Exploring electron beam induced atomic assembly via reinforcement learning in a molecular dynamics environment

Rama K Vasudevan\textsuperscript{1,∗∗,} , Ayana Ghosh\textsuperscript{1,2 ,} Maxim Ziatdinov\textsuperscript{1,2} and Sergei V Kalinin\textsuperscript{1}

\textsuperscript{1} Center for Nanophase Materials Sciences, Oak Ridge National Laboratory, Oak Ridge, TN 37831, United States of America
\textsuperscript{2} Computational Sciences and Engineering Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831, United States of America

E-mail: vasudevanrk@ornl.gov and sergei2vk@ornl.gov

Received 13 August 2021, revised 5 November 2021
Accepted for publication 12 November 2021
Published 21 December 2021

Abstract

Atom-by-atom assembly of functional materials and devices is perceived as one of the ultimate targets of nanotechnology. Recently it has been shown that the beam of a scanning transmission electron microscope can be used for targeted manipulation of individual atoms. However, the process is highly dynamic in nature rendering control difficult. One possible solution is to instead train artificial agents to perform the atomic manipulation in an automated manner without need for human intervention. As a first step to realizing this goal, we explore how artificial agents can be trained for atomic manipulation in a simplified molecular dynamics environment of graphene with Si dopants, using reinforcement learning. We find that it is possible to engineer the reward function of the agent in such a way as to encourage formation of local clusters of dopants under different constraints. This study shows the potential for reinforcement learning in nanoscale fabrication, and crucially, that the dynamics learned by agents encode specific elements of important physics that can be learned.

Supplementary material for this article is available online

Keywords: dopants, reinforcement learning, molecular dynamics, graphene

Assembly of matter atom by atom towards devices and materials with desired functionality is perceived as the ultimate goal of nanoscience and nanotechnology, as envisioned by some of the leading scientists of the 20th century [1, 2]. Beyond the emergent and serendipitous applications in quantum computing, nanorobotics, and medicine, the ability to fabricate atomic scale structures is instrumental for physical research. As stated by Feynman, ‘what I cannot make I cannot understand’. Correspondingly, a capability to fabricate atomically defined structures and probe their functionalities is key to unlocking the fundamental physics of the atomic world.

While perceived as largely speculative for most of the last century, the experiments by Don Eigler via Scanning Tunneling Microscopy demonstrated the potential of this method to create selected atomic structures [3, 4] and heralded the explosive growth of nanoscience in the last thirty years. In accordance with the Feynman statement, many...
examples of fundamental studies such as quantum corrals [5],
molecular cascades [6], quantum dots [7] and more have been
demonstrated. Over the last decade, the combination of surface
chemistry and STM manipulation was established as an
enabling component of quantum computing infrastructure
allowing for creation of phosphorus qubits [8–10]. Remark-
ably, while not having the same degree of precision, many
other atoms including oxygen and Sulphur can be manipu-
lated [11, 12].

In the last decade, it was shown that sub-atomically
focused electron beam in scanning transmission electron
microscopy (STEM) can induce controllable changes in
atomic structure on the atomic level. While electron beam
damage was recognized as an issue since the earliest days of
electron microscopy, the advances brought by aberration
correction [13–15] illustrated that the effects of energy
transfer from the electron beam to the solid can be much more
subtle, including activation of a set of chemical transforma-
tions and phase transitions, [16, 17] lateral and rotational
motion of atoms and molecular groups, and vacancy forma-
tion. It was immediately realized that control of these trans-
formations can be used as a basis for atomic fabrication [18],
and direct controllable motion of Si atoms in graphene
[19, 20], single-site doping [19], and even formation of the
homoatomic [21] and heteroatomic [22] molecular structures
has been demonstrated. It should be noted that electron beam-
induced patterning has been known and well-studied for
considerably longer, with examples including writing patterns
in graphene sheets [23] with STEM, as well as nanocrystal-
line film fabrication with the scanning electron beam [24],
where the nanocrystal growth can be controlled by the degree
of irradiation.

However, the electron beam manipulation in STEM, as
compared to probe-based manipulation in STM and non-
contact Atomic Force Microscopy (nc-AFM), has a number
of significant differences. First, STEM offers the potential
for much higher throughput, since the intrinsic latencies in
electron beam motion and image acquisition are much lower
than that of the mechanical probe in STM and noncontact-
AFM. However, while the probe-induced chemistry in STM
can be well-controlled, electron beams can induce a broad
variety of beam-induced reactions, with associated difficulty
of control. Both considerations necessitate the automatization
of the process, i.e. incorporation of automated workflows
controlling the microscope, to discover the cause-and-effect
relationships between probe position and induced dynamic
changes in the atomic structure, and further harness these
towards controllable modification of solids.

At the same time, there has been considerable interest in
the use of reinforcement learning (RL) for tackling a variety
of sequential decision-making tasks, with applications as far
ranging as robotics to drug design to computer gameplay.
The advantage that RL agents can provide is that, if suitably
trained, they will be capable of automating the atomic
manipulation process with greater speed as well as potentially
greater efficiency, in addition to the obvious benefit of scal-
ability of the process. The first step towards this goal is to
train RL agents in a simulated atomic fabrication
environment, to determine the feasibility and major chal-

Here, we explore the use of reinforcement learning for
atomic manipulation implemented using a model molecular
dynamics-run (MD) environment. The effect of the electron
beam is modeled using deterministic energy momentum
transfer to the selected atoms in the MD system. We show
that utilizing RL, it is possible to train agents to perform
atomic-level assembly in a very highly stochastic environ-
ment, and furthermore, inspect learned policies. The asso-
ciated policy shifts are subtle, indicating the highly dynamic
and somewhat unpredictable nature of the environment, and
we explore how reward shaping affects the final policy. Some
notes on extensibility of the process are discussed.

**RL environment: molecular dynamics simulation**

As a model system, we have utilized Si dopants in graphene
with a Lennard–Jones (LJ) potential given within the python-
based atomic simulation environment (ASE) [26] Full details
of the environment and model are given in the in supple-
mentary S1 (available online at stacks.iop.org/NANO/33/
115301/mmedia). Briefly, the MD simulations utilize a 64
atoms hexagonal graphene supercell where one or a few C
atoms can be replaced by an Si dopant. Momenta can be
imparted on the atoms in all three (x, y, z) dimensions. This
sets the environment for the reinforcement learning for-
mulation of the problem. At the same time, we emphasize that
this is a very simplistic environment that is not intended to
accurately describe all aspects of momentum transfer from the
electron beam. Indeed, this is for all intents a simplistic
model, and we are aware of the limitations of utilizing an LJ
potential. However, our idea is not to utilize the most accurate
model, given the heavy computational requirements of RL
methods, but rather to show the case of an MD simulation in
which RL can be utilized for matter manipulation. In general,
dispersive forces are important for graphene surfaces which
are only captured by high level quantum mechanical
approaches like couple cluster methods and density functional
theory simulations, and density functional tight binding with
inclusion of such interactions. The electronic structure infor-
mation then, can be imported to molecular dynamics envi-
ronments with appropriate (reactive) force fields such as
ReaxFF to study beam effects on reconstruction of geometry,
formation and propagation of defects, temperature depend-
ence and healing mechanisms of graphene. Even though,
these simulations can be performed stand-alone and turn out
to be computationally expensive, they become simply
unfeasible, to be used environments at the present time given
computational constraints and the sample inefficiency of RL.
Nonetheless, our environment provides a suitably difficult
enough ‘toy’ model to gauge the feasibility of the idea.
RL problem setup

Prior to implementation of reinforcement learning on top of the molecular dynamics environment, we briefly discuss the fundamental RL concepts, namely the environment, state, action, and reward. The environment describes the system within which agent operates. In this case, this is molecular dynamics model. The state is defined as what is available or visible to the agent, and in this case, is the static projection of the surface, emulating the traditional high-angle annular dark field (HAADF) images. Here the state also includes the previous HAADF image, so that knowledge of the dynamic changes is inherently encoded in the shifts in atomic coordinates. The actions are the actions that the agent can take, and here comprises the momentum transfer from the beam to the atomic unit emulating the beam positioning in the specified registry towards the selected nucleus and assuming deterministic momentum transfer. Finally, reward is an externally defined measure that can be tuned based on the desired structure and/or properties. Here, the ultimate target of the learning is the assembly of the desired atomic structure. Exact formulation of the reward function comprises an important part of RL algorithm that strongly affects the learning speed and will be discussed below.

An example of the state is shown in (a). The three Si dopant atoms are always initiated to be in the same position at the beginning of each episode. After 300 time steps, the HAADF image is returned to the agent. The added velocities or actions lead to change in position of atoms even after performing the task for a small time as compared to situations when no external energy is being introduced to the system for constant energy simulations. The agent’s policy is a multivariate Gaussian policy, the mean and variance of which are parameterized by a neural network (figure 1(b)). This neural network, termed the ‘actor’ neural network, takes as input the HAADF images at time \( t = t, t = t - 1 \), and passes the data through two convolutional layers and two fully connected layers, before outputting the mean and standard deviations for the multivariate gaussian policy. This same network structure is duplicated for the ‘critic’ network, which outputs the value of the state \( V(s) \) (discussed below). Based on the outputs of the actor network, the multivariate gaussian is sampled, and then the momentum transfer to the three dopants is achieved via modification of the individual dopant’s \( x, y \) and \( z \) components of velocity in the MD simulation, as shown in figure 1(c). The process then repeats until 6000 MD time steps are completed, signifying the end of the episode.

Reward function and RL training

We aim to train policies that will assemble the three Si dopants as closely as possible, as has been shown experimentally. We define the reward function as

\[
R = w_0 d_S + w_1 d_c + w_2 A_{\text{sum}},
\]

where \( w_0, w_1 \) and \( w_2 \) are weight coefficients, \( d_S \) defines the sum of (Euclidean) distances between the three Si dopants, \( d_c \) is the sum of (Euclidean) distance of dopants to the center of the lattice, and \( A_{\text{sum}} \) is the magnitude of the actions imparted on the atoms. The first part of the reward function deals with the desirability to create trimer-like structures, and this sets \( w_0 \)
to be <0 (to penalize dopants moving away from each other). Similarly, \( w_1 \) is <0 given that we wish to penalize dopants agglomerating away from the center of the image. Finally, \( w_2 \) can be set to 0, in which case there is no penalty for large momentum transfers; we explore this parameter in detail below but note that it serves as a kind of regularization to limit large momentum transfers which can have undesirable effects (such as inducing more lattice instability, causing vacancies, etc). Default values for \( w_0 = -100 \) and \( w_1 = -300 \). The reward is returned after every action, i.e. a total of 20 times per episode. Clearly, given the negative values of the weights, the maximum available reward for the agent is less than 0.

The goal of RL [27] is to learn a policy to maximize the cumulative (usually ‘discounted’) reward, i.e. to maximize the objective function \( J \):

\[
J(\pi) = \mathbb{E} \left[ \sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \right],
\]

where \( \gamma \) is a discount factor between 0 and 1, actions \( a_t \) are drawn from the policy \( \pi(a_t|s_t) \) and the states \( s_{t+1} \) are drawn from the environment, as a result of taking action \( a_t \) at state \( s_t \). Maximizing this expectation can be achieved via perturbing the policy parameters \( \theta \) in the direction of the gradient of \( J \) with respect to the policy parameters, and is given by the policy gradient formulation:

\[
\nabla_\theta J(\pi) \approx \sum_{t=0}^{\infty} \nabla \log \pi(a_t|s_t; \theta) R_t,
\]

where \( R_t \) is the cumulative discounted return from time \( t \), i.e. \( R_t = \sum_{i=0}^{\infty} \gamma^i r(s_{t+i}, a_{t+i}) \). Note that this formulation ensures that actions that affect future rewards are encouraged. Unfortunately, this formulation has very high variance in practice, and this variance can be substantially reduced via use of an appropriate baseline. One choice is the value function baseline, giving rise to the actor-critic algorithm:

\[
\nabla_\theta J(\pi) \approx \sum_{t=0}^{\infty} \nabla \log \pi(a_t|s_t; \theta) (R_t - V(s_t)),
\]

where the ‘Value’ function \( V(s_t) = \mathbb{E}\left[\sum_{i=0}^{\infty} \gamma^i r(s_{t+i}, a_{t+i})\right] \). This formulation is sometimes termed as ‘advantage actor-critic’ (A2C) [28], and intuitively expresses the idea that actions that provide greater advantage than the current expectation of the value of being in that state should be encouraged, and those that provide less returns than expected should be discouraged. This value function is estimated by a neural network termed the critic. A recent extension of the A2C approach is the so-called Stein variational policy gradient (SVPG) approach [29, 30] More details of implementation can be found in supplementary S2.

Figure 2. The mean reward of the training agents versus training iterations for three different weight regularization parameters are plotted in (a). The rewards as a function of training runs are shown in (b)–(d) for the corresponding regularization parameters. Different colors in (b)–(d) correspond to different agents. Smoothing of the training curves has been applied to each case with a window size of 10.
Results from the mean of the trained agents is shown in figure 2(a) for three different values of weight regularization parameter, $w_2$. Of note, all of the means show steady improvement through training, although it is evident that individual agents show non-monotonic increases which is somewhat typical of training in RL. The reward values are not directly comparable due to the different reward functions for each of (b)–(d), but progress appears saturated at the high weight regularization, and near saturated for the other two scenarios.

Subsequently, we visualized the results of the highest performing agent in the case of $w_2 = 0$, i.e. no weight regularization. An example run is shown in figure 3, which indicates the progression of states through episode step 1–20. Arrows indicate the actions taken at each step to each dopant atom, and the color of the arrow indicates the degree of ‘z’ component to the velocity added. The agent shows strong success at moving the dopants together towards the center of the lattice, and this progress is highly repeatable across runs. This confirms that RL agents can be trained in this environment to adjust dopant momenta such that they agglomerate towards pre-chosen sites on the graphene lattice.

**Effect of weight regularization**

To observe the effects of weight regularization, we trialed the trained agents with three different values of weight regularization parameter $w_2$ for five episodes and accumulated statistics of the actions performed. The violin plot of the actions for the three different agents are shown in figure 4(a), and clearly indicates that higher levels of weight regularization led to smaller changes to velocity, i.e. that this achieves what the reward function was designed to accomplish. It should be noted, however, that this did not in turn lead to the goal of reducing lattice disordering during the dopant-moving assembly process; the reason may be that rather than the overall magnitude of the velocity vector, it is the individual components of velocity that act to disrupt the lattice structure. This behavior can also be tied to an equidistribution of magnitudes of velocities that is considered at the initialization stage of the MD simulations, if the components are not mentioned individually, reflecting in the computed forces of individual atoms. We have also seen that specifying the velocity components individually as an action helps to retain the lattice structure. It is of course possible to alter the reward function such that we directly penalize the lattice disruption, however this can be more difficult in terms of design of reasonable metrics of lattice degradation and is not explored in this study. Moreover, note that in experiment, it has already been shown possible to move Si dopants in graphene without significant damage to the host lattice [19]. Finally, very strong weight regularization did result in decreased ability to agglomerate the dopants towards the center of the lattice, but this effect was less pronounced when the weight regularization was weaker.

To obtain more insight into how the shape of the reward function affects the learned policies, we plot the policy for a randomly chosen state from the environment. The actions performed on this state by the three agents are shown in...
The actions for three different agents are shown as violin plots in (a), which were trained with different levels of weight regularization (WR). The actions performed on a particular state and respective policies from actions are plotted in (b) and (c), respectively.

Figure 4. The actions for three different agents are shown as violin plots in (a), which were trained with different levels of weight regularization (WR). The actions performed on a particular state and respective policies from actions are plotted in (b) and (c), respectively.

figure 4(b), and the respective policy from which the actions were sampled are plotted as Gaussians in figure 4(c). All three agents applied velocity vectors that were to the right, for the green and red dopant atoms, but differed in their preference to move the blue Si dopant. Inspection of the actual policies in figure 4(c) reveal several expected trends. The first is that the policies do not deviate much from ideal Gaussians, i.e. changes to velocity should be rather subtle. This may be due to the high stochasticity of the environment. Next, as the weight regularization increases, the variance of the policy reduces, in accordance with reducing the overall strength of the actions to increase the cumulative reward. Finally, and perhaps most importantly, the $z$-component of the velocity appears to be important because there is considerable weight on either side of the peak; this suggests that some small $z$-component of velocity is beneficial for the dopant motion. Similar conclusions were arrived at in mechanistic analysis by Susi [20, 31–33] previously and is therefore very notable that the RL agent has, through only training, also discovered this mechanism. That said, we again emphasize that the point here is that the policy learns important dynamics of the environment.
Ensemble training

Finally, we note that the Stein variational policy gradient method enables training multiple agents in an ensemble. One of the advantages of such an approach is that this variance can improve the robustness of policies, i.e. learning multiple different methods to achieve the same target objective. An example of how the policies differ is shown in figures 5(a), (b), where a random state is selected, and the two top-achieving agents are tested. Substantial differences in the policies are evident and are particularly evident for example for the momentum transfer to dopant 3 (green). This is further confirmed by inspecting the trained agents throughout the run (supplementary S3) where differences between the two agents are readily apparent.

Summary and discussion

This analysis illustrates the need for development of models of atomistic effects during beam solid interactions, including fast momentum transfer using classical knock-on mechanism and more subtle effects associated with local excitation of bonding electrons [34, 35] and hot carriers [36]. Indeed it will be necessary to incorporate the appropriate level of theory to correctly stimulate the true system dynamics, but at the same time, the simulations must run rapidly for RL to actually be feasible. This would therefore require likely a combination of hardware acceleration (e.g. FPGA or GPU), as well as potentially speed ups via approximations, e.g. utilizing a variety of ML approaches [37]. Furthermore, these would then need to be integrated with STEM simulators that can produce realistic HAADF images, such as via the muSTEM code [38], for conversion of the simulation structure to the actual state.

Notably, the dearth of theoretical models in this domain suggests that future exploration should target co-navigation approaches, when the model is being refined along the experiment and in turn used to guide RL strategy. Finally, given the extremely short latencies of the STEM experiment, this in turn necessitates development of edge computing capabilities bringing the MD processing to the edge of STEM experiment. Overall, this approach is expected both to enable atom-by-atom fabrication and in the process gain insight into beam-induced chemical effects.

At the same time, even with trained policies one needs to consider numerous factors for experimental implementation. One issue is the microscope drift and that imaging itself may cause motion of the dopants. The drift and the imaging issues can be somewhat negated by drift correction capabilities, along with sparse scanning, enabling low-dose imaging with reconstruction provided by compressed sensing and/or gaussian process algorithms, as we have previously explored [39]. The stochastic nature of the policies ensures that it should still function in the highly dynamic environment (e.g. atoms randomly moving, defects being generated, etc). However, distribution shift [40] between the real experiment and the training environment will potentially require re-training.

To summarize, here we combine the reinforcement learning approach with molecular dynamics simulations to explore atomic fabrication with artificial agents. For the chosen toy model, we found it is possible to engineer the momentum transfer in such a way as to deterministically move dopants towards desired locations and create agglomerations, in view of future potential of atomic-scale matter assembly via an automated approach. We explore the use of regularization in the reward function to temper the momentum transfer, and inspect multiple policies trained in ensemble to produce multiple methods of achieving the same target goal, potentially improving robustness.

Acknowledgments

This effort was performed and partially supported (RKV, MZ) at the Oak Ridge National Laboratory’s Center for Nanophase Materials Sciences (CNMS), a US Department of Energy,
Office of Science User Facility, and by US Department of Energy, Office of Science, Office of Basic Energy Sciences Data, Artificial Intelligence and Machine Learning at DOE Scientific User Facilities program under Award Number 34532 (AG, SVK).

Data availability statement

All data that support the findings of this study are included within the article (and any supplementary files).

ORCID iDs

Rama K Vasudevan @ https://orcid.org/0000-0003-4692-8579
Ayana Ghosh @ https://orcid.org/0000-0002-0432-3689
Maxim Ziatdinov @ https://orcid.org/0000-0003-2570-4592
Sergei V Kalinin @ https://orcid.org/0000-0001-5354-6152

References

[1] Drexler E 1987 Engines of Creation: The Coming Era of Nanotechnology (New York: Anchor Books)
[2] Feynman R P 1960 There’s plenty of room at the bottom Caltech Eng. Sci. 23 22–36
[3] Eigler D M, Lutz C P and Rudge W E 1991 An atomic switch realized with the scanning tunneling microscope Nature 352 600–3
[4] Eigler D M and Schweizer E K 1990 Positioning single atoms with a scanning tunneling microscope Nature 344 524–6
[5] Heller E J, Crommie M F, Lutz C P and Eigler D M 1994 Scattering and absorption of surface electron waves in quantum corrls Nature 369 464–6
[6] Heinrich A J, Lutz C P, Gupta J A and Eigler D M 2002 Molecule cascades Science 298 1381–7
[7] Fuechsle M, Mahapatra S, Zwanenburg F, Friesen M, Eriksson M and Simmons M Y 2010 Spectroscopy of few-electron single-crystal silicon quantum dots Nat. Nanotechnol. 5 502–5
[8] Fuechsle M, Miwa J A, Mahapatra S, Ryu H, Lee S, Warschkow O, Hollenberg L C L, Klimeck G and Simmons M Y 2012 A single-atom transistor Nat. Nanotechnol. 7 242–6
[9] Schofield S R, Curson N J, Simmons M Y, Rues F J, Hallam T, Oberbeck L and Clark R G 2003 Atomically precise placement of single dopants in Si Phys. Rev. Lett. 91 136104
[10] Weber B et al 2012 Ohm’s law survives to the atomic scale Science 335 64–7
[11] Ko W, Ma C X, Nguyen G D, Kolmer M and Li A P 2019 Atomic-scale manipulation and In Situ characterization with scanning tunneling microscopy Adv. Funct. Mater. 29 24
[12] Vasudevan R K, Tselev A, Gianfrancesco A G, Baddorf A P and Kalinin S V 2015 Atomic-scale electrochemistry on the surface of a manganese by scanning tunneling microscopy Appl. Phys. Lett. 106 143107
[13] Deylby N, Krivanek O L, Nellist P D, Batson P E and Lupini A R 2001 Progress in aberration-corrected scanning transmission electron microscopy J. Electron Microsc. 50 177–85
[14] Krivanek O L et al 2010 Atom-by-atom structural and chemical analysis by annular dark-field electron microscopy Nature 464 571–4
[15] Krivanek O L, Delby N, Spence A J, Camps R A and Brown L M 1997 Electron Microscopy and Analysis ed J M Rodenburg 1997 (Bristol: IOP Publishing) pp 35–40
[16] Pennycook T J, Jones L, Pettersson H, Coelho J, Canavan M, Mendoza-Sanchez B, Nicolosi V and Nellist P D 2014 Atomic scale dynamics of a solid state chemical reaction directly determined by annular dark-field electron microscopy Sci. Rep. 4 7555
[17] Zheng H M, Rivest J B, Miller T A, Sadtler B, Lindenberg A, Toney M F, Wang L W, Kiesielowski C and Alivisatos A P 2011 Observation of transient structural-transformation dynamics in a Cu2SN nanorod Science 333 206
[18] Kalinin S V, Borisovich A and Jesse S 2016 Fire up the atom forge Nature 539 485–7
[19] Dyck O, Kim S, Kalinin S V and Jesse S 2017 Placing single atoms in graphene with a scanning transmission electron microscope Appl. Phys. Lett. 111 113104
[20] Toma Susi J C M and Kotakoski J 2017 Manipulating low-dimensional materials down to the level of single atoms with electron irradiation Ultramicroscopy 180 163–72
[21] Dyck O, Kim S, Jimenez-Izal E, Alexandrova A N, Kalinin S V and Jesse S 2018 Building structures atom by atom via electron beam manipulation Small 14 1801771
[22] Dyck O, Zhang C, Rack P D, Fowlkes J D, Sumpter B, Lupini A R, Kalinin S V and Jesse S 2020 Electron-beam introduction of heteroatomic Pt–Si structures in graphene Carbon 161 750–7
[23] Zhang W, Zhang Q, Zhao M-Q and Kuhn L T 2013 Direct writing on graphene ‘paper’ by manipulating electrons as ‘invisible ink’ Nanotechnology 24 275301
[24] Zhang W, Liu Z-Q and Furuya K 2008 Fabrication and characterization of cellular iron nanocrystalline film Nanotechnology 19 135302
[25] Rajak Pankaj, Wang Beibei, Nomura Ken-ichi, Luo Ye, Nakano Aiichiro, Kalia Rajiv and Vashishta Priya 2021 Autonomous reinforcement learning agent for stretchable kirigami design of 2D materials npj Computational Materials 7 102
[26] Al A H L E 2017 The atomic simulation environment—a python library for working with atoms J. Phys.: Condens. Matter 29 273002
[27] Sutton R S and Barto A G 2018 Reinforcement learning: An introduction (Cambridge, MA: MIT press)
[28] Mnih V, Badia A P, Mirza M, Graves A, Lillicrap T, Harley T, Silver D and Kavukcuoglu K 2016 Asynchronous methods for deep reinforcement learning PMLR pp 1928–37
[29] Liu S, Borodinov N, Vleck I, Lu D, Laanait N and Vasudevan R K 2020 Application of variational policy gradient to atomic-scale materials synthesis arXiv:2006.15644
[30] Liu Y, Ramachandran P, Liu Q and Peng J 2017 Stein variational policy gradient 33rd Conference on Uncertainty in Artificial Intelligence, UAI 2017 (Sydney, Australia)
[31] Susi T, Kepaptsoglou D, Lin Y C, Ramasse Q M, Meyer J C, Suenaga K and Kotakoski J 2017 Towards atomically precise manipulation of 2D nanostructures in the electron microscope 2D Mater. 4 042004
[32] Susi T et al 2014 Silicon-carbon bond inversions driven by 60-keV electrons in graphene Phys. Rev. Lett. 113 5
[33] Tripathi M, Mittelberger A, Pike N A, Mangler C, Meyer J C, Verstraete M J, Kotakoski J and Susi T 2018 Electron-beam manipulation of silicon dopants in graphene Nano Lett. 18 5319–23
[34] David B, Lingerfelt T Y, Yoshimura A, Ganesh P, Jakowski J and Sumpter B G 2021 Nonadiabatic effects on
defect diffusion in silicon-doped nanographenes Nano Lett. 21 236–42

[35] Lingerfelt D B, Ganesh P, Jakowski J and Sumpter B G 2019 Electronically nonadiabatic structural transformations promoted by electron beams Adv. Funct. Mater. 29 1901901

[36] Weber W J, Zarkadoula E, Pakarinen O H, Sachan R, Chisholm M F, Liu P, Xue H Z, Jin K and Zhang Y W 2015 Synergy of elastic and inelastic energy loss on ion track formation in SrTiO3 Sci. Rep. 5 7726

[37] Kochkov D, Smith J A, Alieva A, Wang Q, Brenner M P and Hoyer S 2021 Machine learning–accelerated computational fluid dynamics Proc. Natl Acad. Sci. 118 e2101784118

[38] Allen L J and Findlay S 2015 Modelling the inelastic scattering of fast electrons Ultramicroscopy 151 11–22

[39] Kelley K P, Ziatdinov M, Collins L, Susner M A, Vasudevan R K, Balke N, Kalinin S V and Jesse S 2020 Fast scanning probe microscopy via machine learning: non-rectangular scans with compressed sensing and gaussian process optimization Small 16 6

[40] Lin Z, Thomas G, Yang G and Ma T 2020 Model-based adversarial meta-reinforcement learning Advances in Neural Information Processing Systems ed H Larochelle, M Ranzato, R Hadsell, M F Balcan and H Lin 33 (Red Hook, NY: Curran Associates, Inc.) 10161