each variable inside main setup zone#####

##### **running\_mode(bool)**

if true then disable all the I/O function

##### **SORBATE**(list of list with each list item containing the sorbate element in each domain)

element symbol for sorbate

the shape of SORBATE is the same as pickup\_index

##### **BASAL\_EL**(a list of elements to specify the anchor reference to the ternary complex)

only used in the domain containing ternary complex species

The first item in each item list is always None, since the first one is referenced to the substrate surface

By default each item after the first one has a referenced element from the previous element)

##### **pickup\_index**(a list of index list with items from the match index table above)

representative of different binding configurations for different domains

make sure the half layer indexes are in front of the full layer indexes

In this new version, you can have multiple sites being assigned simultaneously on the same domain

For example, in the case of [[0,6,6],[4],[10,14]] there are three sites assigned to domain1, i.e. bidentate site and the other two outer-sphere site

##### **sym\_site\_index**(a list of list of [0,1])

a way to specify the symmetry site on each domain

you may consider only site pairs in this version ([0,1])

The shape is the same as pickup\_index, except that the inner-most items are [0,1] instead of match index number

It will be set up automatically

##### **full\_layer**(a list of either 0 or 1 with 0 for short and 1 for long slab)

used to specify the step for full layer termination, the items in this list must have a one to one corresponding to the items appearing in the pick\_up\_index for FL

##### **half\_layer**(a list of either 2 or 3 with 2 for short and 3 for long slab)

Analogous to full\_layer but used for half layer termination case

##### **full\_layer\_pick**(a list of value of either None, or 0 or 1)

used to specify the full layer type, which could be either long slab (1) or short slab (0)

don't forget to set None for the half layer termination domain

Again Nones if any must be in front of numbers (Half layer domains in front of full layer domains)

concerns about None has been automatically setup in this new version

##### **half\_layer\_pick**(a list of value of either None, or 2 or 3)

Analogous to full\_layer\_pick but used for half layer termination

##### **OS\_X(Y,Z)\_REF**(a list of None, or any number)

set the reference coordinate xyz value for the outer-sphere configuration, which could be on either HL or FL domain

these values are fractional coordinates of sorbates

if N/A then set it to None

such setting is based on the symmetry operation intrinsic for the hematite rcut surface, which have the following relationship

$x_1+x_2=0.5/1.5$ ,  $y_1-y_2=0.5$  or  $-0.5$ ,  $z_1=z_2$

The shape is like  $[[[]],[[]]]$ , each item corresponds to different domains

The number of items within each domain is twice (considering symmetry site pair) the number of sorbate for that domain

**DOMAIN\_GP**(a list of list of domain indexes)

use this to group two domains with same surface termination (HL or FL) together

the associated atom groups for both surface atoms and sorbates will be created (refer to manual)

This feature is not necessary and so not supported anymore in this version.

**water\_pars**(a lib to set the interfacial waters quickly)

This water molecules are regarded as adsorbed water molecules with lateral and vertical ordering which will have effect on both the specular and offspecular rods

you may use default which has no water or turn the switch off and set the number and anchor points

**layered\_water\_pars**(a lib to set layered water structure)

layered water structure factor only have effect on the specular rod

Based on the equation(29) in Reviews in Mineralogy and Geochemistry v. 49 no. 1 p. 149-221

key of 'yes\_OR\_no': a list of 0 or 1 to specify whether or not considering the layered water structure

key of 'ref\_layer\_height' is a list of atom ids (domain information not needed) to specify the reference height for the layered water heights

**layered\_sorbate\_pars**(a lib to set layered sorbate structure)

pretty much the same as layered\_water\_pars

key of 'el' is the symbol for the resonant element

**USE\_BV**(bool)

a switch to apply bond valence constrain during surface modelling

**TABLE\_DOMAINS**(list of 0 or 1, the length should be higher than the total domain number)

specify whether or not generate the associated pars for each domain

[0,1,1] means only generate the pars for last two domains

**RAXR\_EL**(resonant element)

**NUMBER\_SPECTRA**(number of RAXR spectras)

Note each spectra, there will be an independent set of fitting parameters (a,b,A,P)

**RESONANT\_EL\_LIST**(a list of integer number (either 1 or 0))

Used to specify the domain containing resonant element

0 means no resonant element on the domain

1 means considering resonant element on the domain

**E0**=13000

Center of Scan energy range for RAXR data

**F1F2\_FILE**="Pb.f1f2"

Absolute file path for the f1f2 file containing anomalous correction items at each energy

**F1F2**=None

Global variable to hold the f1f2 values after loading the f1f2 file

### **COVALENT\_HYDROGEN\_RANDOM**(bool)

a switch to not explicitly specify the protonation of surface functional groups

different protonation scheme (0,1 or 2 protons) will be tried and compared, the one with best bv result will be used

### **BV\_OFFSET\_SORBATE**(a list of number)

it is used to define the acceptable range of bond valence sum for sorbates

[bv\_eachbond\*N\_bonds-offset,bv\_eachbond\*N\_bonds] will be the range

set a random number for a clean surface (no sorbate), but don't miss that

### **SEARCH\_RANGE\_OFFSET**(a number)

used to set the searching range for an atom, which will be used to calculate the bond valence sum of sorbates

the radius of the searching sphere will be the ideal bond length plus this offset

### **commands**(a list of str to be executed inside sim function)

eg. ['gp\_O1O2\_O7O8\_D1.setoc(gp\_Fe4Fe6\_Fe10Fe12\_D1.getoc())']

used to expand the functionality of grouping or setting something important

### **USE\_COORS**(a list of [0,0] or [1,1] with two items for two symmetry sites)

you may want to add sorbates by specifying the coordinates or having the program calculate the position from the geometry setting you considered

eg1 USE\_COORS=[[0,0]]\*len(pickup\_index) not use coors for all domains

eg2 USE\_COORS=[[1,1]]\*len(pickup\_index) use coors for all domains

eg3 USE\_COORS=[[0,0],[1,1],[1,1]] use coors for only domain2 and domain3

### **COORS**(a lib specifying the coordinates for sorbates)

keys of COORS are the domain index and site index, ignore domain with no sorbates

len(COORS[(i,j)]['sorbate'])=1 while len(COORS[(i,j)]['oxygen'])>=1, which is the number of distal oxygens

make sure the setup matches with the pick\_up index and the sym\_site\_index as well as the number of distal oxygens

if you dont consider oxygen in your model, you still need to specify the coordinates for the oxygen(just one oxygen) to avoid error prompt

### **O\_NUMBER\_HL/FL**(a list of list of [a,b],where a and b are integer numbers)

one to one corresponding for the number of distal oxygens, which depend on local structure and binding configuration

either zero oxygen ligand or enough ligands to complete coordinative shell

### **O\_NUMBER\_HL/FL\_EXTRA**(used to define the distal oxygen number for a surface species binding to the distal oxygen of basal element)

### **MIRROR**(a list of true or false)

Used to specify the way you add a distal oxygen to a surface complex with monodentate or bidentate binding configuration

Or in a case of tridentate binding mode with octahedral local structure

### **SORBATE\_NUMBER\_HL/FL**(a list of list of [a], a can be either 1 or 2 or 0 for clean surface)

If considering two symmetry sites, then a=2

If considering one site (distribute the two on two different domains), then a=1

If considering clean surface, then a=0

### **SORBATE\_NUMBER\_HL/FL\_EXTRA**(used to specify the number of outer-part of ternary complex species)

### **COUNT\_DISTAL\_OXYGEN**(bool)

True then consider bond valence also for distal oxygen,otherwise skip the bv contribution from distal oxygen

#### **ADD\_DISTAL\_LIGAND\_WILD**(list of bool)

the distal oxygen could be added by specifying the pars for the spherical coordinate system (r, theta, phi), which is called wild here, or be added in a specific geometry setting for a local structure (like tetrahedra)

you can specify different case for different domains

and this par is not applicable to outersphere mode, which should be set to None for that domain

#### **DOMAINS\_BV**(a list of integer numbers)

Domains being considered for bond valence constrain, counted from 0

#### **BOND\_VALENCE\_WAIVER**(a list of oxygen atom ids [either surface atoms or distals] with domain tag)

When each two of thoes atoms in the list are being considered for bond valence, the valence effect will be ignored no matter how close they are

Be careful to select atoms as bond valence waiver

#### **GROUPING\_SCHEMES**(a list of lists with two items, with each item being the domain index starting from 0)

Define how you want to group the surface atoms together from two different domains with same termination type

A function will generate all the associated commands to do the grouping

[[0,1]] means group surface atoms from the first (0) and second domain(1)

If you dont want to do any grouping, set this to be []

#### **GROUPING\_DEPTH**(a list of integers less than 10)

Define how deep you want to group your atoms. You can define a maximum grouping depth to 10

You should count the atom layers upward from the 10th atom layer

[6,10] means you want to group the 5th atom layer to 10th atom layer for domain 1 and group all top ten atom layers together for domain2

Don't forget that you have a Iron layer which is explicitly included in HL but the occ set to 0 to account for the missing Fe sites

So you should count that atom layer too when considering the grouping depth

#####

#### **Explanation of other global variable (you barely need to change anything here)**

- 1) output\_file\_path: place for all those output files
- 2) BV\_TOLERANCE: a range of acceptable offset for |formal charge-bond valence sum|
- 3) USE\_TOP\_ANGLE: It will take effect only for Pb complex binding under Bidentate mode. See details in the next section.
- 4) INCLUDE\_HYDROGEN: Add hydrogens explicitly, and calculate the bond valence associated with a proton using an equation (not specifically assign the protonation schemes). This feature is not successful, since it will end up with too many variables (variable explosion).
- 5) COHERENCE

It is used to set the coherent rule, which is a list with each item of a library having one key (either True or False corresponding to coherent or incoherently addition of the domains, which are specified by index numbers by the value of the library) ; magnitude of structure factor from each library will be calculated before being summed up. Some examples are as follows:

[[True:[0,1]],{False:[2,3]}]: coherent addition of domain1 and domain2, whereas incoherent addition of domain 3 and domain 4

Note that results of [{True:[0,1]],{True:[2,3]}} are different from results of [{True:[0,1,2,3]}]

Note that the chemically equivalent domains (like domain1A and domain 1B) are also added incoherently.

- 6) **SEARCH\_MODE\_FOR\_SURFACE\_ATOMS**: If using bv constraint, you can either specify the coordinated members for each surface atoms (False) or search atoms within a spherical range with radius (2.5Å by default) adjustable (True).
- 7) **R0\_BV**: A library of r0 values to be used in the bond valence bond strength relationship. The keys are the tuples of two atom elements (the order does not matter). Such a library must comprise all possible cases.
- 8) **UPDATE\_SORBATE\_IN\_SIM**: There are two different ways to fit the sorbates: First way is by doing some geometrical operations (like rotation along some axis); second way is by defining a 3d box for sorbates to move inside. If it is set to True, then the first way will be applied, otherwise the second way will be applied.
- 9) **SORBATE\_ATTACH\_ATOM**: It is a list of anchor atoms for binding sorbates. A list of one, two and three items corresponds to Monodentate, Bidentate and tridentate binding mode. For outer-sphere mode, it is an empty list.
- 10) **SORBATE\_ATTACH\_ATOM\_OFFSET**: It has the same shape as the **SORBATE\_ATTACH\_ATOM**, and it is used to define the translation offset for the anchor atoms. It can be any symbols from ['-x', '+x', '+y', '-y', None]. '-x' represents translation along negative x axis for one unit, and so on for the other symbols. None represents no translation.
- 11) **ANCHOR\_REFERENCE**: It is an id for an atom with a position, which will be used as a reference to create a spherical coordinate system to calculate the sorbate position. It is specifically for bidentate binding configuration. Under the spherical coordinate system described above, the zero rotation angle will correspond to a sorbate at a position on a plane formed by two anchor atoms and such a reference point.
- 12) **ANCHOR\_REFERENCE\_OFFSET**: It is the translation offset for the anchor reference point.
- 13) **COVALENT\_HYDROGEN\_RANDOM**: a switch to consider protonation randomly. Bond valence sum will be calculated based on a non-protonated, single protonated or doubly protonated state, whichever will be applied if it gives rise to the most satisfactory bond valence sum.
- 14) **POTENTIAL\_COVALENT\_HYDROGEN\_ACCEPTOR**: A list of atoms to be considered for the protonation on a random basis.
- 15) **COVALENT\_HYDROGEN\_ACCEPTOR**: A list of atoms to be considered for addition of protons explicitly. And it takes effect only when the **COVALENT\_HYDROGEN\_RANDOM** is set to False.
- 16) **COVALENT\_HYDROGEN\_NUMBER**: Define the number of protons to be added with a one to one corresponding to atoms defined in **COVALENT\_HYDROGEN\_ACCEPTOR**.
- 17) **POTENTIAL\_HYDROGEN\_ACCEPTOR**: a list of atoms to be considered for accepting bond valence from a possible hydrogen bond. May or may not count the hydrogen bond depending on the bond valence saturation state.

## Output files

A bunch of files will be created when you hit the simulation button with **running\_mode** being set to False. These files will be saved in the "output\_file\_path" you specified in the main setup zone.

### Files include:

- 1) **Model\_domainI.cif** (I is integer from 1 and it depends on the number of domains in your model. Eg, if you have two domains in your model, it will spit out **Model\_domain1.cif** and **Model\_domain2.cif**): it is a cif file used for structure visualization using any molecular structure visualization tool<VESTA recommended>, and it should be noted the file contains structural information for one set of sorbate, the other symmetry related set of sorbates will not be contained in this file.
- 2) **Model\_domainI.xyz**: xyz file for structure visualization (contains only one set of symmetry related site pair).
- 3) **Model\_domainI\_dsv.xyz**: the same as above except that it contains information for both set of symmetry sites.
- 4) **Table.tab**: table file used to setup the fitting parameters, it could be imported into the gx file. It saves a lot of typing for you.
- 5) **Temp\_plot**: dumped file, which could be restored for CTR profiles plotting. (Read details about plotting below)
- 6) **Temp\_plot\_raxr**: dumped file, which could be restored for RAXR profiles plotting if you have RAXR dataset in the model. (Read details about plotting below)
- 7) **Temp\_plot\_edens**: dumped file, which could be restored for electron density profiles plotting. (Read details about plotting below)



- 8) Temp\_plot\_fourier\_synthesis: dumped file, which could be restored for electron density profile of resonant element using Fourier synthesis if you have RAXR dataset in the model. (Read details about plotting below)
- 9) Model\_domain/A\_publication\_combined.dat: information exported for the best fit model. Make sure you have exported the best fit parameter file saved in the output\_file\_path with a name of “test.tab” exactly. If you don’t have the test.tab file in the right place, this file will not be created. There are 9 columns in this file:  
Element X(fra) Y(fra) Z(fra) dx(errors in Angstrom) dy(errors in Angstrom) dz(errors in Angstrom) u occ

#### How to plot CTR, RAXR, electron density profiles:

The routine for plotting is create\_plots.py saved in “\$GenX folder\$/supportive\_functions”. Change working directory to that position, and run the command “execfile(“create\_plots.py”)”. Make sure you change some flags and file paths accordingly before you run this program.

#### **Hints for code extension for considering a different substrate in the future**

The most important thing is to identify and define the symmetry operation for a specific substrate. Locate all possible binding sites and index those sites the way as in the hematite case.

## **Appendix**

### **Naming rules**

#### 1. IDs for atoms within each domain

##### 1) Surface atoms

- a. Half layer termination. O1\_n\_m\_D1A: n can be any number from 1 to 20, and m can be either 0 (top slab) or (lower slab). Fe1\_n\_m\_D1A: n can be any number from [4, 6, 8, 9, 10, 12], and m has the same definition as surface oxygens. The suffix will tell you the information of domain is affiliated to. For example, D1A means that it belongs to the domain 1 with higher symmetry slab (A or lower symmetry slab if it is suffixed by B).
- b. Full layer termination (long): O1\_n\_m\_D1A: n can be any number from 1 to 20, and m can be either 0 (top slab) or (lower slab). Two more cases are O1\_11\_t\_D1A and O1\_12\_t\_D1A specifically representing the two terminal oxygens. Fe1\_n\_m\_D1A: n can be any number from [2, 3, 4, 6, 8, 9, 10, 12], and m has the same definition as surface oxygens.
- c. Full layer termination (short): O1\_n\_m\_D1A: n can be any number from 1 to 20 (5-20 for top slab, 1-20 for lower slab), and m can be either 0 (top slab) or (lower slab). As a comparison, now two terminal oxygens have ids of O1\_5\_0\_D1A and O1\_6\_0\_D1A. Fe1\_n\_m\_D1A: n can be any number from [2, 3, 4, 6, 8, 9, 10, 12], and m has the same definition as surface oxygens.

2) Sorbates: Pbn\_D1A, n can be any integer number, Pb can be replaced by any other element.

3) Distal oxygens: HOn\_Pbm\_D1A, n and m can be any integer number. It means n<sup>th</sup> (like third if n=3) distal oxygen of Pbm (like Pb1).

4) Waters: Osn\_D1A, n can be any integer number from 1.

#### 2. Atom groups

##### 1) Surface atoms:

gp\_O1O2\_O7O8\_D1: group atoms of O1 and O2 from domain1A to O7 and O8 from domain1B together using the associated glide plane symmetry relationship for dx dy dz movements.

gp\_O1O2\_O7O8\_D1\_D2: group O1, O2, O7 and O8 for domain 1 and domain 2 (including A, B subdomains) together using the associated glide plane symmetry relationship for dx dy dz movements.

##### 2) Sorbates:

gp\_sorbates\_set1\_D1: group first set (what “set1” means, the set index has an increment of 1) of sorbates (including metals and distal oxygens) together without considering symmetry relationship (used to set equal occupancy for metal and its distal oxygens).

gp\_Pb\_set1\_D1: group two symmetry related Pb atoms together (4 in total if considering those for the symmetry related domains). Pb can be replaced with another other element symbol; set1 means first set consisting of two

symmetry related atom within each domain (you can have multiple sets if you consider multiple sites being occupied simultaneously). And also note that the adjacent set indexes are 2 apart, so it goes from set1 to set3 to set5 and so on. It is used to set equal u and oc for symmetry related adsorbed metals (not including any oxygen group).

gp\_Pb1\_D1\_D2 or gp\_Sb1\_D1\_D2: group first (usually the only) sorbate from domain 1 (A and B subdomains) and domain 2 (A and B subdomains) together using glide plane symmetry relationship. Note that you want to group two domains together always because you want to avoid steric unfeasibly situation when including two set of sorbates within one unit cell. Therefore, you split two set of sorbates into two symmetry related domains.

### 3) Distal oxygens:

gp\_HO1\_set1\_D1: group two symmetry related distal oxygen atoms together (4 in total if considering those for the symmetry related domains), the set index is the same as that described above (set1→set3→set5→) for the sorbate, and the number after HO specify the distal oxygen, so if there are 3 distal oxygens coordinated with the sorbate, then we use \_HO1\_, \_HO2\_ and \_HO3\_ to distinguish those.

gp\_HO\_set1\_D1: group all the distal oxygens for the first sorbate (like Pb1, what set1 means here). So the set index is different from those defined above (increment of 2) in that it starts from 1 and with an increment of 1. Corresponding to gp\_sorbates\_set1\_D1, it is used to set equal u or oc for the distal oxygens associating with the symmetry related sorbate from two different domain (domainA and domainB).

gp\_HO1\_D1\_D2: refer to gp\_Pb1\_D1\_D2

### 4) Waters:

gp\_Os1\_D1: group Os1 together from domain 1 (A and B subdomains)

gp\_Os1\_D1\_D2: refer to gp\_Pb1\_D1\_D2

gp\_waters\_set1\_D1: discrete grouping for each set of water at same layer, group u, oc and dz

## 3. Fitting parameters

For each domain, there will be a user-defined parameter instance being created automatically. The parameters inside each instance dynamically change upon different binding configuration. Besides that, there are the other two par instance, i.e. inst (through which you set the scaling factor for your fitting) or rgh (through which you set the roughness of your structure model). More details are stated as below.

1) inst.set\_inten: set the overall scaling factor during model fitting

2) rgh.setBeta: set the roughness for the surface (usually <0.2)

3) rgh\_domainN.setWt:set the domain waight for domainN

### Parameters for binding in monodentate mode (MD)

It has never been tested for binding cases under MD mode. The features are not well coded, and the existing ones could be buggy. It should be accomplished in the future once we are working on a case under this mode.

### Parameters for binding in bidentate mode (BD)

4) rgh\_domainN.setAnchor\_offset\_**BD**\_n:set anchor offset (read geometry section for detail) for sorbate set n (n should be even number from 0. 0 means using this par for sorbate 0 and sorbate 1, whereas 2 means using this par for sorbate 2 and sorbate 3, and so on).

And note that the BD in the par name means setting this par under bidentate mode, so TD means tridentate mode and MD means Monodentate mode and OS means outersphere mode. Such naming rule is implicitly applied hereafter. Effect of this par will shift the sorbate and distort the local structure of the sorbate.

5) rgh\_domainN.setAngle\_offset\_**BD**\_n: set angle offset for first distal oxygen to distort the local structure.

6) rgh\_domainN.setAngle\_offset2\_**BD**\_n: set angle offset for second distal oxygen to distort the local structure.

7) rgh\_domainN.setOffset\_**BD**\_n: set offset for first distal oxygen, and it will change the sorbate-oxygen bond length.

8) rgh\_domainN.setOffset2\_**BD**\_n: set offset for second distal oxygen, and it will change the sorbate-oxygen bond length.

9) rgh\_domainN.setPhi\_**BD**\_n: set rotation angle for sorbate species including metal and the associated distal oxygens). The polyhedral will be rotated about an axis (formed by anchor atoms) by this degree. And note that the zero rotation angle is arbitrarily set so that the maximum sorbate-surface cation distance is maintained.

10) rgh\_domainN.setTop\_angle\_offset\_**BD**\_n: set a offset to the ideal top angle. For example, the ideal top angle for tetrahedral is 109.5 °. An offset to this ideal value will make a distortion to the local structure.

11) `rgl_domainN.setR(Theta, Phi)1_m_BD_n`: set the coordinates (r, theta and phi) in a spheric coordinate system for distal oxygens if under ADD\_DISTAL\_WILD mode.

Depending on the number of distal oxygens, you may have multiple r, theta and phi in the setting, so the m can be any integer number higher than 1.

Parameters for binding in tridentate mode (TD)

12) `rgl_domainN.setDr1(2 or 3)_oct_TD_n`: bond length change in unit of Å along the vector from sorbate to the associated distal oxygen in the case of octahedral local structure.

13) `rgl_domainN.setDr_tetrahedral_TD_n`: bond length change in unit of Å along the vector from sorbate to the associated distal oxygen in the case of tetrahedral local structure.

14) `rgl_domainN.setTop_angle_TD_n`: set the top angle (O-M-O) in the case of trigonal pyramid local structure.

15) `rgl_domainN.setR(Theta, Phi)1_m_TD_n`: set the coordinates (r, theta and phi) in a spheric coordinate system for distal oxygens if under ADD\_DISTAL\_WILD mode.

Depending on the number of distal oxygens, you may have multiple r, theta and phi in the setting, so the m can be any integer number higher than 1.

Parameters for binding in outersphere mode (OS)

16) `rgl_domainN.setRot_x(y,z)_OS_n`: set the rotation angle about x(y,z) axis passing through the sorbate for an octahedral structure. It can be any value in [0,360].

17) `rgl_domainN.setR0_OS_n`: set a bond length in Å for an outersphere local structure.

18) `rgl_domainN.setCt_offset_dx(dy, or dz)_OS_n`: shift the sorbate along with the distal oxygens in x, y and z direction. Note here the values are in fractional unit. So the widest range for dx, dy, dz is [-0.5, 0.5].

Parameters for group instance

Note that the fitting parameters for any one group instance include dx, dy, dz (for group atom movements in x, y and z direction), u (thermal factor) and oc (occupancy).

For example, a group instance of `gp_O1O2_O7O8_D1` could be called in the following ways to control the position, thermal factor and occupancy of group atoms:

`gp_O1O2_O7O8_D1.setdx`

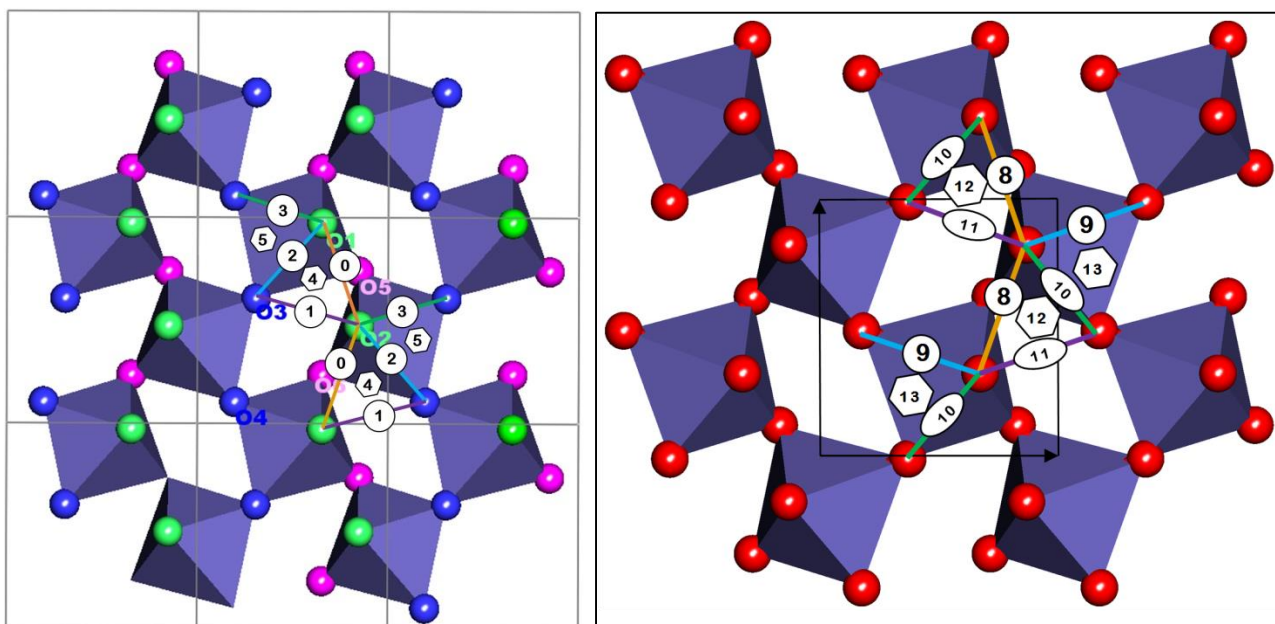
`gp_O1O2_O7O8_D1.setdy`

`gp_O1O2_O7O8_D1.setdz`

`gp_O1O2_O7O8_D1.setu`

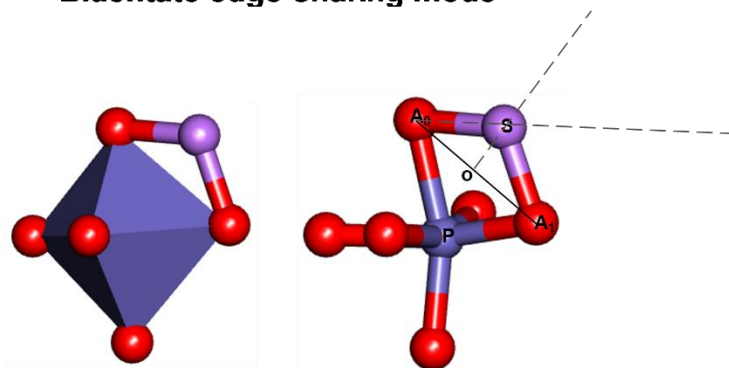
`gp_O1O2_O7O8_D1.setoc`

**Potential binding sites pairs** on half layer (left) and short full layer (right) termination surface, numbers in circle (bidentate) or pentagone (tridentate) are the associated matching index used in the script.

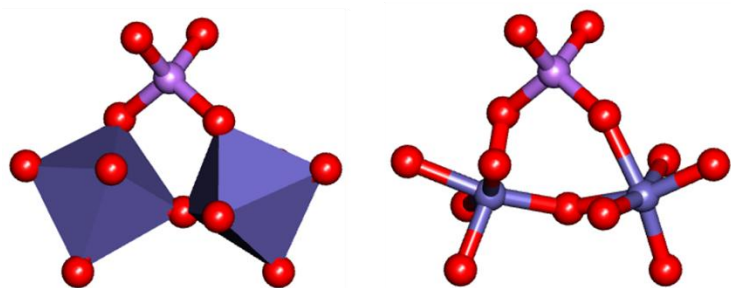


### Geometric solutions to setting sorbate on surface under different binding modes

#### **Bidentate edge-sharing mode**



#### **Bidentate corner-sharing mode**



There are two different motions the sorbate could undergo in a symmetric way (as opposed to distorted way). One is having the sorbate rotate along the anchor axis ( $A_0A_1$ ). Once the sorbate is rotated, the coordinated distal oxygens will be rotated in the exactly same way. You may need a reference point to define the zero rotation angle. By default, the zero rotation angle is obtained when the reference point (P), two anchor points ( $A_0$  and  $A_1$ ) and the sorbate are coplanar. It is obvious, at zero rotation angle, the distance between the sorbate and the reference point is maximum. Besides rotation motion, the sorbate can undergo stretching via moving the sorbate along the OS line, where O is the center point of  $A_0A_1$ .

As a result, the top angle changes ( $A_0SA_1$ ). The associated fitting parameters are  $\text{top\_angle}$  for the trigonal pyramid case and  $\text{top\_angle\_offset}$  (offset from the ideal 109.5 degree) for the tetrahedral case.

To distort the local structure, the sorbate is allowed to move along  $A_0S$  direction.

In the case of **tridentate binding mode**, three anchor points, which form an equilateral triangle, will be needed to apply the associated geometric algorithm correctly. However, three anchor points from a surface usually form a non-equilateral triangle. To overcome this issue, we need to calculate a dummy anchor point, which will form an equilateral triangle with two of the anchor points and is closest to the third anchor point. Several steps to achieve that: draw a plane perpendicular to the anchor axis formed by first two of the anchor points ( $A_0$  and  $A_1$ ); draw a circle on that plane passing through the center point (O) of the anchor axis with radius defined by half of the length of line segment formed by the first two anchors; project the third anchor ( $A_3$ ) on that plane, then connect the center of the circle to the projected point ( $A_{3,\text{prj}}$ ); the cross point is the dummy point (D) we want to find. The schematic diagram is as follows. It should be noted that with three points defined the whole local structure is defined, so we don't have any more freedom to move the sorbate coordinated structure as in the case of bidentate mode.

