

# An off-lattice, self-learning kinetic Monte Carlo method using local environments

Dhrubajit Konwar, Vijesh J. Bhute, and Abhijit Chatterjee

Citation: *The Journal of Chemical Physics* **135**, 174103 (2011); doi: 10.1063/1.3657834

View online: <https://doi.org/10.1063/1.3657834>

View Table of Contents: <http://aip.scitation.org/toc/jcp/135/17>

Published by the *American Institute of Physics*

---

## Articles you may be interested in

[Long time scale kinetic Monte Carlo simulations without lattice approximation and predefined event table](#)

*The Journal of Chemical Physics* **115**, 9657 (2001); 10.1063/1.1415500

[Accurate acceleration of kinetic Monte Carlo simulations through the modification of rate constants](#)

*The Journal of Chemical Physics* **132**, 194101 (2010); 10.1063/1.3409606

[Adaptive kinetic Monte Carlo for first-principles accelerated dynamics](#)

*The Journal of Chemical Physics* **129**, 114104 (2008); 10.1063/1.2976010

[Accuracy of a Markov state model generated by searching for basin escape pathways](#)

*The Journal of Chemical Physics* **138**, 084103 (2013); 10.1063/1.4792439

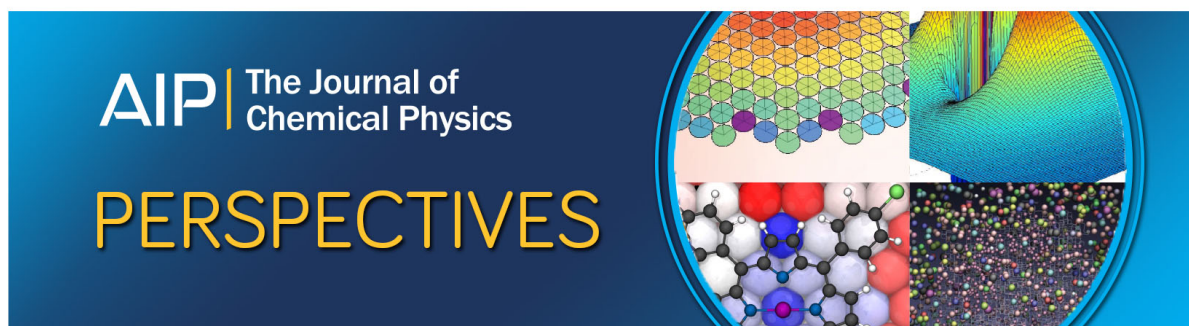
[Building a kinetic Monte Carlo model with a chosen accuracy](#)

*The Journal of Chemical Physics* **138**, 244112 (2013); 10.1063/1.4812319

[A climbing image nudged elastic band method for finding saddle points and minimum energy paths](#)

*The Journal of Chemical Physics* **113**, 9901 (2000); 10.1063/1.1329672

---



# An off-lattice, self-learning kinetic Monte Carlo method using local environments

Dhrubajit Konwar, Vijesh J. Bhute, and Abhijit Chatterjee<sup>a)</sup>

*Department of Chemical Engineering, Indian Institute of Technology - Kanpur, Kanpur, Uttar Pradesh 208016, India*

(Received 22 July 2011; accepted 13 October 2011; published online 2 November 2011)

We present a method called local environment kinetic Monte Carlo (LE-KMC) method for efficiently performing off-lattice, self-learning kinetic Monte Carlo (KMC) simulations of activated processes in material systems. Like other off-lattice KMC schemes, new atomic processes can be found on-the-fly in LE-KMC. However, a unique feature of LE-KMC is that as long as the assumption that all processes and rates depend only on the local environment is satisfied, LE-KMC provides a general algorithm for (i) unambiguously describing a process in terms of its local atomic environments, (ii) storing new processes and environments in a catalog for later use with standard KMC, and (iii) updating the system based on the local information once a process has been selected for a KMC move. Search, classification, storage and retrieval steps needed while employing local environments and processes in the LE-KMC method are discussed. The advantages and computational cost of LE-KMC are discussed. We assess the performance of the LE-KMC algorithm by considering test systems involving diffusion in a submonolayer Ag and Ag-Cu alloy films on Ag(001) surface.

© 2011 American Institute of Physics. [doi:[10.1063/1.3657834](https://doi.org/10.1063/1.3657834)]

## I. INTRODUCTION

Atomic processes that govern macroscopic phenomena in materials often have large activation barriers and occur at time scales that are simply beyond the reach of standard atomistic simulation techniques such as the molecular dynamics (MD) method. Although the recent development of MD-based accelerated dynamics methods<sup>1–5</sup> has successfully extended the MD time scales to microseconds and longer with an accuracy that is almost comparable to MD, these methods continue to remain computationally prohibitive for a large class of important problems such as diffusion, nucleation and growth, crystallization, defect evolution and chemical reactions. Consequently, the kinetic Monte Carlo (KMC) method<sup>6–9</sup> is often the method of choice when milliseconds to hours time scales need to be reached. Instead of studying the vibrational motion of atoms, as done in MD, the KMC method can reach long time scales by obtaining a sequence of state-to-state transitions. In a standard KMC implementation, a catalog of process rates is provided at the outset along with their rates. Subsequently, in each KMC iteration an atomic process is randomly selected from the catalog with a probability that is proportional to its rate, time is advanced, the system moves to a new state, and the list of possible processes for the new state of the system is updated. When the underlying assumptions of KMC are satisfied, namely that the first escape time from a state is exponentially distributed, the statistics collected from the KMC simulations of rare events can be as accurate as those from MD.

Despite the higher computational efficiency of KMC, approximations introduced in standard KMC implementations

place major limitations on KMC. Experimental and computational studies of atomic processes over the last few decades have demonstrated that even for simple material systems, it is difficult to guess all the relevant process mechanisms for a given material system.<sup>10–12</sup> Therefore, providing an accurate process catalog at the outset is not a straightforward task for materials that are not well understood. Even when the types of possible atomic moves are known, their rates depend on the state of the system. The rates have to be computed for all arrangements of the atoms that are encountered in the dynamics.<sup>13</sup> A common approach for obtaining rates entails the use of mathematical fits that attempt to qualitatively capture the environment dependence of the process,<sup>14–17</sup> which could in some situations result in inaccurate rates. In addition, it is common to perform on-lattice KMC simulations where the continuous positions of atoms are mapped to a discrete lattice. There are a large number of examples, e.g., multicomponent systems, defects in materials including point defects, dislocations, grain boundaries and interfaces, nucleation and growth problems, where the on-lattice representation cannot be employed. However, the use of an on-lattice KMC algorithm simplifies the implementation of KMC because in off-lattice KMC simulations (in off-lattice KMC atoms are allowed to take continuous positions) the system can evolve towards states for which the processes might be completely unknown. Such situations should be avoided if the process catalog is fixed at the beginning of the KMC simulation.

These shortcomings of KMC have prompted the recent development of a number of off-lattice and self-learning KMC algorithms.<sup>18–23</sup> The main goal of these algorithms is to allow atoms to reside in off-lattice positions and to continually search for new processes so that the KMC catalog

<sup>a)</sup>Electronic mail: [achatter@iitk.ac.in](mailto:achatter@iitk.ac.in).

remains accurate at least for the current state of the system.<sup>18</sup> Some self-learning KMC algorithms<sup>19,20,23</sup> store processes that were observed previously in a process catalog. Although only a fraction of the processes stored in the catalog might be possible for a given state of the system, reusing process information can significantly reduce the number of computationally intensive searches for transitions that are possible from a state, especially when the system keeps revisiting the same states. While building a process catalog the main challenge lies in deciding what information should be stored so that the catalog can each time provide the correct list of processes and associated rates. One catalog-building approach involves storing the initial and final positions of all atoms during a process.<sup>18</sup> Whether a process from the catalog can occur or not can be correctly recognized by comparing the atomic positions in the current state of the system with the initial position of the atoms in the process. Since the number of atomic processes increases dramatically with the number of atoms in the system, the catalog being generated using this approach can quickly become unwieldy when system size is large. Another approach entails describing a process in terms of positions of a handful of atoms that are displaced.<sup>19,20</sup> While this approach has been successfully employed for relatively simple systems involving very few atomic processes, the general applicability and accuracy of such an approach is not clearly understood. Later we shall provide examples of processes for which these approaches cannot capture the correct environment dependence of the process rates.

In this paper, we introduce a new off-lattice, self-learning KMC algorithm to overcome this challenge. The underlying philosophy behind the LE-KMC method is that atomic process mechanisms and their rates generally depend on the atomic environments, i.e., arrangement of atoms in a local region of the system. The LE-KMC method finds the atoms and their environments that influence the process and associated rate, and stores this information in a catalog for later use with KMC. The main purpose of this work is to develop a general and systematic framework for identifying which atoms are needed to define a process and characterizing the environment-dependent process rates. Since the number of atomic processes increases dramatically with the number of atoms involved in a process, we believe that this systematic procedure for storing the local process information makes the LE-KMC algorithm efficient without compromising on the accuracy.

The paper has been divided into the following sections. In Sec. II, we provide an overview of the LE-KMC algorithm. In Sec. III, the concept of local environment in LE-KMC is described. Approaches for efficiently finding the unique types of local environments that are observed during a LE-KMC calculation are discussed. Section IV provides details regarding atomic processes in LE-KMC, generating a catalog of LE-KMC processes, selecting a process for a move, and assigning LE-KMC processes to atoms. In Sec. V, we study diffusion in submonolayer thick Ag and Ag-Cu alloy film on Ag(001) to assess the performance of the LE-KMC method. Finally, conclusions are provided in Sec. VI.

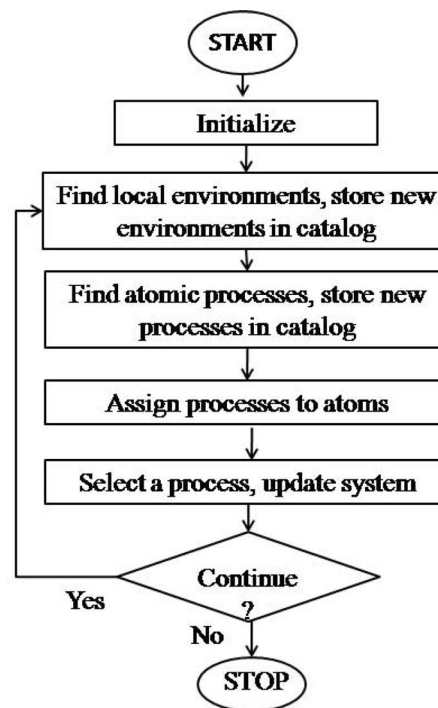


FIG. 1. Flowchart for the LE-KMC algorithm in the learning mode.

## II. THE LOCAL ENVIRONMENT KMC ALGORITHM

In this section, we provide an overview of some of the important features of the algorithm. More details will follow in later sections.

Figure 1 shows the flowchart for LE-KMC. The flowchart has some similarities to standard KMC, although many important differences are present. During the initialization step, the starting atomic coordinates, process catalog from previous calculations, and the interatomic potential information is read. As described later, long-ranged forces, such as elastic and Coulombic effects, can render the LE-KMC method (in the form presented in this work) ineffective due to large memory requirements. Hence, we assume that long-ranged effects are absent. A process catalog might not be available when the material system is being studied for the first time, however, this is not an issue since a search can be performed later for processes from states visited by LE-KMC. The next step entails finding the local environments of all atoms in the current state, classifying these environments and storing the newly found environments in a catalog (see Sec. III). Once the environments are known, LE-KMC can describe any atomic process in terms of these environments. Typically, the system is partitioned into domains so that new processes can be searched in one or more domains in parallel when multiple processors are available. After a transition has occurred, the coordinates of the domain atoms in the initial, transition and final states are obtained, and the rate  $r$  is computed assuming that the Arrhenius rate expression,<sup>24</sup>

$$r = \nu e^{-E_a/k_B T}, \quad (1)$$

is valid. Here  $\nu$  is the prefactor, which is set to  $10^{13} \text{ s}^{-1}$  in this work,  $E_a$  is the activation barrier from the process, which is

obtained using the climbing-image nudged elastic band (CI-NEB) method,<sup>25</sup>  $k_B$  is the Boltzmann constant and  $T$  is the system temperature. The LE-KMC algorithm identifies the process atoms, i.e., those atoms that play a key role in determining the process rate (see Sec. IV A). By comparing this process to the ones already stored in the catalog in terms of the process atom information, their local environments and the rate information, it is determined whether the process found is new (see Sec. IV D). Newly found processes are added to the catalog.

In principle, a variety of techniques, such as accelerated molecular dynamics (AMD)<sup>1-5</sup> and minimum-mode following methods<sup>26-28</sup> can be used to seek processes that are possible from a state. A major advantage of these techniques is that many of the atomic processes can be correctly found without having to guess the relevant processes. Because the use of LE-KMC with each of these process search techniques introduces subtle differences in the resulting LE-KMC algorithm and main goal of this paper is to demonstrate the basic principles of the LE-KMC algorithm, we restrict ourselves to the case where the types of processes that can occur, i.e., the process mechanism, is already known. For instance, in the examples considered in Sec. V we assume that surface diffusion occurs only via hop and exchange mechanism. Note that the process atoms and environment-dependent process rates remain unknown, which will be found by analyzing the results from the NEB calculation. Hence, we refer to the implementation in this work as the NEB-based implementation of LE-KMC.

Once the process catalog is deemed complete, processes are assigned to atoms, a process is randomly selected from a list of processes that are possible in the current state, and the system is updated (Sec. IV E). LE-KMC is most efficient when processes that are already stored in the catalog are reused several times during the dynamics. The update step for the NEB-based implementation of LE-KMC takes advantage of the fact that local environments and processes need to be modified only in a local region where the process has occurred. As a result, the update step is system-size independent. It is important to mention that in general standard AMD and minimum-mode following methods scale linearly with the system size, or worse. However, while using these methods with LE-KMC the computational cost for finding new processes will also depend only on domain size. The sequence of steps described so far are repeated until a stopping criterion is satisfied.

The algorithm given in Fig. 1 is called the learning mode of the LE-KMC method. In the second mode of operation, namely, fixed-catalog mode, LE-KMC has a complete catalog with relevant processes from all possible states that can be encountered in a LE-KMC calculation. Such a catalog would have been generated from past calculations in the learning mode. In Sec. V, we show examples of simple systems where the catalog is completely known. However, generating a catalog that is valid for all material compositions and configurations is in general extremely challenging (in many cases it might be computationally infeasible). Nonetheless, when the catalog contains the relevant processes (either in learning or fixed-catalog mode), the LE-KMC method overcomes

one of the major challenges of current KMC implementations, i.e., efficiently studying state-to-state dynamics while ensuring reasonable accuracy.

### III. LOCAL ATOMIC ENVIRONMENTS IN LE-KMC

Atoms in a material can have different types of local environments. The local environment of atoms determine the types of atomic processes possible and their rates. Classification of atoms based on their local environments is therefore the first step towards developing a self-learning KMC algorithm.

The local environment for atom  $i$  is obtained by storing the positions  $r_{ij} = r_j - r_i$  of atom  $j$  relative to  $i$ , for all atoms  $j$  within a local environment cut-off radius  $r_{env}$ . Here  $r_i$  is the coordinate of atom  $i$ . Hence, a local environment is conceptually similar to a Verlet list in MD. Figure 2(a) shows an example in 2D. The local environment is shown for the atom at the center of the circle. Atoms inside the circle are called environment atoms. Other atoms (shown in light grey) are not part of the environment. More discussion of the choice the environment cut-off radius is provided in Sec. IV B. Although past attempts employing on-lattice representations and graph theoretical methods<sup>19,20</sup> have been successful in capturing the local environments in simple systems, the use of local environments based on relative positions allows extension of self-learning KMC methods to a broader range of materials, including multicomponent systems where the exact relative positions of the atoms in the local environment can influence the processes.

#### A. Classifying the types of local environments in an off-lattice representation

The LE-KMC method begins with an empty catalog of environments when the material system is being studied for the first time. Each time the local environment for an atom is generated, it is compared to the catalog of known environments to find whether it is a new type of environment. New environments are added to the catalog. Unlike on-lattice representations where there is no off-set in the relative positions of atoms in two identical local environments, an exact match in the atomic positions is rarely observed in off-lattice representations. One of the sources of the offset in positions results from the errors that get accumulated as atomic moves are performed with KMC. As mentioned later in Sec. IV E, once a LE-KMC process is selected, atoms are displaced according to final atomic positions given by the process. The small error associated with the atomic displacement is eliminated to some extent using frequent, short energy-minimization steps. However, there is still some uncertainty associated with the atom positions due to this error.

We introduce a variable  $\delta r_{tol}$ , which gives the maximum uncertainty in atomic position from the actual state of the system, i.e., the maximum distance for any atom from its correct energy-minimized position. The maximum uncertainty in the relative distance between two atoms  $i$  and  $j$ , each having an uncertainty  $\delta r_{tol}$ , is then given by  $\Delta r_{tol} = 2\delta r_{tol}$ . Two local environments are said to match when all atoms in the first



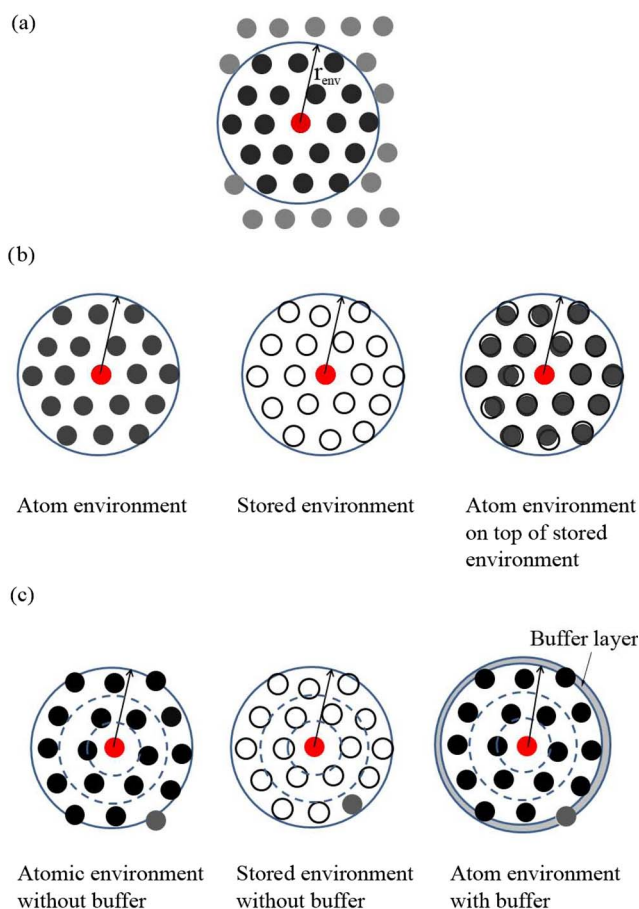


FIG. 2. (a) A 2D schematic showing how the local environment for red-colored atom is obtained using a cut-off radius  $r_{env}$ . Atoms outside the circle (sphere in case of 3D) are not part of the local atomic environment. The relative positions of the environment (black) atoms constitutes the local environment of the red atom. (b) Schematic shows qualitatively the atom-by-atom matching procedure used for comparing two environments: one belonging to an atom (left) and the other stored in the environment catalog (center). When one environment is placed on top of another (see right image) the atoms overlap within a tolerance  $\Delta r_{tol}$ , hence the two environments match. (c) Schematic shows the importance of having a buffer layer of thickness  $\Delta r_{tol}$ . Without the buffer layer the left and center environments do not match because the grey colored atom is lying outside the cut-off  $r_{env}$ . The atom environment matches with the environment stored in the catalog when a buffer layer is added (right image). The dotted lines denote the shells used with the local environment representation (see text).

environment have a corresponding atom in the second environment with the same relative position within tolerance  $\Delta r_{tol}$  (see Fig. 2(b); we call this the atom-by-atom matching procedure). Hence, when we are comparing two environments denoted as  $E_1$  belonging to atom  $i$  and  $E_2$  belonging to atom  $i'$ , for each environment atom  $j$  in  $E_1$  there should be an atom  $j'$  in  $E_2$ , which is the same atom type as  $j$ , such that

$$\|r_{ij} - r_{i'j'}\| < \Delta r_{tol}. \quad (2)$$

When  $\Delta r_{tol} = 0$ , a large number of local environments will be observed since the offset in positions is not accounted. In our calculations, we use  $\delta r_{tol} = 0.2 \text{ \AA}$ .

The computational effort required for the atom-by-atom matching will scale as  $O(N_{env}N_{envatom})$  when a local environ-

ment and  $N_{env}$  environments stored in the catalog are compared. Here  $N_{envatom}$  is the average number of atoms in an environment. Clearly, this approach becomes expensive when the number of environments  $N_{env}$  is large. Instead we employ a form of hash search to filter environments from the catalog which definitely will not match. We categorize the environments using three different keys. The first key is the chemical species type for the atom to which the local environment belongs. Note that although this key is not needed when only one atom type is present in the system, it is required for multicomponent systems. Next, as shown in Fig. 2(c) the relative positions of the environment atoms is stored in shells of width  $\delta r_{shell}$  from radius  $r = 0$  to  $r_{env}$ . The second key is given by the total number of atoms of each chemical species in  $r_{env}$ , while the third key is the number of atoms of each chemical species in each shell. Note that each of these quantities can be quickly evaluated and stored with the local environment description. When an environment  $E_1$  is being compared to the environments in a catalog, the catalog environments for which one or more keys does not match with the keys of  $E_1$  are quickly eliminated from the search. Atom-by-atom matching is performed only for a handful of environments in the catalog for which all the three keys match.

## B. Accounting for uncertainty in atom positions in the hash search

Unfortunately, the hash search algorithm provided so far cannot be employed directly because of the uncertainty in atom positions. The comparison of two identical environments  $E_1$  and  $E_2$  can result in a false negative when one/more atoms in one of the environments is/are present in the adjoining shells or outside the cut-off  $r_{env}$  resulting in the keys 2 and 3 not being matched.

By ensuring that  $\delta r_{shell} > \Delta r_{tol}$  it is guaranteed that if two environment atoms (one in  $E_1$  and other in  $E_2$ ) are to match, the shells in which these atoms reside will correspond to adjoining shells. For example, when the first atom resides in shell  $S$  of  $E_1$ , the second atom can reside in shells  $S - 1$  to  $S + 1$  in  $E_2$ . When  $\delta r_{shell} = \Delta r_{env}$  only one shell is present in the local environment.

An example of a case where an atom is outside  $r_{env}$  is shown in Fig. 2(c). In such cases, the two local environments do not match using the algorithm described so far. This issue can be addressed by adding a buffer layer of thickness  $\Delta r_{tol}$  to the cut-off  $r_{env}$ . Any atom lying within  $r_{env}$  to  $r_{env} + \Delta r_{tol}$  from the center can be a part of the local environment. The buffer atoms are added to the outermost shell. For example, in Fig. 2(c) the total number of environment atoms in the inner, middle and outer shell are 0, 6 and 12, respectively, for both the atom and stored environment when a buffer layer is used. The buffer atoms are considered in the atom-by-atom comparison only when they help in providing a match with a local environment stored in the catalog. The justification is that since the uncertainty in the relative position of a buffer atom is  $\Delta r_{tol}$ , in such cases it could have been possible that the buffer atom should have been inside  $r_{env}$ . Whenever a match is found without considering a buffer atom, then the buffer

atom is ignored by assuming that the atom would have resided outside  $r_{env}$ .

The hash search is modified in the following manner to prevent false negatives. As before, the first step involves comparing the first key, i.e., the chemical species type. The next step, i.e., comparison of the total number of atoms of each chemical species in  $r_{env}$ , is slightly modified. We begin by assuming that the two environments  $E_1$  and  $E_2$  being compared are identical. Let  $p_1$  and  $p_2$  denote the number of atoms of a particular species within a distance  $r_{env} + \Delta r_{tol}$  in environments  $E_1$  and  $E_2$ , respectively. Hence,

$$p_1 = x_1 + y_1, \quad (3)$$

where  $x_1$  is the number of atoms should actually have been outside radius  $r_{env}$  and  $y_1$  is the correct number of  $E_1$  environment atoms that should have been present inside radius  $r_{env}$  provided the uncertainty in the atomic positions in the current state was zero. Similarly,

$$p_2 = x_2 + y_2. \quad (4)$$

Let  $z_1$  and  $z_2$  be the number of atoms in the outermost shell of  $E_1$  and  $E_2$  (including the buffer layer). Furthermore,

$$z_1 = x_1 + w_1 \quad (5)$$

and

$$z_2 = x_2 + w_2, \quad (6)$$

where  $w$  gives the correct number of atoms in the outermost shell that should have been present inside the radius  $r_{env}$  when the uncertainty in atomic positions was zero. Equations (5) and (6) provide bounds for  $x_1$  and  $x_2$ , namely,

$$0 \leq x_1 \leq z_1 \quad (7)$$

and

$$0 \leq x_2 \leq z_2. \quad (8)$$

From Eqs. (7) and (8), we obtain

$$-z_2 \leq x_1 - x_2 \leq z_1. \quad (9)$$

For the two environments to be identical,  $y_1 = y_2$  (the exact value of  $y_1$  and  $y_2$  are not known). Using Eqs. (3) and (4), two environments can be identical when

$$-z_2 \leq p_1 - p_2 \leq z_1. \quad (10)$$

In other words, while accounting for the uncertainty in atomic positions, two environments can be later found to be identical even though the number of atoms  $p_1$  and  $p_2$  are different. In our implementation, we choose  $\delta r_{shell} = 3\delta r_{tol}$ , i.e.,  $\delta r_{shell} = 0.6 \text{ \AA}$ , so that  $r_{shell} > \Delta r_{tol}$ , as we required earlier, and the values of  $z_1$  and  $z_2$  are small.

The third step of the hash search involving the comparison of the number of atoms of a chemical species in each shell is also modified. The number of atoms of a chemical species in each shell is compared starting from the inner-most shell. Atoms are borrowed from the adjoining outer shell to obtain the required match for the number of atoms in the current shell (since we are using  $\delta r_{shell} \geq \Delta r_{tol}$ ). If a match is found the comparison of shells continues. The number of atoms in the adjoining outer shell is lowered appropriately when atoms

have been borrowed from it. After the comparison is over, the distribution of atoms in each shell is reset to the value prior to the comparison. For instance, the third key for the atom environment with buffer layer in Fig. 2(c) is reset to [0, 6, 12]. The modified hash search algorithm still provides an effective procedure for filtering most environments from the catalog.

### C. Steps involved in obtaining LE-KMC environment of an atom

Based on the discussion in Sec. III B, we summarize the steps for obtaining information related to the local environment  $\hat{E}$  of an atom  $i$ . The caret over  $\hat{E}$  signifies that the environment could be a known environment or a completely new environment.

1. Set the first hash key for the environment to be the species type of atom  $i$ . Set the number of environment atoms and the number of atoms in each shell to zero.
2. For all neighboring atoms that are within a cut-off distance of  $r_{env} + \Delta r_{tol}$  from atom  $i$  perform the following steps.
  - (a) Denoting the neighboring atom as atom  $j$ , store the relative coordinates  $r_{ij}$  and its atom type.
  - (b) Find the shell  $S$  in which  $j$  resides using the distance  $|r_{ij}|$  and shell thickness  $\delta r_{shell}$ . The innermost shell (from radius 0 to  $\delta r_{shell}$ ) is the first shell.
  - (c) Update the second hash key by incrementing the number of environment atoms by 1.
  - (d) Update the third hash key by incrementing by 1 the number of atoms of atom type given by the atom type of  $j$  in shell  $S$ .

### D. Steps involved in comparing two LE-KMC environments

Based on the discussion in Sec. III B, here we outline the steps involved in the comparison of two environments  $E_1$  and  $E_2$ . As an example,  $E_1$  can point to the data structure containing the local environment  $\hat{E}$  of atom  $i$  found in Sec. III C and  $E_2$  is a local environment stored in the catalog. If it is found that  $\hat{E}$  is a new environment, it is added to the environment catalog.

1. Compare the first hash key of  $E_1$  and  $E_2$ . If the two keys match go to step 2.
2. Denoting the number of environment atoms of  $E_1$  and  $E_2$  as  $p_1$  and  $p_2$ , respectively, and the number of atoms in the the outermost shells of  $E_1$  and  $E_2$  as  $z_1$  and  $z_2$  find whether Eq. (10) is satisfied. If Eq. (10) is satisfied go to step 3.
3. Perform the following comparison for each shell  $S$  of  $E_1$  and  $E_2$  starting from the innermost shells to the second last shell (because Eq. (10) is satisfied, we do not perform a comparison for the last shell):
  - (a) Check whether same number of atoms are present in the current shell  $S$  of  $E_1$  and  $E_2$ . If the numbers match go to the next shell and repeat step 3a. Otherwise, go to step 3b.

- (b) Select the environment which has fewer atoms in shell  $S$ . Move atoms from the next shell into the shell  $S$  so that the number of atoms in  $S$  matches for both environments. The number of atoms in the adjoining shell (from which atoms have been moved) have to be appropriately lowered. If a match is found in shell  $S$ , then move to the next shell and perform step 3a. Otherwise, the two environments do not match.
4. If the keys have matched so far, perform an atom-by-atom matching for  $E_1$  and  $E_2$  (see Sec. III A). For every environment atom in  $E_1$  ( $E_2$ ) that lies within cut-off distance  $r_{env}$  there should be an environment atom in  $E_2$  ( $E_1$ ) such that Eq. (2) is satisfied.
5. Set hash key values to the values at the beginning of this comparison.

$E_1$  and  $E_2$  are deemed to be identical environments if they match in the atom-by-atom comparison (step 4), otherwise they do not match.

#### IV. ATOMIC PROCESSES IN THE LE-KMC METHOD

A LE-KMC calculation will typically begin with an empty catalog of processes when the material system is being studied for the first time. Once the local environments in the current state of the system are known, atomic processes that are possible from the state are sought. When a process is found, atoms that are involved in the process, referred to as process atoms, are identified. The rate of the process is computed using Eq. (1). Next, one of the process atoms is chosen and the process is made to belong to the environment type of this atom (see Sec. IV B). Finally, it is determined whether the process is new by comparing the process information to processes already stored in the catalog (Sec. IV D). Newly found processes are added to the catalog.

Note that in LE-KMC the process information is local. Hence, a LE-KMC process could occur in more than one location of the system. As in standard KMC, which processes are possible is determined from the process information stored in the catalog. In LE-KMC terminology, we call this procedure as assigning processes to atoms. If a process is selected with KMC for a move, then the atoms to which the process is assigned are moved according to the final positions stored in the process. Subsequently, the system, environments, process catalog also need to be updated where the move has occurred.

In this section, we describe the procedure for extracting the LE-KMC process information once a transition from a state has occurred, comparing the LE-KMC process to processes in the catalog to determine whether the process is new, assigning processes to atoms, selecting processes and updating the system.

##### A. Finding the LE-KMC process information

As mentioned in Sec. II, for simplicity we assume in this work that types of atomic moves that can occur, i.e., which atoms get displaced, and the initial and final positions of the displaced atoms, are known so that the process rate information can be found using NEB. The system is partitioned into

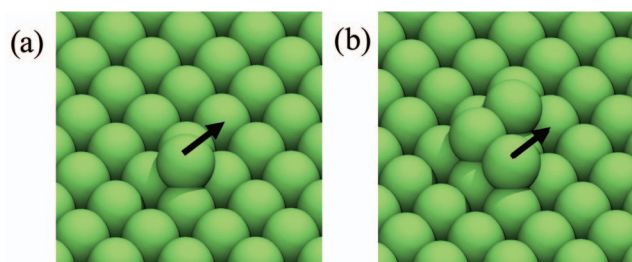


FIG. 3. Examples of two hop processes in Ag/Ag(001) where the activation barriers are different because of the difference in the local environments of the atom being displaced. The activation barrier found using CI-NEB are (a) 0.48 and (b) 0.2 eV.

domains and processes are searched in these domains in parallel. The minimum energy path from the NEB calculation is used to find the process information, i.e., the process atoms (the local environments of the process atoms are also recorded once the process atoms are known) and the process rate (using Eq. (1)). Atoms that are displaced at the end of the process automatically form part of the process atoms. As explained next, other atoms that play a role in the NEB relaxation are required to unambiguously describe a process. This ensures that processes and their associated rates are correctly assigned to atoms while reusing the process information.

Figures 3(a) and 3(b) show examples of two hop processes in Ag/Ag(001) where the difference in local environment of the displaced atoms results in a significant difference in the associated activation barriers. The activation barrier for a hop process in Fig. 3(a). All calculations performed in this work use the Ag-Cu EAM potential developed in Ref. 29. In Fig. 3(b), there are two adatoms next to the adatom which is displaced. The activation barrier for this process is 0.2 eV. If the prefactor in the Eq. (1) for the two processes were equal, the process in Fig. 3(b) is more than  $10^4$  times faster than the process in Fig. 3(a) at room temperature. Similarly other values of activation barrier are obtained as the environment of the displaced atom is modified. These environments will be observed during a LE-KMC calculation as the system moves from one state to another.

We also find examples where atoms that are not displaced after the process can influence the rates. Figure 4(a) shows an exchange process on a Ag(001) surface where one adatom displaces a neighboring surface atom to take its place. The surface atom becomes an adatom at the end of the process. The activation barrier for this exchange move is 0.6 eV. Figure 4(b) shows a closely related system, where the adatom performs an exchange move on a monolayer-thick 21 atom cluster. The activation barrier for this exchange move is 0.646 eV, which is slightly larger than the one for Fig. 4(a). Four Ag atoms have been removed from the 21-atom Ag cluster to obtain the cluster shown in Fig. 4(c). The activation barrier for exchange move on this cluster unexpectedly drops to 0.345 eV. This process is  $10^4$  times faster than the process in Fig. 4(b) at room temperature. After analyzing the NEB results, it becomes evident that the atoms around the displaced atoms relax differently depending on their local environments. For example, in Fig. 4(c) the cluster atoms can move more easily after the four Ag atoms are removed from



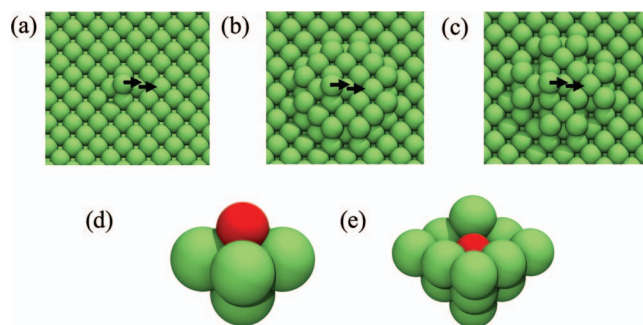


FIG. 4. Examples of exchange processes where the local environment of the atoms being displaced are the same yet the activation barriers are different: exchange move on (a) Ag(001) surface, (b) a 21-atom cluster and (c) a 17-atom cluster have barriers of 0.6, 0.646 and 0.345 eV, respectively. (d) and (e) show the local environment of the adatom and surface atom, respectively, that are displaced at the end of the process ( $r_{env} = 4.2$  Å).

the 21-atom cluster in Fig. 4(b). This results in the large drop in the activation barriers. Figures 4(a)–4(c) provides an example where although the initial local environment of the displaced atoms are the same (local environment of adatom and surface atom in the initial state are shown in Fig. 4(d) and e, respectively), the activation barriers is found to be different in each case depending on the local environment of neighboring atoms. Although we have considered only submonolayer thick films in Sec. V, the large difference in rates of processes such as those in Fig. 4(b) and 4(c) will be important when multilayer films are deposited.

Figure 5 shows another example where seemingly simple systems can give rise to unexpected process rates. In Fig. 5(a), the activation barrier for an adatom hop on the 21 atom cluster is 0.462 eV, which is slightly less than the corresponding barrier of 0.484 eV when a single adatom is present on the Ag(001) surface. The activation barrier for the hop process on the 10-atom cluster shown in Fig. 5(b) is found to be 0.47 eV, which is remarkably close to barrier in Fig. 5(a) even though the cluster is 2 atoms thick in the narrowest direction – a behavior that might appear as unexpected after observing the cluster size dependent activation barriers in Fig. 4. The process shown in Fig. 5(c) involves a jump over a step resulting in an Erlich-Schwobel barrier of 0.74 eV. The standard hop mechanism cannot occur here because the local environment would not permit such a process to take place. From the NEB calculations it is evident that the atoms close to the step edge (where the process occurs) are involved in the energy relaxation.

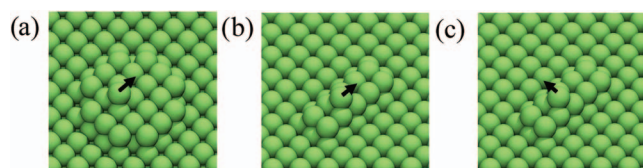


FIG. 5. Examples of hop processes on (a) 21- and (b) 10-atom cluster. The activation barriers for both processes (0.462 and 0.47 eV for a and b, respectively) is close to the activation barrier (0.484 eV) for hop process when a single adatom is present on a Ag(001) surface. (c) When an adatom jumps from the step edge, an Erlich-Schwobel barrier of 0.74 eV is encountered.

These examples demonstrate that the process mechanism and rates depend on local environments in a manner that cannot be guessed using intuition. However, some insights into this dependence can be gained by carefully analyzing NEB results. Therefore, we employ NEB-based analysis to guide the identification of LE-KMC process atoms. Process atoms are identified from the original list of hundreds to thousands of domain atoms used in the NEB calculation by performing a sensitivity analysis of the process activation barrier. For an atom  $i$  deemed to be a process atom (this does not include atoms displaced by the process, which are already considered to be process atoms), a NEB calculation is performed while preventing the atom to move during the NEB relaxation. The resulting activation barrier  $E_i$  can be different from the original activation barrier  $E_o$  for the process. When

$$|E_i - E_o| > \Delta E_{NEB}, \quad (11)$$

atom  $i$  is called a process atom. We suggest a value of  $\Delta E_{NEB} > k_B T$ . Only atoms that significantly influence the activation barrier will satisfy Eq. (11). A disadvantage of the sensitivity analysis is that it is very expensive because one NEB calculation is required for each atom in the domain to determine whether it is a process atom. Such an analysis remains extremely expensive even when atoms are chosen locally where the process has occurred (see Fig. 6(a)).

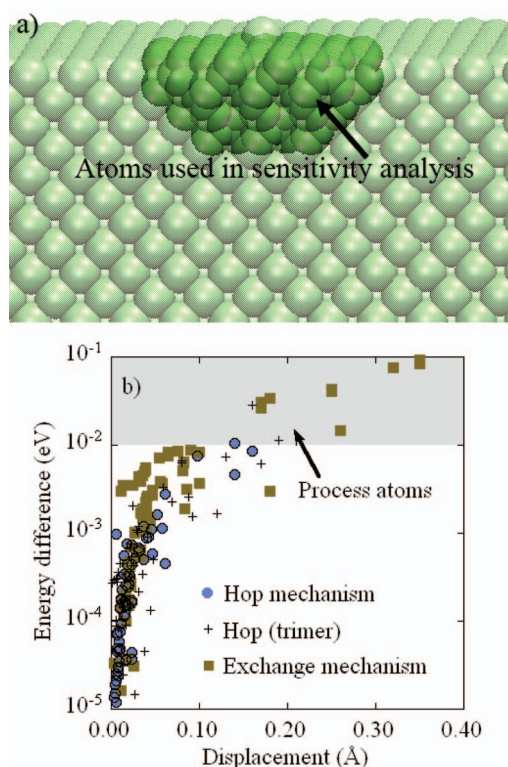


FIG. 6. (a) 106 atoms shown in dark green that were selected for the Nudged Elastic Band (NEB) sensitivity analysis of a hop and exchange moves on Ag(001) when only one adatom is present. The transparent light green atoms were not considered in the sensitivity analysis. (b) Results from sensitivity analysis of the activation barrier for three different processes. Circle, plus and square symbols correspond to the processes shown in Figs. 3(a), 3(b), and 4(a), respectively. The x-axis denotes the maximum movement of an atom during a NEB relaxation. The y-axis gives the change in the activation energy when this atom is not allowed to relax (see more details in text).



In order to find an alternate inexpensive procedure for finding the process atoms, it is important to realize that because the NEB forces are responsible for the atomic movement during energy relaxation, the movement of atoms and energy relaxation are connected (the mathematical nature of this connection is not needed in our analysis). This is demonstrated by Fig. 6(b) which shows a plot of the energy difference  $|E_i - E_o|$  against the maximum movement for different atoms considered in the sensitivity analysis. As an example, a total of 106 atoms shown in green in Fig. 6(a) were considered to generate the results shown by the circles. The x-axis is obtained by finding the maximum movement for the atom from its initial position in any NEB image while computing the original barrier  $E_o$ , i.e., this is obtained from a single NEB calculation. Results for two different hop moves and one exchange move are shown in Fig. 6(b). It is observed that the energy difference given by  $|E_i - E_o|$  in Eq. (11) can vary over several orders of magnitude. Atoms with large movement result in a large change in activation energy for both hop and exchange moves, confirming the connection between energy relaxation and the movement of the atoms in NEB.

Figure 6(b) helps us find the minimum movement  $\Delta x_{NEB}$  required to satisfy Eq. (11). Points lying in the grey region in Fig. 6(b) correspond to the process atoms for  $\Delta E_{NEB} = 0.01$  eV. Using this approach  $\Delta x_{NEB}$  was found to be 0.1 and 0.15 Å for hop and exchange moves, respectively. Once the value of  $\Delta x_{NEB}$  is known for a type of move it is assumed that the same value can be employed for other moves of the same type without requiring any additional NEB sensitivity analysis. Since a NEB calculation is anyway required for computing the barrier in Eq. (1), the position of atoms in each image are compared to their position in the initial state. Atoms which move more than  $\Delta x_{NEB}$  in any image are tagged as the process atoms. By recording the local environment of each process atom, the role that the atoms around the process atoms in the NEB relaxation is fully captured. The remaining details of other process information stored is given in Sec. IV D.

Figures 7(a)–7(c) show the number of process atoms from the process shown in Fig. 3(a) using three different values of  $\Delta x_{NEB}$  that are chosen in an ad hoc manner. As expected, the number of process atoms increases as  $\Delta x_{NEB}$  becomes smaller. Figures 7(d)–7(f) show similar results when an exchange move in Fig. 4(a) is considered. In general, a large number of process atoms (small  $\Delta x_{NEB}$ ) is not desirable. Since LE-KMC processes are classified by comparing the local environments of the process atoms, increasing the number of process atoms could result in a dramatic increase in the number of processes with a concomitant increase in memory requirements. For example, if changes occur in the environment of a process atom with a net movement of 0.01 Å in Fig. 7(c), the resulting hop move will be stored as a new type of process although we may find that the rates are practically unchanged. On the other hand when  $\Delta x_{NEB}$  is too large then atoms relevant to the process are not stored as process atoms, which can result in loss of accuracy of the method. Therefore, the sensitivity analysis is important to LE-KMC as it enables a rational selection of a value for  $\Delta x_{NEB}$ .

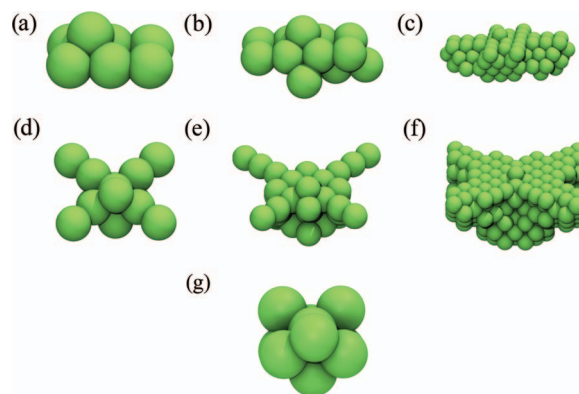


FIG. 7. Process atoms obtained using the  $\Delta x_{NEB}$  criterion for hop (a–c) and exchange (d–f) moves when one adatom is present on Ag(001) surface. The values of  $\Delta x_{NEB}$  used are (a) 0.1 Å, (b) 0.05 Å, (c) 0.01 Å, (d) 0.15 Å, (e) 0.05 Å, and (f) 0.01 Å. The corresponding number of process atoms are (a) 7, (b) 14, (c) 106, (d) 14, (e) 37 and (f) 345. Process atoms for the exchange process shown in Fig. 4(c) using  $\Delta x_{NEB} = 0.15$  Å is shown in panel g.

## B. Role of process atom environments in a LE-KMC process

As described in the examples in Sec. IV A, the maximum movement of each process atom depends on its environment. Figure 7(d) (Fig. 7(g)) shows the process atoms for processes described in Figs. 4(a) and 4(b) (Fig. 4(c)) using  $\Delta x_{NEB} = 0.15$  Å. It is observed that the initial positions of atoms in Fig. 7(g) is in fact a subset of the process atoms in Fig. 7(d). Process atoms in Fig. 7(g) are able to move more than the process atoms in Fig. 7(d) because of the differences in the local environments. If local environment of process atoms were not employed one could have mistakenly used the process in Fig. 7(g) for the system in Figs. 4(a) and 4(b). In this particular case, although the move is correct but the associated process rates would have been incorrect by several orders of magnitude. Hence, local environment of process atoms play a key role in LE-KMC.

Based on these arguments, we conclude that  $r_{env}$  signifies the neighborhood (of the process atoms) that can influence the process atom movement for a given process. A large value of  $r_{env}$  (larger than the interatomic potential cut-off, for example) will result in a more accurate KMC process catalog as the effect of far away atoms on the process rates will be captured more reliably. Unfortunately, a large environment cut-off could also result in a dramatic rise in the number of processes because when  $r_{env}$  is large, changes in the local environment of one or more process atoms would result in a completely new process. Hence, LE-KMC might have to pay a penalty of larger memory requirements in such a situation without any significant improvement in accuracy. Although the interatomic potential cut-off distance can be a possible choice for  $r_{env}$ , the connection between  $r_{env}$  and interatomic potential cut-off is not clear and would depend on the material system.

In order to understand the effect of  $r_{env}$  on the accuracy of the LE-KMC method, we performed a short KMC calculation for the system shown in Fig. 8(a) containing 20 Ag adatoms on Ag(100) surface. The catalog contained

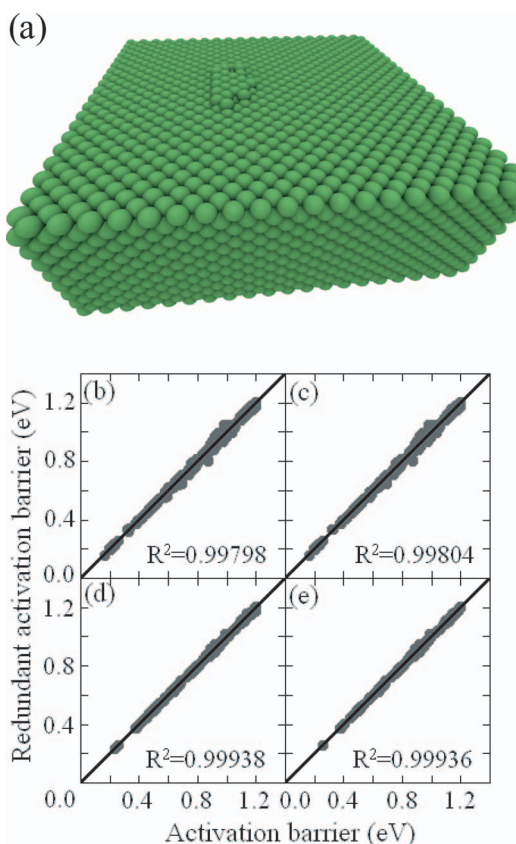


FIG. 8. (a) System containing 20 adatoms used for generating a catalog using  $r_{env} = 7.2$  Å. Activation barrier of processes that are redundant (when  $r_{env}$  is (b) 3, (c) 4.2, (d) 6 and (e) 6.6 Å) is plotted against the activation barrier of the corresponding process which is added to the catalog for the smaller  $r_{env}$  (see text). Lines in panels b-e denote linear fit. The  $R^2$  value for the fit is also mentioned.

2204 environments and 4129 LE-KMC processes at the end of 475 accepted KMC moves using  $r_{env} = 7.2$  Å. Next, a new catalog of environments and processes was generated from this catalog using a smaller value of  $r_{env}$  by ignoring atoms that reside outside the new cut-off  $r_{env}$ . This produces a catalog with a lower information content with fewer types of environments (and in turn processes as well). As one would expect, it is observed that multiple “copies” of the same process are present in the  $r_{env} = 7.2$  Å catalog when a smaller value of  $r_{env}$  is used. Since only one copy of a process can be present in a process catalog, other copies of the process were considered redundant. The activation barriers for these redundant processes were plotted along the y-axis in Fig. 8(b) while the activation barrier for the process that was present in the catalog for smaller  $r_{env}$  is plotted along the x-axis.

Figure 8(b) shows that even while using  $r_{env} = 3.0$  Å the rates from the catalog reasonably match with the original rates from the redundant rates from the  $r_{env} = 7.2$  Å catalog, even though the cut-off for the embedded atom method (EAM) potential employed in our calculations was 5.4 Å. This demonstrates that in some situations  $r_{env}$  can be smaller than the interatomic potential cut-off. The number of points in Figs. 8(b)–8(e) ranges from 872 for  $r_{env} = 6.6$  Å to 1350 for  $r_{env} = 3$  Å. The  $R^2$  value for the linear fit, which is already excellent for  $r_{env} = 3$  Å, improves marginally in

Figs. 8(b)–8(e) as  $r_{env}$  increases. The vertical distance between the points and the 45° line in Figs. 8(b)–8(e) gives the error in the activation barrier due to loss of information. A maximum error of 0.076 eV (0.036 eV) is observed for a process with 0.87 eV (0.89 eV) activation barrier using  $r_{env} = 3$  Å (6 Å). Based on these observations we have performed our LE-KMC studies in Sec. V using only  $r_{env} = 4.2$  and 6 Å.

The environment of process atoms play another role. In LE-KMC an atomic process  $P$  belongs to a particular environment type  $E_P$ .  $E_P$  is determined by the environment type of the process atom  $I$  that has the largest movement out of all process atoms. If two or more atoms have the largest movement within tolerance  $\Delta r_{tol}$ , then the atom with the smallest environment index is chosen as  $E$ . Here  $|\Delta r_I|$  is the distance by which atom  $I$  is displaced. When two or more process atoms have the largest displacement then the atom with the smallest index for the local environments used to select  $E_P$ . Recording the environment type of the process enables efficient assigning of the process to atoms because the process is possible only when an atom of environment type  $E_P$  is present (see Sec. IV E). In addition, the process rate, prefactor and activation barrier and the initial, transition and final state positions relative to atom  $I$  along with the local environment of all process atoms are stored.

### C. Steps involved in finding the LE-KMC process information

Based on the discussion in this section, next we outline the steps involved in extracting the LE-KMC process information from a NEB calculation.

1. Compute the process rate from the activation barrier obtained from NEB using Eq. (1) and record the activation barrier.
2. Tag atoms that are displaced after the process as process atoms.
3. Find the process atom  $I$  to which the process belongs and the environment type  $E_P$  of the process using the procedure described in Sec. IV B.
4. If  $\Delta x_{NEB}$  is not known for the type of move that has occurred, perform sensitivity analysis calculation described in Sec. IV A and find  $\Delta x_{NEB}$ .
5. Tag all atoms that have moved by  $\Delta x_{NEB}$  or more during NEB as process atoms.
6. For all process atoms perform the following steps:
  - (a) Record the local environment type of the process atom.
  - (b) Record the initial positions of all process atoms relative to  $r_I$  (i.e., record  $r_j - r_I$  where  $j$  is a process atom and  $I$  is the atom chosen in step 3). Similarly, record the relative coordinates of all process atoms at transition state and final state.

As described next, the steps 1, 3 and 6 are needed for comparing two processes. Information collected in step 6 is also used for updating the atom positions when a KMC move has occurred.

## D. Comparing two LE-KMC processes

When a LE-KMC process has been obtained from a NEB calculation, it is determined whether the process is new by comparing it to existing processes in the process catalog. Comparison of two processes denoted as  $P_1$  and  $P_2$  involves the following three steps. First the environment type of processes  $P_1$  and  $P_2$  are compared. Next, it is checked whether the difference in the activation barriers for the two processes is less than  $\Delta E_{NEB}$  (chosen to be 0.01 eV in our calculations). Finally, it is checked whether for every process atom in  $P_1$  ( $P_2$ ) there is a corresponding process atom in  $P_2$  ( $P_1$ ) with the same relative initial, transition and final state positions within distance  $\Delta r_{tol}$ , such that the environment type of the atoms match. Processes  $P_1$  and  $P_2$  are deemed to be identical when a match is found for all three steps.

## E. Assigning processes to atoms, selecting processes, and updating the system

Assigning processes to atoms is akin to a pattern recognition problem. One of the main advantages of LE-KMC is that once a process is known it can be assigned to atoms in several different locations where the process is possible. The LE-KMC algorithm can unambiguously determine whether a process can be assigned to an atom since the process information contains all the details needed to obtain a “pattern.”

A process is assigned to an atom  $i$  by first comparing the environment type  $E_P$  of the process and atom  $i$ . Comparison of the environment types helps filter many of atoms in the system for which the process is not possible. When the environment types match, it is checked whether all other process atoms with the correct relative initial positions are present around atom  $i$  within tolerance  $\Delta r_{tol}$ . During this step, it is also ensured that the local environments of the process atoms match. This ensures that when a process  $P$  is possible for a particular group of atoms, only one process of type  $P$  will be present for the group, and that the sum of rates from the state will be correctly obtained.

Processes from the catalog that are possible, i.e., they have been assigned to one or more atoms, are termed active; otherwise they are inactive. One active process is chosen for a move using the n-fold algorithm.<sup>6</sup> The system moves to a new state using the relative final positions stored in the process information. This step is accompanied with an update of atomic environments and process catalog using the procedure described in Sec. III and IV. However, the update is performed only the domain and adjoining domains where the process has occurred. Processes that have been newly discovered are assigned to all atoms in the system wherever possible.

## V. ASSESSING THE PERFORMANCE OF LE-KMC

In this section, the LE-KMC method is applied to the study of surface diffusion in Ag and Ag-Cu submonolayer thick films on Ag(001). The number of local environments and processes, role of the local environment cut-off, CPU and memory requirements, and reuse of catalogs with LE-KMC is studied.

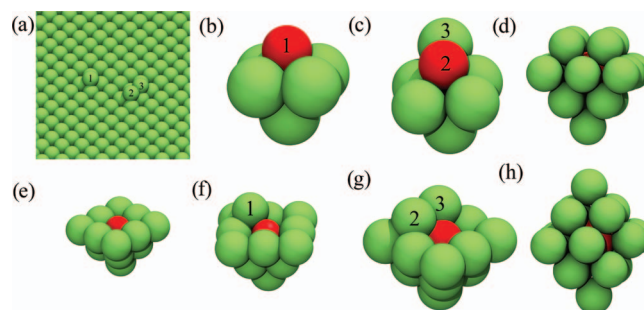


FIG. 9. (a) 3 adatoms on Ag(001). Unique local environments for this system is found using LE-KMC. b–h) Examples of some of the environments obtained with  $r_{env} = 4.2$  Å. In each case the local environment belongs to the atom shown in red color. (b) and (c) are examples of adatom environments, (e) and (g) are examples of surface atom environments, (d) is an example of a sub-surface atom environment and (h) is an example of a bulk atom environment.

Figure 9(a) shows three Ag adatoms on a Ag(001) surface. Some of the unique local atomic environments found for this state are shown in Figs. 9(b)–9(h) for  $r_{env} = 4.2$  Å. For example, the adatom marked as 1 has an environment (see Fig. 9(b)) that is different from the environment (Fig. 9(c)) for the adatom marked as 2. The LE-KMC algorithm is able to identify different types of adatom, surface atom, sub-surface atom and bulk atom environments, which is the first step towards obtaining the correct environment dependence of processes. Figure 10 shows some of the environments for the same state when  $r_{env} = 6$  Å. A total of 12 and 42 environments are observed for  $r_{env} = 4.2$  and 6 Å, respectively. The largest number of atoms in any local environment is 18 and 54, for  $r_{env} = 4.2$  and 6.0 Å, respectively. As expected, the number of environments and the number of environment atoms in local environment increase with increasing value of  $r_{env}$ . Figure 10(e) shows the atoms in different colors based on their local environment type. For instance, all surface atoms that are more than  $r_{env}$  distance away from the adatoms have the same color.

It is expected that as the system visits various states of the system, the arrangement of atoms will change and the number of environments will increase. Figures. 11(a)–11(b) demonstrates this behavior for LE-KMC calculations performed for 5 different systems with varying number of

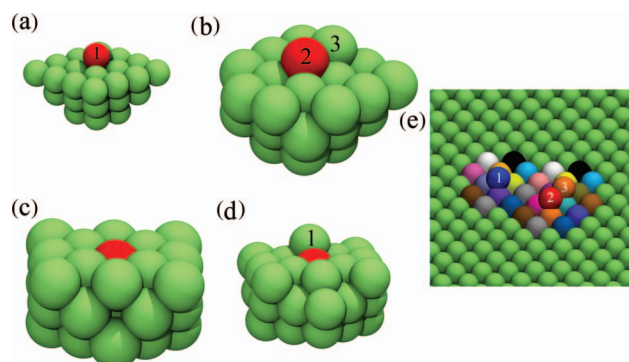


FIG. 10. (a)–(d) Different local environments found using LE-KMC with  $r_{env} = 6.0$  Å for the system shown in (Fig. 9(a)). (e) Atoms are colored according to their environment type.



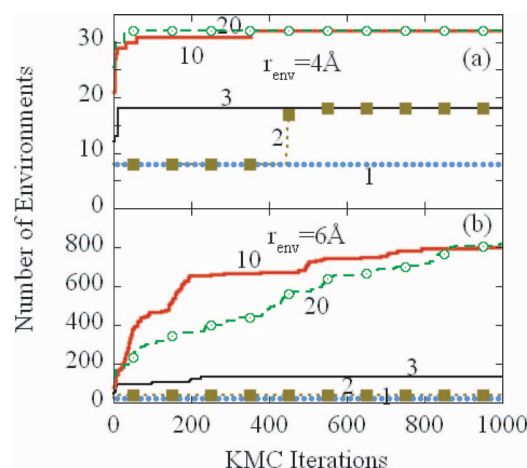


FIG. 11. Number of local environments found during the course of LE-KMC simulations where  $r_{env}$  is (a) 4.2 and (b) 6.0 Å. Five different systems with varying number of adatoms (1, 2, 3, 10 and 20) on Ag(001) surface are studied. The number of adatoms are also mentioned in the plots. The starting state for the 20 adatom case is shown in Fig. 8(a).

adatoms from 1 to 20 using  $r_{env} = 4.2$  and  $6.0 \text{ \AA}$ . The starting state for the 20 adatom case is shown in Fig. 11(c). In each LE-KMC calculation, 1000 iterations were performed at 800 K. Such a high temperature was chosen so that the system can sample more states by selecting high barrier processes with a greater probability. Consider the case where  $r_{env}$  is 4.2 Å (Fig. 11(a)). When one adatom is present, LE-KMC is able to find all environments possible for the system in the first iteration. In the two adatom case, the adatoms are initially far away from each other. Hence, the number of environments is same as the number of environments in the one adatom case. Subsequently, when the two adatoms come close to each other, additional environments are discovered. The number of environments increases when more adatoms are present, however, the increase is marginal even for 20 adatoms. Significantly more number of environments are observed when  $r_{env}$  is 6 Å (Fig. 11(b)). This indicates that the LE-KMC method can become memory-intensive when the material system of interest requires a large environment cut-off.

Figure 12 shows the number of processes for the systems studied in Fig. 11. The number of processes increases when new local environments are discovered or when the environments for process atoms are modified. This behavior is clearly observed for the 20 adatom system with  $r_{env} = 4.2 \text{ \AA}$ . Many new environments are observed during the first few KMC iterations (see Fig. 11(a)), which results in a rapid increase of new processes (see Fig. 12(a)). Subsequently, the number of environments gets saturated, however, new processes are continually discovered as the atoms rearrange themselves to form new combinations of known local environments. A similar behavior is observed for  $r_{env} = 6 \text{ \AA}$ . Just as in the case of the number of local environments, more processes are now possible when  $r_{env}$  is larger. Interestingly, for the 20 adatom case Fig. 12 shows a similar order of magnitude number of processes for  $r_{env} = 4.2$  and  $6.0 \text{ \AA}$  after 1000 iterations. Judging from the slope of the curve, it appears that the number of processes will keep increasing at longer times.

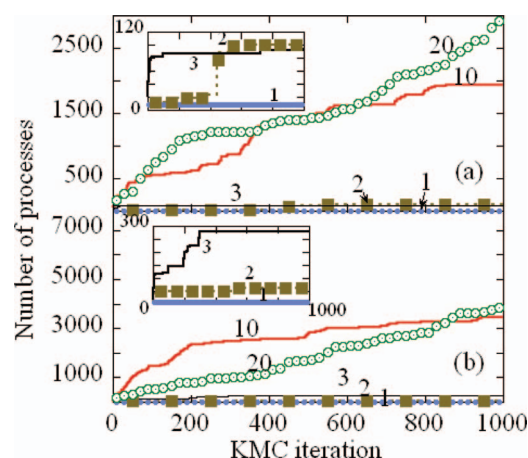


FIG. 12. Number of processes found the LE-KMC simulations in Fig. 11 where  $r_{env}$  is (a) 4.2 and (b) 6.0 Å. The insets show the number of processes for the 1–3 adatom cases over the 1000 iterations.

It appears from Fig. 12 inset that all hop and exchange processes possible for the 1–3 adatom cases are known after 800 iterations. The LE-KMC is therefore operating in the fixed-catalog mode after the initial learning phase is over. On the other hand, LE-KMC is still in the learning mode for the 10- and 20-adatom case. This raises some important questions about whether it is possible to generate a fixed-catalog from LE-KMC that can be directly used with standard KMC algorithm. When the number of environments and processes are large, a fixed-catalog will incur large computational cost while searching for processes and environments, and entail large memory requirements. Even if these processes and environments could be stored in memory, LE-KMC will find environments and processes only for states that are visited during the dynamical evolution of the system. Since the equilibrium probability of occupying states given by the Boltzmann distribution is temperature-dependent, the generated process catalog is also expected to be system temperature dependent. In principle, a comprehensive catalog of processes can be obtained by performing LE-KMC simulations with different system temperatures and initial states. However, from a practical point of view it appears that generating a fixed-catalog that can capture the environment-dependence of all processes possible for a system is infeasible. Instead the learning mode of LE-KMC should be used.

Figure 13 shows the CPU (wall time) requirements of the LE-KMC method using 10 processors. Since LE-KMC operates in the fixed-catalog mode for 1–3 adatom cases, the number of KMC iterations completed rises steeply in a short time. The computational requirements of LE-KMC is orders of magnitude larger than standard lower-accuracy lattice KMC calculations because of the frequent optimizations that were mentioned in Sec. IV A and because of the comparisons performed for environments and processes. Given the preliminary nature of this work, we believe that the efficiency of the LE-KMC implementation can be improved by up to two orders of magnitude. In subsequent publications, we shall show how such substantial speed-up can be achieved. In the learning mode, LE-KMC is even more computationally expensive because of the NEB calculations. Despite this the system size

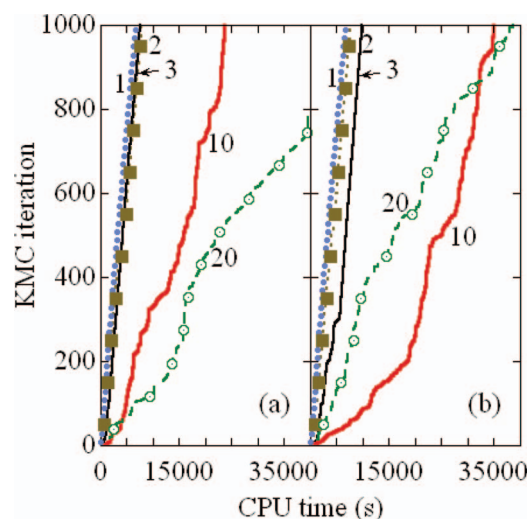


FIG. 13. Computational requirements of the LE-KMC method for the calculations performed in Fig. 11 using  $r_{env}$  as (a) 4.2 and (b) 6.0 Å.

independent cost of LE-KMC makes the method promising as mentioned earlier. The LE-KMC algorithm exhibits sudden acceleration every once a while when the local environments and process are known, and then suddenly slows down when newer environments and processes are encountered (see the 20 adatom case for example).

In Fig. 14, we perform an LE-KMC calculation with 400 adatoms (corresponding to 0.5 ML) at 800 K. The calculation is performed in three stages to demonstrate the ability of the algorithm to reuse local environment and process catalogs. The catalog generated for 20 adatom case in Fig. 11 was used as an input. The 20-adatom simulation is considered as stage 1. Next, 6500 LE-KMC iterations were performed during stage 2. The number of environments and processes found were 74 and 8297, respectively. This calculation required 2.5 days. The calculation was restarted using the catalog gener-

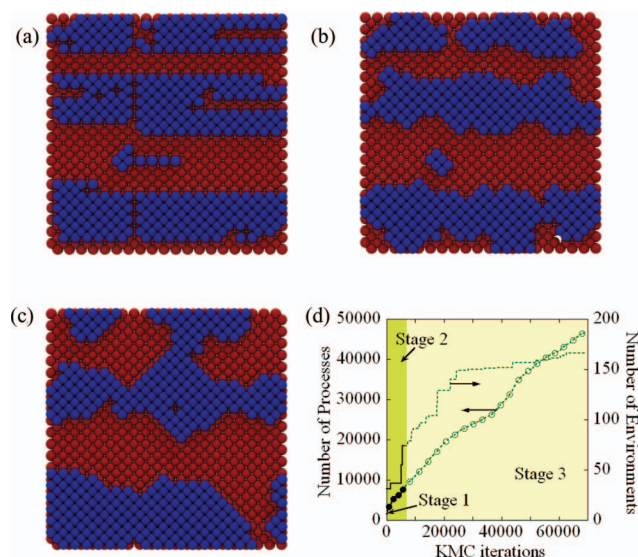


FIG. 14. (a) Starting state of 0.5 ML Ag film on Ag(001). Ag film is shown after (b) 6500 and (c) 68600 iterations. (d) Number of processes and environments found during the LE-KMC calculation of this system.

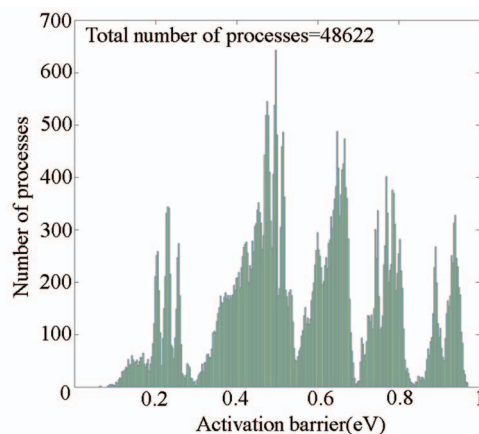


FIG. 15. Activation barrier of 48622 processes that were found for the system in Fig. 14.

ated in stage 2 and an additional 63100 iterations were performed for 18 days. The resulting catalog contained 166 environments and 48622 processes. Figure 14(d) indicates that the number of environments and processes can become extremely large when a large value for  $r_{env}$  is used, for instance, in situations where extended or long-ranged interactions are present. Although, few improvements can be made in the LE-KMC algorithm to tackle this challenge, the detailed understanding of performance of the modified LE-KMC algorithm with long-range interactions will be subject of a future study. The starting and the end states of the system are shown in Figs. 14(a)–14(c). It is observed that the system exhibits the well known time scale separation problem where low barrier processes are selected frequently resulting in small time steps (as shown next). An advantage of our LE-KMC implementation is that the software code can be combined with other methods to develop a general-purpose accelerated KMC code for overcoming the time scale separation problem in a variety of materials.<sup>30</sup>

Fig. 15 shows a histogram for the activation barrier of 48622 processes that were recorded in the LE-KMC catalog of Fig. 14. The minimum and maximum activation barriers are found to be 0.06 and 0.97 eV. Assuming that the prefactor for all processes are the same, 15 orders of magnitude time scales are spanned for the process rates at room temperature. The broad peaks observed in the histogram demonstrate the complex nature of the role of local environments on the activation barriers. It is inferred that slight changes in the local environments must have resulted in the large variation within each of the peaks.

Finally, we study diffusion in Ag-Cu monolayer thick film deposited on Ag(001) at temperature 800 K. The thickness of the Ag-Cu alloy film is 0.495 ML. Fig. 16(a) shows the initial state of the system, while Fig. 16(b) shows the state after 1000 LE-KMC iterations. It is assumed that only hop process mechanism is present. Unlike standard KMC where it would have been challenging to predict barriers for this multi-component system, rates are accurately computed using NEB with LE-KMC without having any prior knowledge of the environment dependence of processes. The number of environments and processes observed in the calculation are shown

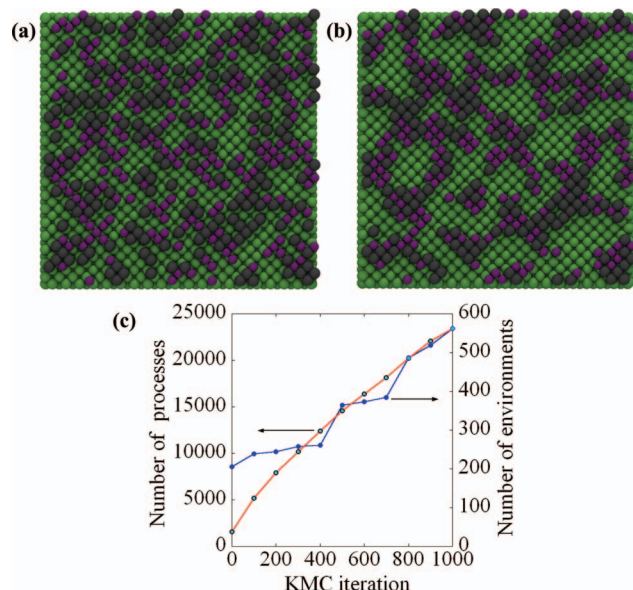


FIG. 16. (a) Starting state of 0.495 ML Ag-Cu alloy film on Ag(001). Ag film is shown after (b) 1000 iterations. (c) Number of processes and environments found during the LE-KMC calculation.

in Fig. 16(c) using  $r_{env} = 4.2 \text{ \AA}$ . Since there are more ways of arranging atoms in a multicomponent system, the number of environments and processes are significantly larger than those we have observed for pure Ag examples. Although 562 environments and 23410 processes are observed, the memory requirements required for storing all environments and processes is less than 160 MB memory because of the local nature of the information stored. This constitutes as a major advantage in terms of memory requirements over standard AMD and minimum-mode following methods where all atoms in a state are stored. By extrapolating the data it appears that the number of processes will saturate at approximately 45000 processes. In such a case all the processes can still be stored in memory. When the size of the catalog exceeds some user defined maximum size, for e.g., 100000 processes, the processes and local environments that have not been observed in the calculation in the recent past can be deleted from the catalog to reduce its size. In such situations the LE-KMC will always operate in the learning mode.

## VI. CONCLUSIONS

In this paper, we have described a new off-lattice, self-learning KMC algorithm called the local environment kinetic Monte Carlo (LE-KMC) method. LE-KMC is designed to overcome many of the limitations of standard KMC implementations, such as employing an on-lattice representation, incorrect environment-dependence of rates, fixed process catalog without allowing the possibility to learn new processes, and using an incomplete process catalog. Although, we have obtained the correct environment-dependence of rates by assuming the types of process mechanisms that are possible for the examples considered in this work, LE-KMC can be easily combined with process search methods, such as accelerated molecular dynamics and minimum-mode following methods,

enabling it to generate on-the-fly a very accurate process catalog that contains all relevant processes for visited states of the system. The LE-KMC algorithm exploits the fact that for several materials systems the atomic processes and their associated rates depend only on the local environment of atoms that participate in the process. LE-KMC proceeds to recognize the atomic environments and describe the processes in terms of these environments at every KMC iteration. Processes are stored in a catalog so that they can be retrieved and reused with the KMC algorithm later during a calculation or for future calculations involving the same materials system. This feature significantly reduces the computational cost associated with guaranteeing an accurate catalog. The LE-KMC algorithm described in this paper is computationally more efficient than standard implementations of accelerated molecular dynamics and minimum-mode following methods because only local information is stored making the LE-KMC method system size independent. As a demonstration of the method, we have studied the evolution of sub-monolayer Ag and Ag-Cu alloy film on Ag(001) for which the diffusion mechanisms are well understood, and show how the environment dependence of processes is reliably captured with LE-KMC. It is observed that although the number of environments and processes stored in the catalog can become large, the memory requirements for examples that were studied in this work are smaller than the memory typically available on present-day personal computers.

## ACKNOWLEDGMENTS

This work was supported by Indian Institute of Technology Kanpur Start-up grant IITK/CHE/20100105. We acknowledge helpful discussions with Arthur F. Voter, Blas Uberuaga and Swati Bhattacharya.

- <sup>1</sup>A. F. Voter, *Phys. Rev. Lett.* **78**, 3908 (1997).
- <sup>2</sup>A. F. Voter, *Phys. Rev. B* **57**, 13985 (1998).
- <sup>3</sup>M. R. Sorensen and A. F. Voter, *J. Chem. Phys.* **112**, 9599 (2000).
- <sup>4</sup>R. A. Miron and K. A. Fichthorn, *J. Chem. Phys.* **119**, 6210 (2003).
- <sup>5</sup>A. F. Voter, F. Montalenti, and T. C. Germann, *Ann. Rev. Mat. Res.* **32**, 321 (2002).
- <sup>6</sup>A. Bortz, M. Kalos, and J. Lebowitz, *J. Comp. Phys.* **17**, 10 (1975).
- <sup>7</sup>D. Gillespie, *J. Comp. Phys.* **22**, 403 (1976).
- <sup>8</sup>A. F. Voter, in *Radiation Effects in Solids* (Springer, Netherlands, 2007), pp. 1–23.
- <sup>9</sup>A. Chatterjee and D. G. Vlachos, *J. Computer-Aided Mater. Des.* **14**, 253 (2007).
- <sup>10</sup>G. Antczak and G. Ehrlich, *Surf. Sci. Rep.* **62**, 39 (2007).
- <sup>11</sup>J. Ferron, R. Miranda, and J. J. de Miguel, *Phys. Rev. B* **79**, 245407 (2009).
- <sup>12</sup>M. C. Tringides, *Surface Diffusion: Atomistic and Collective Processes* (Springer, New York, 1997).
- <sup>13</sup>A. F. Voter, *Phys. Rev. B* **34**, 6819 (1986).
- <sup>14</sup>G. H. Gilmer and P. Bennema, *J. Appl. Phys.* **43**, 1347 (1972).
- <sup>15</sup>M. R. Wilby, D. D. Vvedensky, and A. Zangwill, *Phys. Rev. B* **46**, 12896 (1992).
- <sup>16</sup>C. Saravanan, F. Jousse, and S. M. Auerbach, *Phys. Rev. Lett.* **80**, 26, 5754 (1998).
- <sup>17</sup>M. Schroeder, P. Smilauer, and D. E. Wolf, *Phys. Rev. B* **55**, 10814 (1997).
- <sup>18</sup>G. Henkelman and H. Jónsson, *J. Chem. Phys.* **115**, 9657 (2001).
- <sup>19</sup>O. Trushin, A. Karim, A. Kara, and T. S. Rahman, *Phys. Rev. B* **72**, 115401 (2005).
- <sup>20</sup>F. El-Mellouhi, N. Mosseau, and L. J. Lewis, *Phys. Rev. B* **78**, 153202 (2008).
- <sup>21</sup>K. Reuter and M. Scheffler, *Phys. Rev. B* **73**, 045433 (2006).



- <sup>22</sup>K. Honkala, A. Hellman, I. N. Remediakis, A. Logadottir, A. Carlsson, S. Dahl, C. H. Christensen, and J. K. Nørskov, *Science* **307**, 555 (2005).
- <sup>23</sup>L. Xu and G. Henkelman, *J. Chem. Phys.* **129**, 114104 (2008).
- <sup>24</sup>G. H. Vineyard, *J. Phys. Chem. Solids* **3**, 121–127 (1957).
- <sup>25</sup>G. Henkelman, B. P. Uberuaga, and H. Jónsson, *J. Chem. Phys.* **113**, 9901 (2000).
- <sup>26</sup>G. Henkelman and H. Jónsson, *J. Chem. Phys.* **111**, 7010 (1999).
- <sup>27</sup>L. J. Munro and D. J. Wales, *Phys. Rev. B* **59**, 3969 (1999).
- <sup>28</sup>R. Malek and N. Mousseau, *Phys. Rev. E* **62**, 7723 (2000).
- <sup>29</sup>P. L. Williams, Y. Mishin, and J. C. Hamilton, *Model. Simul. Mater. Sci. Eng.* **14**, 817 (2006).
- <sup>30</sup>A. Chatterjee and A. F. Voter, *J. Chem. Phys.* **132**, 194101 (2010).