# Supplementary Information: Graph theory approach to determine configurations of multidentate and high coverage adsorbates for heterogeneous catalysis

Siddharth Deshpande[1, †], Tristan Maxson[1, †], Jeffrey Greeley[1, *]

1. Davidson school of Chemical Engineering, Purdue University, 480 W Stadium Mall Drive, West Lafayette, IN, 47907

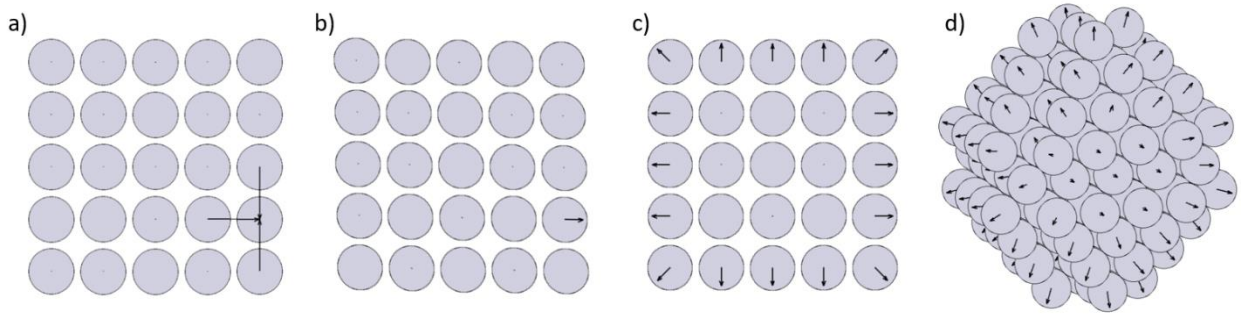
† These authors contributed equally to this work

* Corresponding author: jgreeley@purdue.edu

# A. Graph based method extra information

## Surface normal generation method

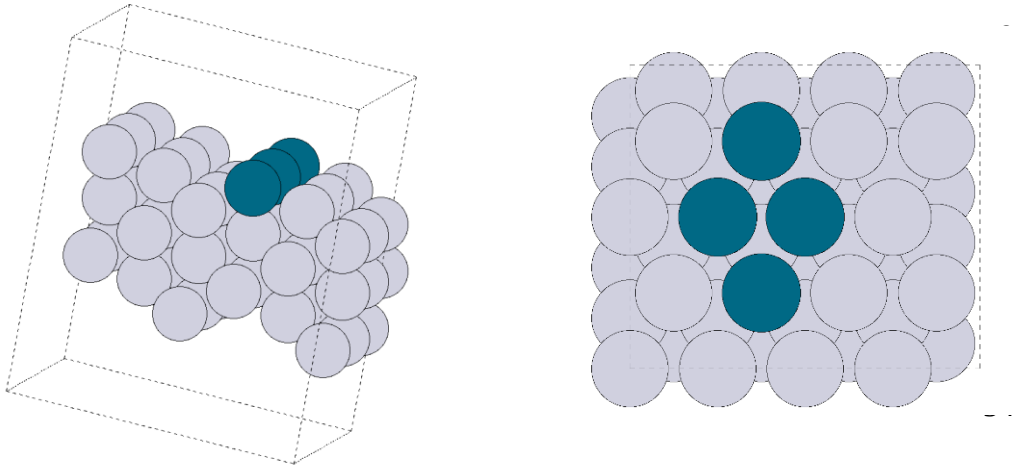
Surface normals are approximated by the bonding of the atoms surrounding the surface atom. A surface atom should be undercoordinated in the direction of vacuum, which can be determined by finding the summation of vectors from the nearest neighbors towards the atom. A tolerance is then defined to account for relaxation of the atoms and if the resulting vector's magnitude is less, the atom is not part of the surface. If it is larger, the surface normal is the normalized resulting vector. Surface normals for sites involving more than one atom can be approximated by summing the normals of the site's atoms and renormalizing. For single layer 2D systems, it isn't possible to generate normals in this manner, but this can be handled by directly assigning normals along the perpendicular axis. For periodic slab calculations, both the top and bottom surfaces will be detected as surface and one side will have to be filtered out by position or direction of the normal. Figure below shows the example for generating normal for the surface atoms in Supplementary Figure 1 a), the periphery atoms are the surface atoms and a shows all the vectors that the algorithm will detect to a given surface atom. Supplementary Figure 1 b) then shows the outward direction detected, when all the vectors to the atom are summed together. Supplementary Figure 1 c) and d) represent the normal detected for all surface atoms for a slab and than a nanoparticle case.



Supplementary Figure 1: Vectors from neighbors to a surface atom (a), normalized summation of vectors for the surface atom (b), calculated normals for all surface atoms (c), and calculated normals for a full 3D nanoparticle (d).

## Site detection method

We have chosen to detect sites for adsorption by first determining surface atoms as previously described, then finding connected groups of two, three, or four atoms. By ensuring there is a continuous cycle of bonding from each atom to the next, we find all the relevant sites for adsorption. This method does encounter potential problems with some cases such as small step edges and four coordinated hollow sites on face centered cubic surfaces. Since stepped surfaces are typically large calculations, it is possible that too thin of a surface in the direction parallel to the step edge will result in a cycle that goes across the periodic boundary condition. Similarly, four coordinated hollow sites can be detected with this method, but it is possible for two adjacent three coordinated hollow sites to also be detected instead as well see Supplementary Figure 2. These problems can be solved by either placing an adsorbate, detecting its nearest neighbors by bonding, then reassigning its coordination or by rejecting sites where all site atoms are not equidistant to the adsorbate to some tolerance. This problem can be specific to the type of system being investigated as appropriate adsorption sites for a metal surface, metal nanoparticle, or graphene sheet may have to be determined differently.



Supplementary Figure 2: Pictured left is an incorrectly detected 3-cycle along the periodic boundary condition for a fcc(211) surface, the average position of the site corresponds to a top site. Pictured right is an incorrectly detected 4-cycle for a fcc(111) surface, the average position of the site corresponds to a bridge site.

## Tuning the graph generation

Graph generation relies on a few parameters that need to be tuned for the system, namely the graph radius and the nearest neighbor calculation. The graph radius can be imagined as the number of nearest neighbor shells added to the graph. For example radius of one, includes the nearest neighbors of the active site, two includes the next nearest neighbors as well. Therefore, graph radius can be increased to encode more information into each graph, but too large of a radius may cause adsorbates in the same chemical environment to be considered different due to very small differences in atoms too far away to matter. For this reason, this parameter needs to be tuned initially for different types of systems until it produces chemical environments that are unique and have sufficient information necessary to estimate properties such as the adsorption energy as a function of the chemical environment. The nearest neighbor calculations rely on the covalent radii of the atoms as well as a small tolerance (skin factor) and both parameters can be adjusted. Since

periodic boundary conditions are relevant for surface calculations, there is also a parameter which allows the generation to expand the cell across the periodic boundary conditions which becomes important if the graph radius is similar to the number of atoms in any direction of the cell. This parameter will cost computational time for graph generation so it should be as low as possible. If the system is in vacuum in all directions, this parameter can be reduced to (0,0,0), and to consider one additional repetition in the x and y direction, it can be set to (1,1,0) (default).

### Multi-dentate Graph Generation

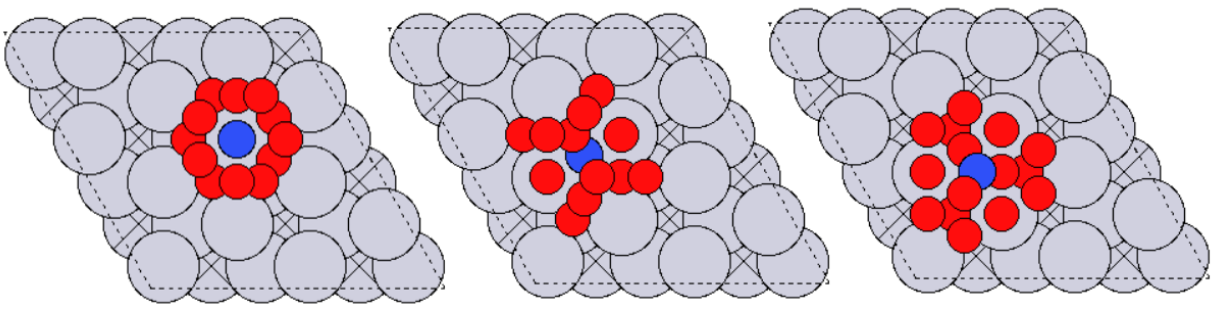
The method for single-dentate graph generation also works for multi-dentate graph generation without modification. This is possible because of differing bond weights when finding the ego graph of the full graph for an adsorbate. We define adsorbate-adsorbate bonds to have a weight of 0, which allows for entire adsorbates to be found when any atom of it is found in the graph. This also ensures, that when multiple adsorbates are present, it is easy to distinguish the number of adsorbates present in the model. Adsorbate-surface bonds have a weight of 0.5 and surface-surface bonds are given a weight of 1, which simply means that the ego distance within the surface is directly related to number of coordination shells or the radius. Assigning different weights to bonds is important to ensure that the algorithm correctly identifying all interactions. These properties can be tuned to get different graph generation behavior and add further capabilities such as detecting hydrogen bonds etc.

## B. Evolutionary Algorithm and high coverage phase diagram

### Brute force comparison on Pt(111) and $Pt_3Sn$(111)

Our method is largely motivated by the difficulty of brute force solutions for systems of high coverage. In manually trying to estimate high coverage configurations, it is possible to miss

important configurations and this task is difficult or impossible for a researcher to do by hand. To exhaustively calculate all configurations of NO on the Pt(111) and $Pt_3Sn$(111) systems, we used a brute force approach for filling all sites as long as the site was not within 2.4Å of another adsorbate, this prevents sites that are too close from being filled partially reducing the complexity of the problem based on physical constraints. The generated sites were then compared to each other using our graph-based method by considering a large radius of 5 shells to account for the entire surface. Further, the effects of the distance constraint is shown in Supplementary Figure 1 for the unique sites on the Pt surface. The blue atom indicates a pre-occupied site and the red atoms show the sites that were invalidated. This constraint was also used in the evolutionary-based approach. While the Pt(111) system can be brute forced for a single adsorbate with current hardware for a small cell, the case of $Pt_3Sn$(111) grows in complexity much faster and requires a larger cell to model. Also, the introduction of defects or mixed adsorbates could make it unreasonable to brute force, motivating our proposed evolutionary approach to this problem. The number of possible configurations calculated by brute force and reduced by symmetry is presented in Supplementary Table 1 showing the complexity for a root 12 cell. Pt(111) and $Pt_3Sn$(111) was only explored to the upper limits of what we expect would be considered reasonable on current hardware to calculate at the DFT level.



Supplementary Figure 3: The sites invalidated by adsorption based on distance for a Pt surface with a distance of 2.4 Å. Top, bridge, and hollow sites are colored from left to right in blue and the invalidated sites are colored red.

| Surface coverage | Pt(111) | $Pt_3Sn$(111) | $Pt_3Sn$(111) (No Sn sites) | $Pt_3Sn$(111) (Evolutionary) |
|---|---|---|---|---|
| 1/12 | 4 | 9 | 5 | 5 |
| 2/12 | 43 | 140 | 34 | 34 |
| 3/12 | 499 | 1871 | 199 | 137 |
| 4/12 | 3981 | <30000 | 674 | 266 |
| 5/12 | <50000 | | 1320 | 86 |
| 6/12 | | | 1269 | 25 |

Supplementary Table 1: The number of configurations generated (reduced by comparing graphs) for each system of Pt or $Pt_3Sn$. The different $Pt_3Sn$ numbers arise from different simplifications.

## Evolutionary algorithm

At a low coverage (one or two adsorbates), the evolutionary algorithm performs in a similar manner as brute force filling of sites except for symmetry detection via atomic graphs. All sites are filled with the adsorbate, duplicates are removed by isomorphism, then the structures are computed. The next step is to remove duplicates from the computed structures as some structures may relax to the same state, identifying the lowest energy structures for a given case. These structures are then all filled again in a similar way, resulting in a slightly higher coverage. Once the number of calculations becomes too high to efficiently run, only the best candidates are run further using an energy cutoff of 0.25 eV from the lowest energy structure at that coverage. This

greatly reduces the number of required calculations. We found that this method finally generated nearly identical structures that the brute force method did for the $Pt_3Sn$ system. This cutoff is picked to allow for enough sampling of the problem space. For a given case, this cutoff needs to be tuned by keeping in mind that a lower energy cutoff would possibly miss some minor improvements, but larger numbers would cost more computational hours.

Further, we think that this algorithm broadly has two tuning knobs, which can be tuned to generate configurations for any system:

1. The number of the nearest neighbors considered: Currently we use a radius of 2, which means we are considering interactions two shells out from the adsorbate. For less strongly bound adsorbate, this number can be increased to get more unique configurations.
2. The energy cutoff for structures to consider for the next generation: This number decides what is the range of structure's energies (relative to the lowest energy structure) that should be considered to populate the next generation. If the interaction of the adsorbate with the surface is strong, then a small energy cutoff (eg. 0.25 eV considered in this case) should be sufficient to capture the complexity of the system, but, if there are other important interactions, this number can be increased to account for a larger variety of structures to be populated.

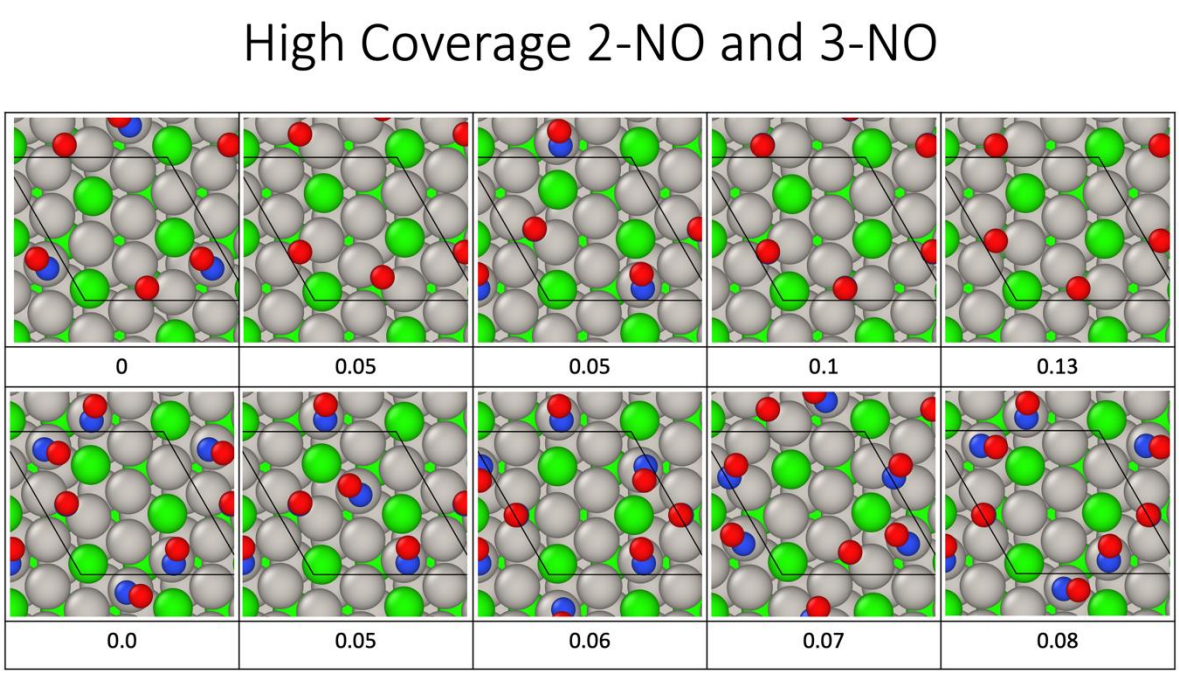
In general, these two tuning knobs can be used to maintain a balance between the computational expense and the exhaustive search for all possible configurations. If a large radius is considered, and all configurations are taken to generate the new generation, then the sample space will be completely explored (we note that a similar approach was used in our brute force method).

It may be possible for some systems that this straightforward approach would have problems when high coverages have strong adsorbate-adsorbate interactions which do not appear at low coverages. In this case, more structures may be found by working from high to low coverage

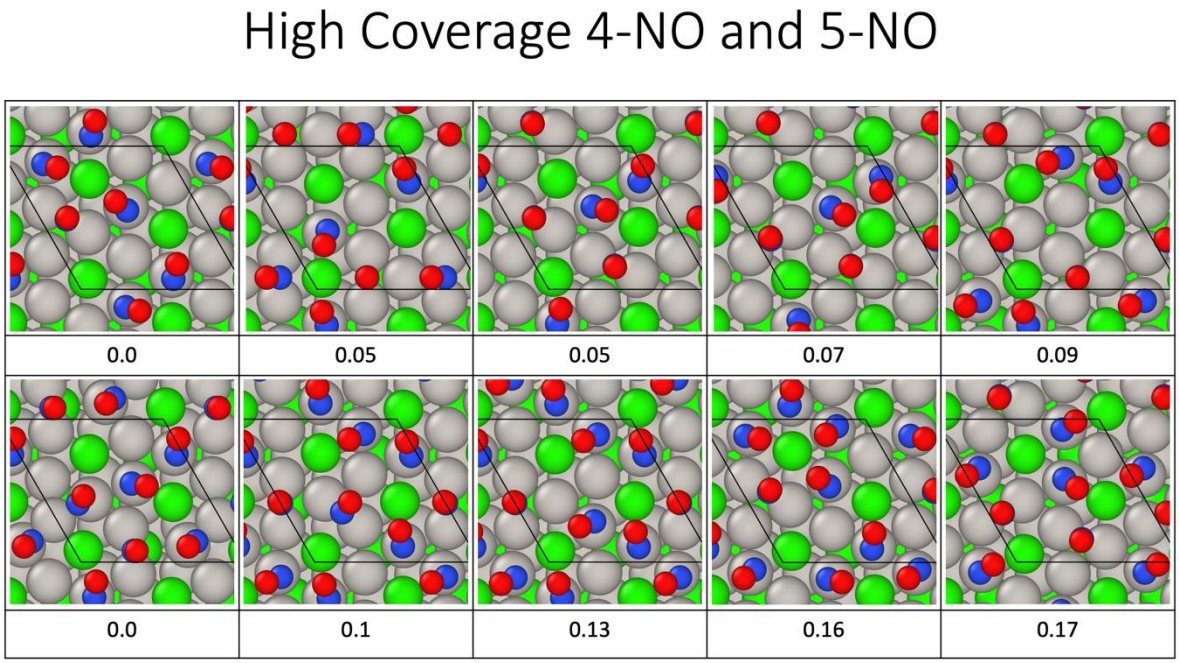
as well. By removing adsorbates from known configurations or performing rearrangements on lowest energy structures, we propose that more relevant structures may be found. It is also possible to determine an appropriate energy cutoff automatically by looking for the energy cutoff that starts to produce a bimodal distribution of energies, showing that a new type of unfavorable interaction has been found for some configurations.

## Important structures for evolutionary algorithm

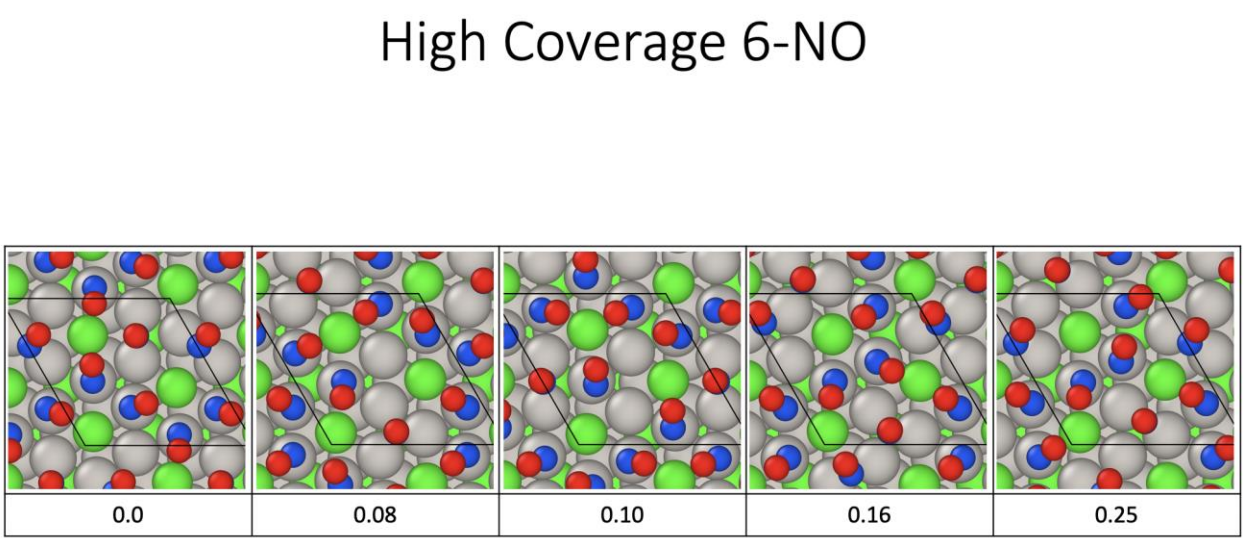
We present structures for some of the most stable configurations that we find for the different coverages of NO. The structure, along with their energies relative to the most stable structure is shown in the figures below (Supplementary Figure 3, 4 and 5).




Supplementary Figure 4: Selected high coverage configurations for the 2/9 and 3/9 ML of NO




Supplementary Figure 5: Selected high coverage configurations for the 4/9 and 5/9 ML of NO




Supplementary Figure 6: Selected high coverage configurations for the 6/9 ML of NO.

## Coverage dependent phase diagram

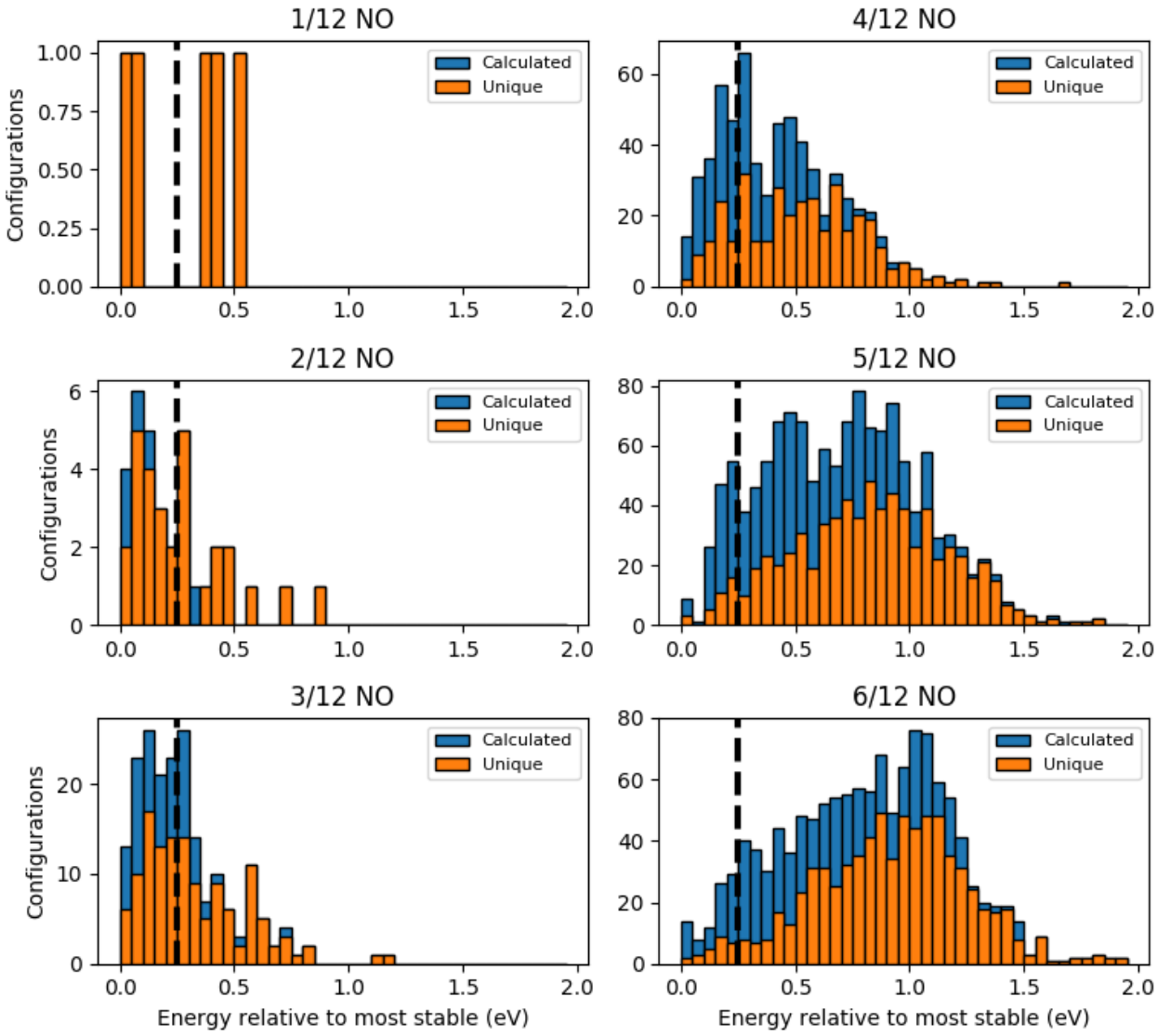
To calculate the coverage dependent phase diagram, the most stable configurations for a given coverage were first calculated using the evolutionary algorithm approach. Then, the free energy of a given configuration was plotted as a function of the partial pressure of NO, using the formulae:

$$\Delta G = \frac{1}{9} \times [E_{n \times NO^*} - E_* + n \times (ZPE_{NO^*} - TS_{NO^*} - \mu_{NO(g)})]$$

$$\mu_{NO(g)} = E_{NO(g)} + ZPE_{NO(g)} - TS_{NO(g)} + kT \times \ln(\frac{P}{P^0})$$

where, $E_{nNO^*}$ and $E_*$ represent the DFT calculated energies for the most stable structure containing 'n' NO atoms on the surface, and the clean surface respectively. ZPE(NO*), $TS_{NO^*}$ and $\mu_{NO(g)}$ represent the zero point energy and the entropy change of the adsorbed NO on the surface, and the chemical potential of gas phase NO respectively. $E_{NO(g)}, ZPE_{NO(g)}, TS_{NO(g)}$ and $kT \times \ln(\frac{P}{P^0})$ represent the DFT calculated energy of a single NO molecule, the entropy of gas phase NO using standard tables and the pressure correction respectively. The entropy of adsorbed NO is taken to be zero in this case as it binds strongly to the surface. The prefactor of '1/9' is included to account for area normalization w.r.t. to the total number of available surface atoms, and in this case the number is calculated given that only 'Pt' atoms bind NO.

## Energy histograms




Supplementary Figure 6: Histograms plotting the energetics of all the considered structures in the brute force approach.

Shown above are the histograms containing all the simulations performed for the coverage dependent phase diagram using the brute force approach. The blue bars show all the cases, while the orange ones show the unique cases detected using the graph-based algorithm post simulation. Also shown in black vertical line is the number of structures that will be considered, if a 0.25 eV cutoff is considered as a selection criterion for the evolutionary algorithm.