Journal of Physics: Condensed Matter

Structure prediction of surface reconstructions by deep reinforcement learning

Structure prediction of surface reconstructions by deep reinforcement learning

Søren A. Meldgaard, Henrik L. Mortensen, Mathias S. Jørgensen, and Bjørk Hammer*
Department of Physics and Astronomy, Aarhus University, DK-8000 Aarhus C, Denmark.

(Dated: May 5, 2020)

We demonstrate how image recognition and reinforcement learning combined may be used to determine the atomistic structure of reconstructed crystalline surfaces. A deep neural network represents a reinforcement learning agent that obtains training rewards by interacting with an environment. The environment contains a quantum mechanical potential energy evaluator in the form of a density functional theory program. The agent handles the 3D atomistic structure as a series of stacked 2D images and outputs the next atom type to place and the atomic site to occupy. Agents are seen to require $1\,000-10\,000$ single point DFT evaluations, to learn by themselves how to build the optimal surface reconstructions of anatase $TiO_2(001)$-(1×4) and rutile $SnO_2(110)$-(4×1).

INTRODUCTION

Machine learning (ML) methods are currently impacting many aspects of modern society. ML is appearing in commodity products making user interaction smarter (handwritten text and voice recognition, natural language processing, facial recognition, automated tagging on social media, etc.), in health care assisting diagnosis formulation, and in contemporary science research involving large data sets. In the latter case, the ML methods do for instance allow for extraction of models that need not be formulated on human-understandable grounds. Depending on the setting, such models may be useful in steering the research efforts, e.g. by suggesting new chemical compounds [1], or by identifying rare events and odd observations. Especially one ML discipline, *computer vision* with deep neural networks is showing remarkable success in a wide range of applications by making intelligent decisions based on data. Examples include cancer prediction from MRI scans [2], image generation [3–5] and autonomous vehicles [6] among many others.

In the domains of computational materials science and chemistry, reports on use of computer vision is still scarce [7–9]. Many other types of ML methods have, however, recently proven remarkably successful in improving computational capabilities in these domains [10, 11]. Accomplishments include predictive models of potential energy surfaces [12–16], accelerated nudged elastic band calculations [17–19], enhanced sampling methods [20, 21] and prediction of physical properties [22–25]. However, while ML enables fast predictions of properties, the task of discovering the correct structure to evaluate remains. A recurring problem is structure determination from experimental information such as scanning tunneling microscopy and low-energy electron diffraction. Considerable research has gone into developing automated search algorithms relying on computational chemistry methods to predict the structure. Widely used methods include evolutionary algorithms [26–31], simulated annealing [32] and random structure search [33], among others. See [34] for a review of structure prediction. Despite the automated nature of the algorithms, finding the optimal structure is still a time-consuming process due to costly computational chemistry methods such as density functional theory (DFT) and the immense configurational and compositional space. To overcome the former issue, several successful attempts of modeling the potential energy surface during a structure search [35–41] have proven to severely reduce the computational load of DFT calculations. Nevertheless a brute-force approach is still intractable due to the size of chemical space.

To deal with the complexity problem efficiently, efforts have been put into changing the inherently stochastic nature of the aforementioned structure prediction algorithms, to algorithms that make informed choices [42–46]. Such methods are capable of traversing the configurational and compositional spaces more efficiently, thereby opening up for tackling more complex problems. Reinforcement learning [47], a subfield of machine learning, describes precisely such a framework where a decision maker (an agent) performs intelligent choices based on automated interactions with an environment to solve an optimization problem. Recent examples include control problems such as games [48, 49] and fluid dynamics [50] where the machine learning model learns how to optimize an objective function by interacting with the environment. In materials science and chemistry the method of reinforcement learning has been used in molecular generation [51–53] and retrosynthetic planning [54].

Recently, we proposed a ML method combining computer vision and reinforcement learning, the atomistic structure learning algorithm (ASLA) [9]. It enables structure prediction of 2D materials and molecules without any prior training or knowledge. A neural network 'sees' a 2D atomistic structure and learns autonomously to build sound and stable chemical compounds while interacting only with a first principles quantum mechanical energy calculator. ASLA requires only positions and atom type information, hence no auxiliary data such as bond-types or domain-specific information is necessary, making ASLA transferable to different supercells and stoichiometries. Unlike earlier work we are thus not limited to studying molecules [51–53] but may study periodic

* hammer@phys.au.dk

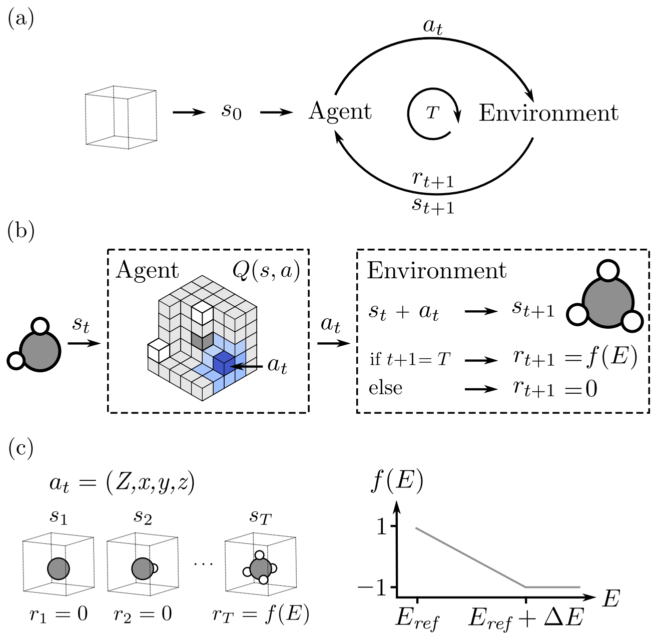

Figure 1. (a) A representation of an empty cell is fed to an agent tasked with predicting the next action. The environment responds with a new state and a reward and then prompts the agent for the next action. (b) Illustration of one step of the building process. The agent predicts Q-values for possible locations and feeds the action with the highest Q-value to the environment. The environment constructs the new structure and calculates a reward. (c) The action corresponds to atom type and location. Rewards are zero for intermediate states, whereas in the final state a reward based on the energy of the completed structure is given. Here E_{ref} is the energy of the best structure observed and $\Delta E > 0$ is a hyperparameter.

structures. Finally we note that the energy calculator can be substituted with a ML-model, making ASLA compatible with methods that reduce the computational load of DFT calculations.

In this work we extend the ASLA method to 3D structures by incorporating additional configurational information in the representation of the structure. In the first section we recap the ASLA algorithm and introduce the 3D version. We then demonstrate the capabilities of the method by studying the anatase $TiO_2(001)$-(1×4) reconstruction [55], where we incorporate experimental information. Finally we study the elusive structure of rutile $SnO_2(110)$-(4×1) which remained unsolved for more than three decades before it was recently found [56].

METHOD

Reinforcement learning attempts to solve decision processes where an agent is interacting with an uncontrollable environment. Specifically, an agent is given a *state* and must decide on an *action*. The environment responds with a new state and a *reward*. The process then repeats itself and the goal of the agent is to accumulate as much reward as possible. In the episodic case the process ends after a number of interactions and the agent may then device a better strategy for the next episode.

ASLA depicted in figure 1, combines reinforcement learning and structure prediction by formulating the energy optimization problem as a decision process. Specifically, an agent is given a representation of an incomplete structure and decides the location and type of the next atom to place. The environment responds with a new state representing the structure with the added atom, and a reward to be described later. To model the agent interacting with the environment, time is discretized such that the agent performs action a_t, at time t, and ends up in state s_{t+1} with reward r_{t+1} in the following time step. Interactions between the agent and environment end at time step T, which is specified by the user and determines the number of atoms the agent is to place. The reward is chosen as zero for intermediate actions whereas the final reward is given by

$$r_T = f(E_{DFT}(s_T)) \qquad (1)$$

$$f(E) = \max\left(\frac{2}{\Delta E}(E_{ref} - E) + 1, -1\right), \qquad (2)$$

where E_{DFT} is the energy of the structure and E_{ref} is the lowest energy observed in the search. All structures with energies higher than $E_{ref} + \Delta E$ are given a reward of -1 to focus the reward resolution on structures in an interval of ΔE. To maximize the sum of rewards, a policy, $\pi(a|s)$, determines the probability of each action given a state, such that the expected cumulative reward when following policy π is maximized. We choose to model the policy implicitly through a Q-function given by

$$Q_\pi(s,a) = \mathbb{E}_\pi\left[\sum_{k=t}^{T-1} r_{k+1} \middle| s_t = s, a_t = a\right] \qquad (3)$$

$$= \mathbb{E}_\pi[r_T | s_t = s, a_t = a], \qquad (4)$$

where we used the fact that intermediate rewards are zero. The Q-function determines the expected cumulative reward when taking action a in state s, and then following policy π. In the case of a deterministic policy and environment the expectation can be calculated exactly using a single sample. To compute the optimal policy one must find the Q-function that satisfies

$$Q_*(s,a) = \max_\pi Q_\pi(s,a), \qquad (5)$$

i.e. the Q-function for the policy with the highest expected return. Instead of computing $Q_\pi(s,a)$ for all policies we iteratively refine a policy implicitly defined as a function approximator of the Q-values. Specifically we infer a greedy deterministic policy by

$$\pi(a_i|s) = \begin{cases} 1, & \text{if } a_i = \text{argmax}_a\, Q_\pi(s,a) \\ 0, & \text{else} \end{cases} \qquad (6)$$

which always chooses the action that maximizes the expected reward. The initial function approximation will result in a suboptimal policy as no training data is present yet, but allows for evaluations of state-action pairs that have never been sampled. To improve the policy, the agent interacts with the environment, receives new rewards, and subsequently updates the function approximator which iteratively improves the policy. Actions chosen during interactions with the environment are based on estimated Q-values from the function approximator. Poor actions will therefore progressively be ruled out as training data accumulates, which tremendously reduces the number of policies to attempt. However, to stimulate exploration, a modified epsilon-greedy strategy is used, where now the best move according to the function approximator is chosen only with probability $1 - \varepsilon$. Here $\varepsilon = \varepsilon_0 + \varepsilon_\nu$, where ε_ν is the probability of the actions being taken from the $\nu = 2\%$ best actions and ε_0 is the probability of the actions being picked completely randomly. Unlike a standard epsilon-greedy strategy where $\varepsilon = \varepsilon_0$, we have added ε_ν to avoid oversampling the many poor actions in the large action space. Additionally, every fifth episode is done with a greedy policy to ensure promising structures are explored.

To avoid convergence issues in the DFT program, actions with atoms closer than 0.5 times the sum of the involved covalent radii are prohibited. We note that an alternative to this approach could have been to rewrite the DFT program to simply return a large energy in the event of convergence problems and have ASLA learn from that not to place atoms too closely.

For the Q-function approximator we choose the target

$$Q_{target}(s,a) = \max_{i \in X(s,a)} r_T^{(i)}, \tag{7}$$

where X is the set of episode indices of all observed structures involving the state-action pair (s,a) somewhere along their build sequences. The max-operation assures that the agent trains towards the best reward observed for any recurring state-action pair. The method allows for swift change of strategy whenever a favorable observation is made. However, it also means that new observations must be increasingly favorable to become competitive as the agent learns to refine a certain type of atomistic structure, which might represent a local minimum energy structure. To prevent stagnation in such local minima we penalize structures that are built more than ten times by modifying the energy

$$\tilde{E}^{(i)} = E^{(i)} + \sum_m A \exp\left(-\frac{|s_T^{(i)} - s_T^{(m)}|^2}{2\sigma^2}\right), \tag{8}$$

where m runs over the structures to be punished and A and σ are constants, $A > 0$. The distance between structures is evaluated using the Bag of Bonds descriptor [57]. To represent the Q-function a convolutional neural network (CNN) identical to that used in previous work [9] is used. The network contains three hidden layers. Each layer consists of 10 channels with leaky relu activations [58] and a final layer with a tanh activation. The input representation will be covered in the next paragraph. The updates are done following each episode using TensorFlow [59] with mini batch gradient descent [60] using the adam optimizer [61] on the mean squared error cost function with L2 regularization [60]:

$$J = \frac{1}{N_{batch}} \sum_{i=1}^{N_b} ||Q_{\pi,i} - Q_{target,i}||^2 + \beta||\theta||^2, \tag{9}$$

where N_{batch} is the batch size, θ are the network parameters and β is a parameter controlling the degree of regularization. The batch always contains the best structure, the newest structure and the rest are sampled uniformly from previously built structures.

Having recapped the original ASLA algorithm we now describe changes to the representation and network architecture. To present the structure as an input to the neural network a choice of representation must be made. In the 2D version of ASLA we chose a one-hot encoding of the atom positions on a 2D grid with a third dimension (channels) accounting for atom type. Specifically, a matrix of 1's at atom positions and 0's everywhere else are made for each atom type. The representation is the 3D tensor made by stacking the matrices. In this work we extend the channels to carry compositional *and* configurational information by one-hot encoding the third atom position coordinate, *the third dimension*, in the channels. An atom type is allowed to adopt for the coordinate in this third dimension one of several values, each represented as its own one-hot encoded channel. The CNN acts on this representation to produce Q-values with minimal extra computational resources compared to the 2D regime. For improved generalization of the CNN it is beneficial to obey the Hamiltonian symmetries; namely translational, rotational and mirror *invariance* of the energy. Consequently the Q-values must be *equivariant* with respect to the aforementioned transformations. The translational equivariance is given from the network architecture while rotational and mirror equivariance are learned by data augmentation.

In the next two sections we give examples of its use. First, for a rather simple case (TiO_2), where atoms are expected from experimental information to be on one of two planes, making the problem simple to illustrate and discuss. Next, for a more complex situation (SnO_2), where atoms may in principle attain any position in the third dimension, but where we show how a pragmatic approach involving a few layers in the third dimension suffices to solve the problem.

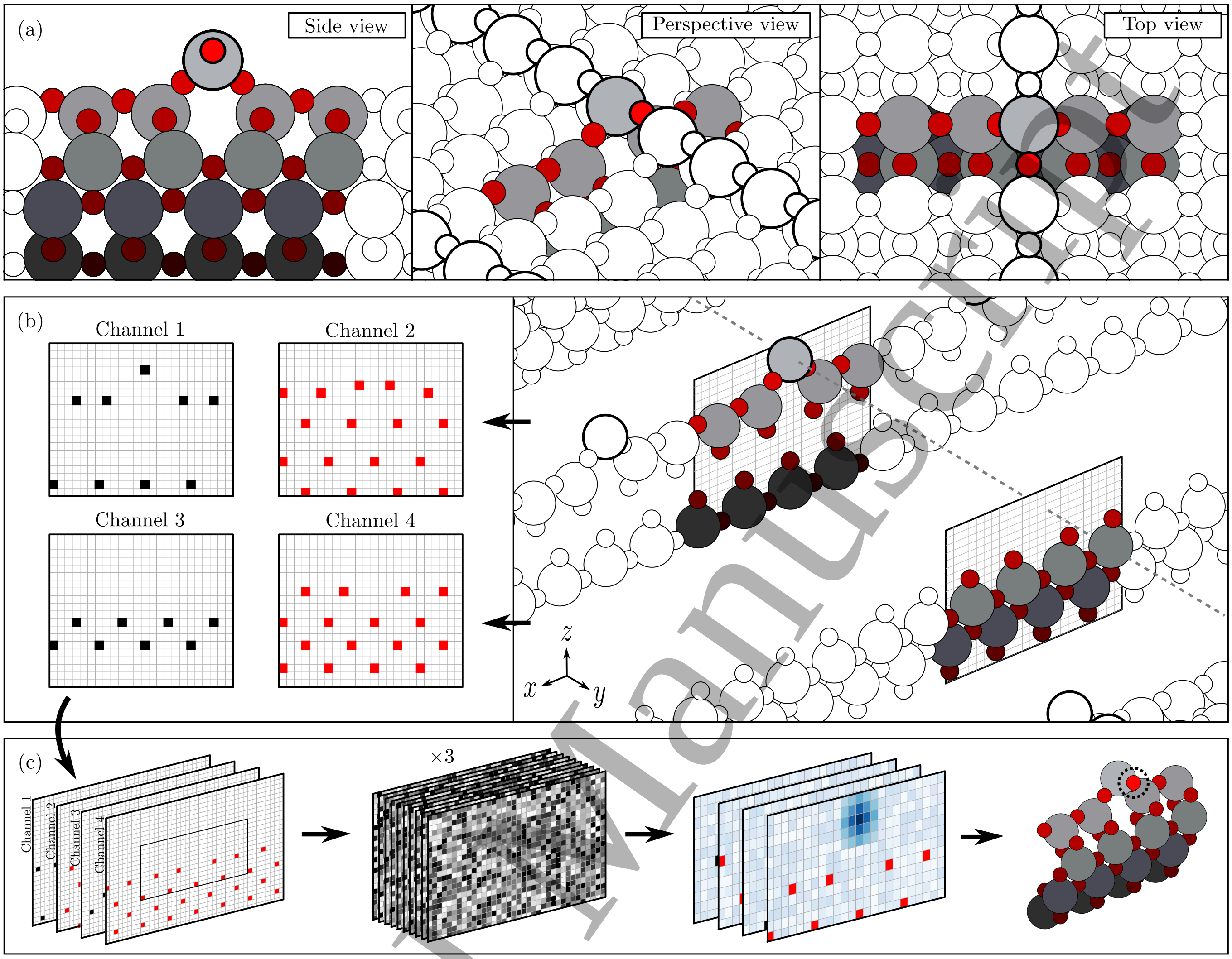

Figure 2. Anatase $TiO_2(001)$-(1×4) reconstruction. (a) Structure of the TiO_2 reconstruction. Color darkness scales with distance into bulk. (b) Illustration of the one-hot representation made from the unit cell with one oxygen atom missing. The representation is made by concatenating the four channels. (c) Schematic of predicting the Q-values for the final action. Note that the representation shrinks during the convolutional operations in the neural network, such that atoms are only placed within the predefined black box.

TiO_2 RECONSTRUCTION

As a first example the anatase $TiO_2(001)$-(1×4) reconstruction [55] shown in figure 2a is attempted. For this system, scanning tunneling microscopy experiments reveal four mirror-planes perpendicular to the surface. Two of them are xz-planes, which combined with the short length (L_y) of the cell in the y-direction leads us to restrict the y-positions of any atom to $y = 0$ or $y = L_y/2$. The xz-planes are hence treated as original ASLA pixel layers and the y-direction becomes *the third direction* to be treated similar to a color channel. Effectively, two xz-grids are introduced, each with two channels (one for each atom type), coinciding with the bulk positions of TiO_2 anatase as seen in figure 2b. A fully convolutional neural network (figure 2c) then predicts Q-values for each grid point within a predefined box. The s_0 state, *the template*, used in training the ASLA agents is a slab of three anatase $TiO_2(001)$-(1×4) bulk layers with a total of 12 Ti and 24 O atoms. All s_T states have had the agent direct the environment to place an extra 5 Ti atoms and 10 O atoms resulting in structures with 17 Ti and 34 O atoms for which the total energy calculations under periodic boundary conditions can be performed. To save on the computational demand, we employ Density Functional based Tight Binding (DFTB) calculations that are faster than full DFT calculations. For DFTB we use parameters from ref. [62], which are known to reproduce the correct global minimum [46]. ASLA hyperparameters are given in section Computational information.

An agent having trained itself on this task for 100, 400, and 1100 episodes is inspected in figure 3. The

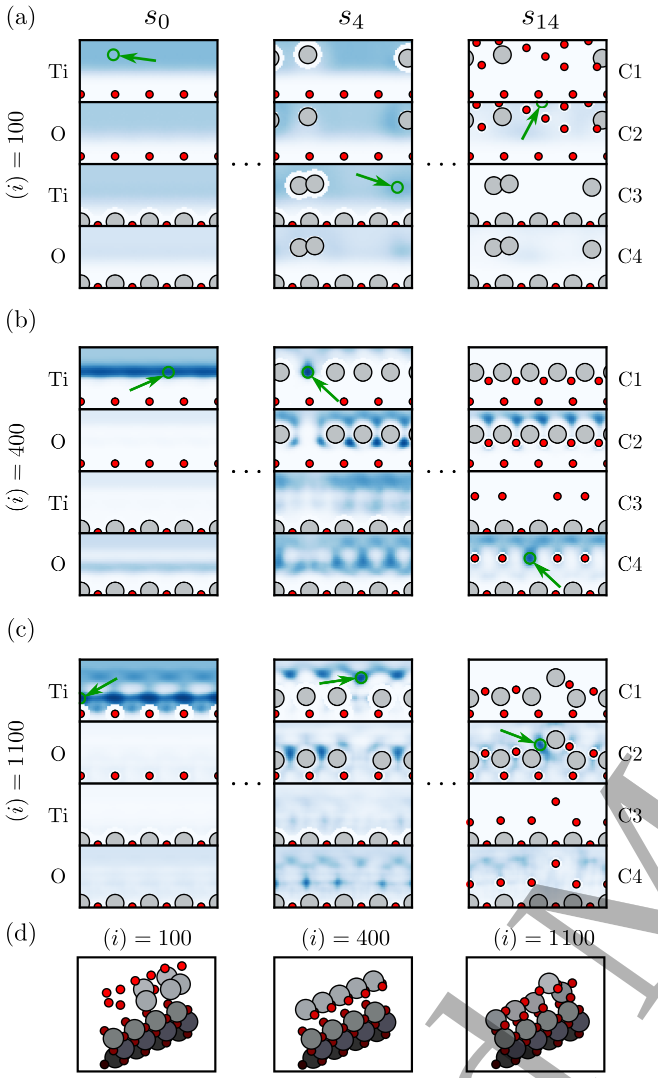
(a)
s_0
s_4
s_{14}
Ti
O
Ti
O
C1
C2
C3
C4
$\langle i \rangle = 100$
(b)
$\langle i \rangle = 400$
(c)
$\langle i \rangle = 1100$
(d)
$\langle i \rangle = 100$
$\langle i \rangle = 400$
$\langle i \rangle = 1100$

Figure 3. Q-values of an agent. To guide the eye we display all atoms for a given y-value, even though only one type is present in each channel. (a) The agent places atoms far from the bulk positions, but otherwise randomly. (b) A layer of TiO_3 molecules is placed far from the bulk positions. (c) fivefold coordinated Ti atoms are placed in a layer close to the bulk positions. An added row of TiO_3 molecules replaces a bridging O. (d) Final structure, s_{15}, in episode 100, 400 and 1100 respectively.

figure shows the Q-values that the neural network predicts when the input is either the bare template, s_0, or one of the two states, s_4 and s_{14}, that result from the greedy policy at those training episodes. After 100 episodes (figure 3a), the agent places atoms far from the bulk positions, but is otherwise ignorant of any chemical rules. At 400 episodes (figure 3b), it builds the suboptimal solution of placing all titanium atoms in a row. All titanium atoms bind to four oxygen atoms and there is minimal bonding to the bulk layers. Eventually the global minimum is found after 1100 episodes (figure 3c), where all but one titanium atom bind to five oxygen atoms, with the left over titanium recreating the motif found after 400 episodes but now added above the other atoms. This is the well-established ad-molecule model of the global minimum energy reconstruction of anatase TiO_2 first suggested by Lazzeri and Selloni in 2001 [55], but now recovered by a deep neural network-based agent, that attains sufficient chemical insight after 1100 self-guided reinforcements steps each involving a single point DFTB calculation. The role of the agent has been to navigate the configurational space to assemble the oxygen and titanium in the proper way. Since the agent always queries the full DFTB energy landscape there is no energy-uncertainty in the evaluated structure. Finally the structure is relaxed without the grid revealing exact overlap with the previously suggested structure by Lazzeri and Selloni [55].

TRANSFER PROTOCOL FOR THE SnO_2 RECONSTRUCTION

In the above example for TiO_2, the search for optimal structure was limited to atoms being in one of two xz-planes. We now turn to a harder problem, reduced rutile $SnO_2(110)$-(4×1), for which the experimental information is less and the cell dimensions larger, meaning that both in-plane directions must be considered with many grid points. We hence choose the direction perpendicular to the surface, the z-direction, as the one to be treated as *the third dimension*, cf. the method section. A small, but sufficient number of xy grid-planes must be chosen for the Sn and O atoms before commencing the search. While deciding on these planes, we shall be considering the simpler problem of the surface termination of stoichiometric rutile $SnO_2(110)$-(1×1).

The stoichiometric $SnO_2(110)$-(1×1) system is illustrated in figure 4. A one layer slab template constituting the s_0 state is shown with the white atoms. As indicated, one and three xy planes are required to accommodate the two Sn and four O atoms at their static bulk positions in the template. For the two Sn and four O atoms that ASLA places on top of the template, we need at least the same grid planes, shifted $\Delta z = 3.45$ Å (the distance between bulk layers), in order to be able to find the unreconstructed bulk truncated crystal as a possible surface structure. However, in order to allow for surface relaxation we add two additional O layers between the shifted O bulk heights. For Sn we add channels mid-way between the shifted Sn bulk heights and the closest O layer. ASLA is then tasked with placing four O atoms and two Sn atoms and the computational run is repeated 50 times from a random initialization of the network. As no Sn-O DFTB parameters are available, a full DFT description is used and described in section Computational information, where the ASLA hyperparameters are also given. For this stoichiometric (1×1) cell, the global minimum energy structure turns out to be the bulk ter-

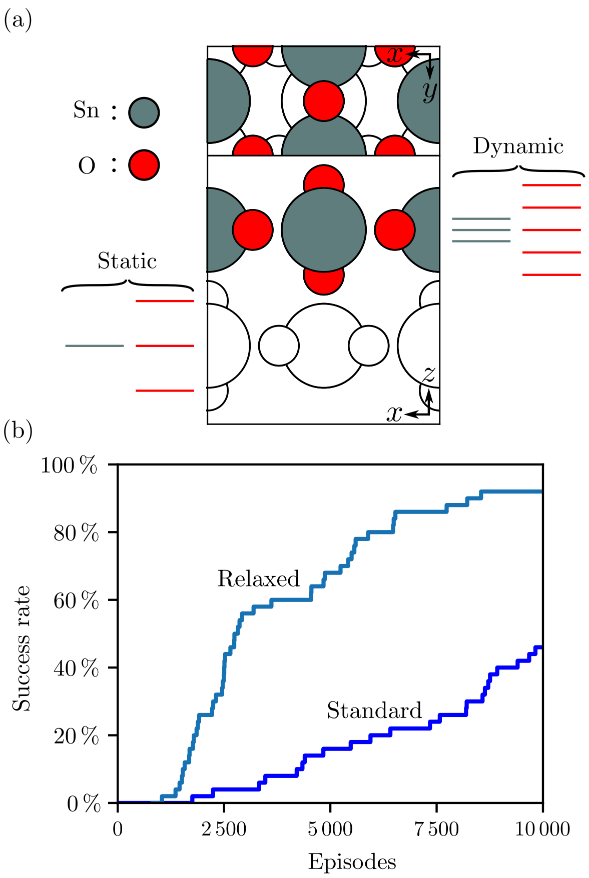

Figure 4. (a) ASLA setup for stoichiometric rutile SnO_2(110)-(1 × 1) cell. The bulk layers (static) serve only as input channels to the CNN whereas the dynamic channels can accommodate new atoms. (b) Success curve for the bulk terminated surface. 'Standard' reflects a normal ASLA run whereas 'relaxed' perform a full relaxation when a structure is penalized. An average of 7 388 DFT calculations were done in in the 10 000 episodes as the agents occasionally build an already known structure.

minated surface with the success curve seen in figure 4b. The 'standard' curve shows the success rate if only the exact global minimum structure on the grid is accepted. Alternative, one can use DFT to relax the structures that ASLA penalized, equation 8, which gives rise to the 'relaxed' curve. Here it is seen that in many runs, the agents build structures that would relax into the global minimum energy structure at an early stage of the search, where the agents have not yet learned to place the atoms perfectly. However, since the relaxations come at a large computational expense they are not sufficiently rewarding to become part of the ASLA method and will not be considered in the remainder of this paper. Instead, in the searches below we record a run as having successfully identified the global minimum energy structure as soon as the energy is less than 0.05 eV higher than the global minimum energy (evaluated *per atom* placed by ASLA).

We now turn to the SnO_2(110)-(4 × 1) reconstruction on reduced rutile SnO_2 which has attracted considerable research interest [63–68] and for which the atomistic structure remained elusive until recently where it was predicted using a combination of first-principles search methods and experimental information [56]. Here we demonstrate ASLAs capabilities by predicting the structure starting from the bulk terminated (110) surface using the grid heights introduced for solving the surface structure for the stoichiometric (1 × 1) cell. Experimentally the surface is observed to release oxygen upon heating, with sputtering and annealing further reducing the surface. Consequently a reduced stoichiometry will be used.

The problem is divided into steps where an agent is progressively trained on larger and larger systems that become computationally more and more demanding. This is possibly because agents are represented by fully convolutional networks meaning that they take an input of *any size* and output a Q-value map of the *same size*. Consequently the same agent can be used with any supercell and thus be trained initially on small problems and transferred subsequently to larger problems for further training. For the SnO_2(110)-(4 × 1) surface problem, the chosen transfer protocol is illustrated in figure 5a, where the final structure is the recently found reconstruction.

The protocol is as follows: i) an agent is tasked with solving the problem of placing Sn_2O_2 in a (1 × 1) cell, ii) the state of the agent, that built the lowest energy structure is subsequently used as starting guess for an agent that is further trained while tasked with placing Sn_3O_3 in a (2 × 1) cell, iii) finally, the state of the agent that built the best structure is transferred as a starting guess for an agent faced with the full problem of placing Sn_6O_6 in a (4 × 1) cell. To obtain statistics, the protocol is followed in 30 independent restarts.

To validate the approach we plot in figure 5b the combined success curves over the 30 restarts for each step in the protocol. The (1 × 1) task requires the most episodes before the first global minimum is found reflecting the chemical ignorance of the initial agent. Interestingly, the final success is still the highest with 19/30. For the (2×1) system slightly fewer episodes are required for the global minimum and a slightly lower final success of 17/30 is seen. Finally the (4 × 1) problem requires the least attempts before the global minimum energy structure is found in some of the runs. Here the success rate ends at 15/30 after 10000 episodes. In contrast, a search without transfer knowledge discovers the global minimum in only 2 out of 30 runs. One can hypothesize that the transferred agents gain an offset as unfavorable actions are ruled out based on the transferred knowledge learned on the smaller problems. The smaller slopes of the success curves for the (2×1) and (4×1) problems, however, reveal that despite the transferred knowledge, the complexities of these problems persist to represent considerable challenges.

Performing for the ever-best determined structure in each of the runs for the (4 × 1) problem a full DFT

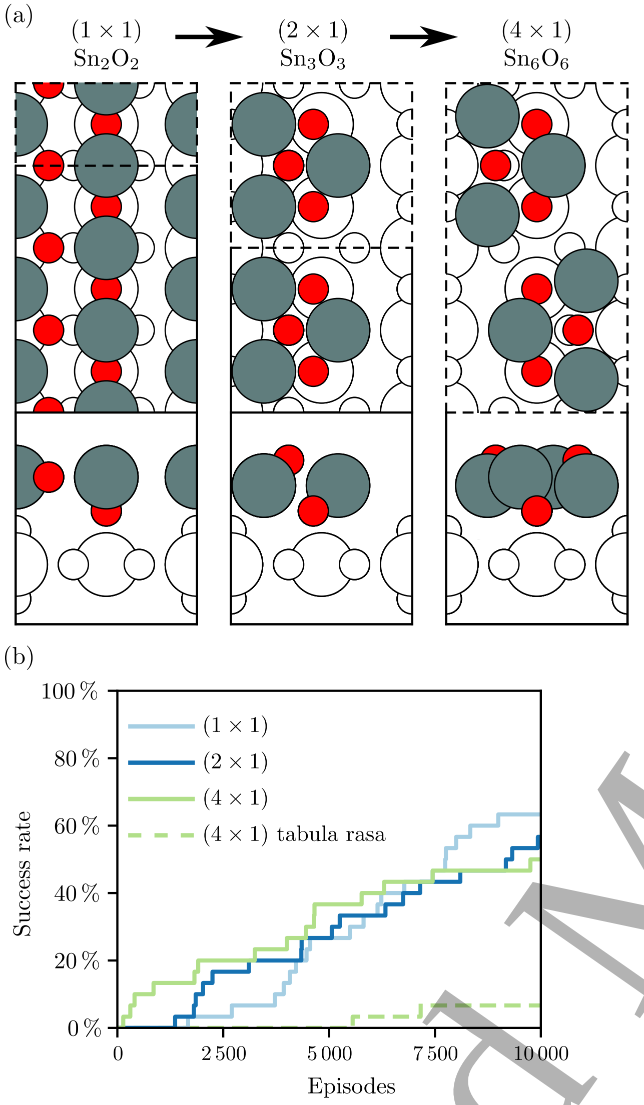

Figure 5. (a) Transfer protocol from a small to large unit cell. Dashed line marks the unit cell. (b) Success curve for each of the three sell sizes. The tabula rasa search starts without transferred knowledge.

relaxation reveals the global minimum energy structure in another 7 of the runs. This means that in 22 of 30 restarts, ASLA does identify structures within the basin containing the global minimum energy structure. ASLA thus convincingly discovers without human intervention the rutile $SnO_2(110)$-(4 × 1) reconstruction of reduced SnO_2 that otherwise remained elusive from the 1980's until 2017 [56].

To comment on the total CPU time spent in having ASLA identify the global minimum compared to conventional methods, two issues must be considered: (1) For the ASLA approach, the CPU resources spent on solving the preceding (1 × 1) and (2 × 1) problems whose agents are transferred onwards are vanishing compared to the resources spent on the (4 × 1) problem owing to the unfavorable scaling of DFT calculations. (2) For the conventional methods, a large number of single-point calculations are required for the relaxation of guessed structures, typically of the order 10^2 single-point calculations. With these two issues considered, the original search [56] reporting a total of 961 unique structures found for Sn_6O_6 after relaxation is thus seen to compare in computational expense to more than 10 restarts with the present ASLA approach. This actually renders the ASLA approach comparable if not favorable in terms of computational cost.

CONCLUSION

In the present work we have demonstrated a 3D structure prediction algorithm driven by reinforcement learning. By including additional configurational information in an image-like representation of a structure a convolutional neural network is able to sequentially place atoms thereby building the global minimum structure. By studying TiO_2 it was shown how ASLA learns to make rational choices, unlike purely stochastic search methods. Furthermore the strength of ASLA was demonstrated by transferring knowledge from one system to another to ultimately solve the complicated $SnO_2(110)$-(4 × 1) reconstruction. Having seen that ASLA may solve 3D problems, we expect that the development of a version of ASLA using 3D convolutions will be highly rewarding for systems where dense sampling is required in all three dimensions. Imposing rotational and mirror data augmentation with 3D convolutions will result in full E(3) symmetry, and is thus expected to perform even better for molecules, clusters, and crystalline structures, that may obtain any orientation in euclidean space.

ACKNOWLEDGMENTS

We acknowledge support from VILLUM FONDEN (Investigator Grant, Project No. 16562)

COMPUTATIONAL INFORMATION

A. TiO_2

Energy calculations were performed using DFTB and the atomic simulation environment (ASE) [69] in a 15.74 × 3.94 × 25 supercell with a grid of 1 × 1 × 1 k-points and periodic boundary conditions. The representation used a grid spacing of 0.197 Å resulting in a grid of 80 × 36 points and a depth of four.

B. SnO$_2$

Energy calculations were performed with DFT and ASE using the grid-based projector-augmented wave (GPAW) method [70, 71] using the PBE functional [72] with $(2 \times 2 \times 1)$ k-points using a basis of linear combinations of atomic orbitals [73] and periodic boundary conditions. Cells of size $(7.15 \times X \times 30)$ where $X \in \{3.25, 6.50, 13.00\}$ are used for increasing system size. Similarly the width of the penalty term is $\sigma = 10, 20$ and 40 respectively. A grid spacing of 0.325 Å is used giving rise to a grid of $22 \times X$ for $X \in \{10, 20, 40\}$ with 8 output channels.

C. Hyperparameters

Network hyperparameters are identical to those used in previous work [9]. Exploration parameters are likewise similar except for ε_ν which is higher due to the larger action space. A focused hyperparameter optimization has not been performed.

TABLE I. Fixed hyperparameters.

Hyperparameter	
ε	$2/T$
ε_0	20% of ε
ε_ν	80% of ε
ν	2%
ΔE (eV)	30
A (eV)	10
N_{batch}	512
β	10^{-5}
Learning rate	10^{-3}
Filter width (Å)	3
# Hidden layers	3
# Filters	10

[1] C. W. Coley, D. A. Thomas, J. A. M. Lummiss, J. N. Jaworski, C. P. Breen, V. Schultz, T. Hart, J. S. Fishman, L. Rogers, H. Gao, R. W. Hicklin, P. P. Plehiers, J. Byington, J. S. Piotti, W. H. Green, A. J. Hart, T. F. Jamison, and K. F. Jensen, Science **365**, eaax1566 (2019).

[2] R. Cuocolo, M. B. Cipullo, A. Stanzione, L. Ugga, V. Romeo, L. Radice, A. Brunetti, and M. Imbriaco, European radiology experimental **3**, 35 (2019).

[3] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, in *Advances in Neural Information Processing Systems 27* (2014) pp. 2672–2680.

[4] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," (2015), arXiv:1511.06434.

[5] M. Mirza and S. Osindero, "Conditional generative adversarial nets," (2014), arXiv:1411.1784.

[6] S. Grigorescu, B. Trasnea, T. Cocias, and G. Macesanu, Journal of Field Robotics (2019), 10.1002/rob.21918.

[7] K. Mills, M. Spanner, and I. Tamblyn, Phys. Rev. A **96**, 042113 (2017).

[8] G. B. Goh, C. Siegel, A. Vishnu, N. Hodas, and N. Baker, in *IEEE Winter Conference on Applications of Computer Vision* (2018) pp. 1340–1349.

[9] M. S. Jørgensen, H. L. Mortensen, S. A. Meldgaard, E. L. Kolsbjerg, T. L. Jacobsen, K. H. Sørensen, and B. Hammer, The Journal of Chemical Physics **151**, 054111 (2019).

[10] K. T. Butler, D. W. Davies, H. Cartwright, O. Isayev, and A. Walsh, Nature **559**, 547 (2018).

[11] J. Schmidt, M. R. G. Marques, S. Botti, and M. A. L. Marques, npj Computational Materials **2019**, 83 (2019).

[12] J. Behler and M. Parrinello, Phys. Rev. Lett. **98**, 146401 (2007).

[13] A. P. Bartók, M. C. Payne, R. Kondor, and G. Csányi, Phys. Rev. Lett. **104**, 136403 (2010).

[14] G. Schmitz, I. H. Godtliebsen, and O. Christiansen, The Journal of Chemical Physics **150**, 244113 (2019).

[15] J. S. Smith, B. T. Nebgen, R. Zubatyuk, N. Lubbers, C. Devereux, K. Barros, S. Tretiak, O. Isayev, and A. E. Roitberg, Nature Communications **10**, 2903 (2019).

[16] R. Jinnouchi, F. Karsai, and G. Kresse, Phys. Rev. B **100**, 014105 (2019).

[17] A. A. Peterson, The Journal of Chemical Physics **145**, 074106 (2016).

[18] O.-P. Koistinen, F. B. Dagbjartsdóttir, V. Ásgeirsson, A. Vehtari, and H. Jónsson, The Journal of Chemical Physics **147**, 152720 (2017).

[19] J. A. Garrido Torres, P. C. Jennings, M. H. Hansen, J. R. Boes, and T. Bligaard, Phys. Rev. Lett. **122**, 156001 (2019).

[20] J. Zhang and M. Chen, Phys. Rev. Lett. **121**, 010601 (2018).

[21] F. Noé, S. Olsson, J. Köhler, and H. Wu, Science **365**, eaaw1147 (2019).

[22] M. Rupp, A. Tkatchenko, K.-R. Müller, and O. A. von Lilienfeld, Phys. Rev. Lett. **108**, 058301 (2012).

[23] K. T. Schütt, H. E. Sauceda, P. Kindermans, A. Tkatchenko, and K. Müller, The Journal of Chemical Physics **148**, 241722 (2018).

[24] V. Tshitoyan, J. Dagdelen, L. Weston, A. Dunn, Z. Rong, O. Kononova, K. A. Persson, G. Ceder, and A. Jain, Nature **571**, 95 (2019).

[25] R. Ouyang, E. Ahmetcik, C. Carbogno, M. Scheffler, and L. M. Ghiringhelli, Journal of Physics: Materials **2**, 024002 (2019).

[26] B. Hartke, The Journal of Physical Chemistry **97**, 9973 (1993).

[27] D. M. Deaven and K. M. Ho, Phys. Rev. Lett. **75**, 288 (1995).

[28] A. R. Oganov and C. W. Glass, The Journal of Chemical Physics **124**, 244704 (2006).

[29] L. B. Vilhelmsen and B. Hammer, Phys. Rev. Lett. **108**, 126101 (2012).

[30] L. B. Vilhelmsen and B. Hammer, The Journal of Chemical Physics **141**, 044711 (2014).

[31] T. Ishikawa, T. Miyake, and K. Shimizu, Phys. Rev. B **100**, 174506 (2019).

[32] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, Science **220**, 671 (1983).

[33] C. J. Pickard and R. J. Needs, Journal of Physics: Condensed Matter **23**, 053201 (2011).

[34] A. R. Oganov, C. J. Pickard, Q. Zhu, and R. J. Needs, Nature Reviews Materials **4**, 331 (2019).

[35] E. L. Kolsbjerg, A. A. Peterson, and B. Hammer, Phys. Rev. B **97**, 195424 (2018).

[36] A. Denzel and J. Kästner, The Journal of Chemical Physics **148**, 094114 (2018).

[37] K. Xia, H. Gao, C. Liu, J. Yuan, J. Sun, H.-T. Wang, and D. Xing, Science Bulletin **63**, 817 (2018).

[38] M. Van den Bossche, The Journal of Physical Chemistry A **123**, 3038 (2019).

[39] E. Garijo del Río, J. J. Mortensen, and K. W. Jacobsen, Phys. Rev. B **100**, 104103 (2019).

[40] N. Bernstein, G. Csányi, and V. L. Deringer, npj Computational Materials **5**, 99 (2019).

[41] M. K. Bisbo and B. Hammer, Phys. Rev. Lett. **124**, 086102 (2020).

[42] T. L. Jacobsen, M. S. Jørgensen, and B. Hammer, Phys. Rev. Lett. **120**, 026102 (2018).

[43] K. H. Sørensen, M. S. Jørgensen, A. Bruix, and B. Hammer, The Journal of Chemical Physics **148**, 241734 (2018).

[44] F. Häse, L. M. Roch, C. Kreisbeck, and A. Aspuru-Guzik, ACS Central Science **4**, 1134 (2018).

[45] S. A. Meldgaard, E. L. Kolsbjerg, and B. Hammer, The Journal of Chemical Physics **149**, 134104 (2018).

[46] S. Chiriki, M.-P. V. Christiansen, and B. Hammer, Phys. Rev. B **100**, 235436 (2019).

[47] R. S. Sutton and A. G. Barto, *Reinforcement learing* (The MIT Press, 2018).

[48] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, T. Lillicrap, K. Simonyan, and D. Hassabis, Science **362**, 1140 (2018).

[49] O. Vinyals, I. Babuschkin, W. M. Czarnecki, M. Mathieu, A. Dudzik, J. Chung, D. H. Choi, R. Powell, T. Ewalds, P. Georgiev, J. Oh, D. Horgan, M. Kroiss, I. Danihelka, A. Huang, L. Sifre, T. Cai, J. P. Agapiou, M. Jaderberg, A. S. Vezhnevets, R. Leblond, T. Pohlen, V. Dalibard, D. Budden, Y. Sulsky, J. Molloy, T. L. Paine, C. Gulcehre, Z. Wang, T. Pfaff, Y. Wu, R. Ring, D. Yogatama, D. Wnsch, K. McKinney, O. Smith, T. Schaul, T. Lillicrap, K. Kavukcuoglu, D. Hassabis, C. Apps, and D. Silver, Nature **575**, 350 (2019).

[50] G. Novati, L. Mahadevan, and P. Koumoutsakos, Phys. Rev. Fluids **4**, 093902 (2019).

[51] E. Putin, A. Asadulaev, Y. Ivanenkov, V. Aladinskiy, B. Sanchez-Lengeling, A. Aspuru-Guzik, and A. Zhavoronkov, Journal of Chemical Information and Modeling **58**, 1194 (2018).

[52] M. Popova, O. Isayev, and A. Tropsha, Science Advances **4**, eaap7885 (2018).

[53] Z. Zhou, S. Kearnes, L. Li, R. N. Zare, and P. Riley, Scientific Reports **9**, 10752 (2019).

[54] J. S. Schreck, C. W. Coley, and K. J. M. Bishop, ACS Central Science **5**, 970 (2019).

[55] M. Lazzeri and A. Selloni, Phys. Rev. Lett. **87**, 266105 (2001).

[56] L. R. Merte, M. S. Jørgensen, K. Pussi, J. Gustafson, M. Shipilin, A. Schaefer, C. Zhang, J. Rawle, C. Nicklin, G. Thornton, R. Lindsay, B. Hammer, and E. Lundgren, Phys. Rev. Lett. **119**, 096102 (2017).

[57] K. Hansen, F. Biegler, R. Ramakrishnan, W. Pronobis, O. A. von Lilienfeld, K.-R. Müller, and A. Tkatchenko, The Journal of Physical Chemistry Letters **6**, 2326 (2015), pMID: 26113956.

[58] A. L. Maas, A. Y. Hannun, and A. Y. Ng, in *ICML Workshop on Deep Learning for Audio, Speech and Language Processing* (2013).

[59] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," (2015).

[60] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (The MIT Press, 2016).

[61] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," (2014), 1412.6980.

[62] G. Dolgonos, B. Aradi, N. H. Moreira, and T. Frauenheim, Journal of Chemical Theory and Computation **6**, 266 (2010).

[63] E. de Frésart, J. Darville, and J. Gilles, Solid State Communications **37**, 13 (1981).

[64] F. Jones, R. Dixon, J. Foord, R. Egdell, and J. Pethica, Surface Science **376**, 367 (1997).

[65] A. Atrei, E. Zanazzi, U. Bardi, and G. Rovida, Surface Science **475**, L223 (2001).

[66] J. Oviedo and M. Gillan, Surface Science **513**, 26 (2002).

[67] C. L. Pang, S. A. Haycock, H. Raza, P. J. Møller, and G. Thornton, Phys. Rev. B **62**, R7775 (2000).

[68] P. Ágoston and K. Albe, Surface Science **605**, 714 (2011).

[69] A. H. Larsen, J. J. Mortensen, J. Blomqvist, I. E. Castelli, R. Christensen, M. Duak, J. Friis, M. N. Groves, B. Hammer, C. Hargus, E. D. Hermes, P. C. Jennings, P. B. Jensen, J. Kermode, J. R. Kitchin, E. L. Kolsbjerg, J. Kubal, K. Kaasbjerg, S. Lysgaard, J. B. Maronsson, T. Maxson, T. Olsen, L. Pastewka, A. Peterson, C. Rostgaard, J. Schiøtz, O. Schütt, M. Strange, K. S. Thygesen, T. Vegge, L. Vilhelmsen, M. Walter, Z. Zeng, and K. W. Jacobsen, J. Phys. Condens. Matter **29**, 273002 (2017).

[70] J. J. Mortensen, L. B. Hansen, and K. W. Jacobsen, Phys. Rev. B **71**, 035109 (2005).

[71] J. Enkovaara, C. Rostgaard, J. J. Mortensen, J. Chen, M. Dułak, L. Ferrighi, J. Gavnholt, C. Glinsvad, V. Haikola, H. A. Hansen, H. H. Kristoffersen, M. Kuisma, A. H. Larsen, L. Lehtovaara, M. Ljungberg, O. Lopez-Acevedo, P. G. Moses, J. Ojanen, T. Olsen, V. Petzold, N. A. Romero, J. Stausholm-Møller, M. Strange, G. A. Tritsaris, M. Vanin, M. Walter, B. Hammer, H. Häkkinen, G. K. H. Madsen, R. M. Niem-

inen, J. K. Nørskov, M. Puska, T. T. Rantala, J. Schiøtz, K. S. Thygesen, and K. W. Jacobsen, Journal of Physics: Condensed Matter **22**, 253202 (2010).

[72] J. P. Perdew, K. Burke, and M. Ernzerhof, Phys. Rev. Lett. **77**, 3865 (1996).

[73] A. H. Larsen, M. Vanin, J. J. Mortensen, K. S. Thygesen, and K. W. Jacobsen, Phys. Rev. B **80**, 195112 (2009).