# A MULTISCALE METHOD FOR EPITAXIAL GROWTH[*]

YI SUN[†], RUSSEL CAFLISCH[‡], AND BJORN ENGQUIST[§]

**Abstract.** In this paper we investigate a heterogeneous multiscale method (HMM) for interface tracking and apply the technique to the simulation of epitaxial growth. HMM relies on an efficient coupling between macroscale and microscale models. When the macroscale model is not fully known explicitly or not accurate enough, HMM provides a procedure for supplementing the missing data from a microscale model. Here we design a multiscale method that couples kinetic Monte-Carlo (KMC) simulations on the microscale with the island dynamics model based on the level set method and a diffusion equation. We perform the numerical simulations for submonolayer island growth and step edge evolutions on the macroscale domain while keeping the KMC modeling of the internal boundary conditions. Our goal is to get comparably accurate solutions at potentially lower computational cost than for the full KMC simulations, especially for the step flow problem without nucleation.

**Key words.** heterogeneous multiscale method, epitaxial growth, interface tracking, level set method, kinetic Monte-Carlo, island dynamics

**AMS subject classifications.** 65C20, 74S20, 74S60, 82D25

**DOI.** 10.1137/090747749

**1. Introduction.** Epitaxial growth is a technique whereby a thin film grows on an underlying crystalline substrate as the result of deposition of new material onto that substrate. Since an epitaxial film can (at least in principle) grow as a single crystal without grain boundaries or other defects, this method produces crystals of the highest quality [12]. However, epitaxial growth is challenging to mathematically model and numerically simulate because of the wide range of length and time scales that spans from the atomistic to the continuum [13, 18, 14].

There are a number of existing numerical methods in the literature. Kinetic Monte-Carlo (KMC) simulation is a popular and well-established way of studying epitaxial growth. This stochastic model simulates the growth on an atom-by-atom basis using probabilistic rules to govern deposition, diffusion, and other growth processes. But this tool is limited to small length scale simulations on the order of a micron. As a result, the Burton–Cabrera–Frank (BCF) model [4] has gained some popularity for larger scale simulations when combined with the level set method [11, 6, 19]. Schulze, Smereka, and E [22, 21] have developed a hybrid KMC-BCF technique, which is computationally efficient while retaining more of the stochastic details that dominate the KMC simulations (and presumably real growth) than deterministic BCF simulations. More numerical methods and simulation results can be found in [30].

[†]Courant Institute of Mathematical Sciences, New York University, New York, NY 10012. Current address: Statistical and Applied Mathematical Sciences Institute, Research Triangle Park, NC 27709 (yisun@samsi.info). The first author also acknowledges the support of IPAM, where some of this work was completed.
[‡]Department of Mathematics, California NanoSystem Institute and Institute for Pure and Applied Mathematics, University of California, Los Angeles, CA 90095 (caflisch@math.ucla.edu). The research of this author was supported in part by NSF grant EFRI-1025104.
[§]Department of Mathematics and Institute for Computational Engineering and Sciences, The University of Texas at Austin, Austin, TX 78712 (engquist@ices.utexas.edu). The research of this author was supported in part by NSF grant DMS-0714612.

In this paper we present a multiscale technique that couples the KMC simulations on the microscale with the island dynamics model based on the level set method and a diffusion equation. Our main interest is to capture the motion of the island boundary at the macroscopic level in the cases where the computed velocity of the boundary will require simulations on the microscale. It is possible to get accurate solutions over a larger length scale at potentially lower computational cost than for the full KMC simulations. On the other hand, we may keep more stochastic details about the atomic-scale mechanisms at the internal island boundary than for the full continuum BCF model. We emphasize that the current work mainly focuses on the layer-by-layer growth without accurately modeling nucleation.

Our technique is based on the heterogeneous multiscale method (HMM) [8], which provides a unified framework for designing efficient numerical methods for problems with multiple scales. When a macroscale model is not explicitly given or not accurate enough, HMM provides a general strategy for supplementing the missing information from an explicitly given microscale model. We refer the readers to a review article [9] for a coherent summary of the status of HMM and more applications.

This paper is organized as follows. We give a background of several numerical methods for epitaxial growth in section 2. First, we introduce the KMC simulations for a discrete model. Then the level set method for the island dynamics model on the continuum level is described and the general framework of HMM is also given. In section 3 we describe the details of our HMM strategy for epitaxial growth and discuss its implementation. In section 4, the results of submonolayer island growth and step edge evolution are presented. We conclude in section 5.

**2. Background.** In this section we give the background needed to develop our multiscale strategy, which combines elements of the KMC simulations and the continuum island dynamics model. First, we outline the basic elements of the KMC simulations and their implementation. Then we illustrate the island dynamics model using the level set method for tracking island boundaries. Finally, we introduce the general framework of HMM.

**2.1. KMC simulations.** A popular discrete model of epitaxial growth is the so-called solid-on-solid (SOS) model, in which the epitaxy is described by an array of columns of atoms, with the requirement that no vacancy sites may exist within the epitaxy (Figure 2.1). This means that each deposited atom is sitting on top of another atom; then the surface is characterized by a single-valued function—the heights of the columns $h(x, y)$ of the substrate array coordinates $(x, y)$.

For the dynamical growth rules of the model, it is necessary to make some assumptions. We usually suppose that the motion of individual atoms takes place instantaneously and that the motions are independent and Markovian. Atoms hop from site to site with a frequency proportional to the rates of processes. For simplicity, the idealized model we employed is monatomic, in which the stress effect is avoided. Evaporation is also neglected, as for typical temperature ranges in which growth is carried out there is negligible dissociation flux from the surface. We consider the processes that are expected to be the most elementary in epitaxial growth [7]. The two steps in the simulation are the following: (i) deposition of atoms onto the surface and (ii) migration of adatoms on the surface. Moreover, in order to focus on the layer-by-layer growth in our simulation, we dismiss the Ehrlich–Schwoebel barriers [10, 23] at step edges which keep an atom from hopping down a step. Although in the present work we focus on single-species and monatomic crystals, our method is not limited to this particular KMC model.
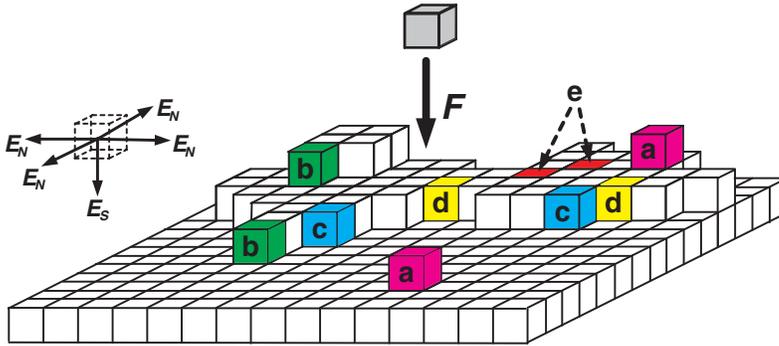
FIG. 2.1. *Schematic representation of a simple cubic nearest-neighbor SOS model. Since overhangs and bulk vacancies are excluded, the surface is fully described by an integer single-value function of the heights $h(x, y)$ on the substrate. In the KMC simulations, the deposition flux of atoms is represented by $F$, and several example atoms with different numbers of nearest neighbors are shown. For the atoms labeled by (a), their energy barriers are $E_D = E_S$; for the atoms (b), $E_D = E_S + E_N$; for the atoms (c), $E_D = E_S + 2E_N$; for the atoms (d), $E_D = E_S + 3E_N$; and for the atoms (e), $E_D = E_S + 4E_N$.*

In the deposition step [7], we generate atoms at random locations on the lattice with a time interval

$$(2.1) \qquad \Delta t = 1/FA$$

in which $F$ is the deposition flux and $A$ is the area of the substrate. The subsequent migration of surface adatoms is taken as a nearest-neighbor hopping process with its rate being given by the Arrhenius relation

$$(2.2) \qquad K(T) = K_0 \exp\left(-E_D/k_B T\right),$$

where $K_0$ corresponds to the atom vibrational frequency, $E_D$ is the hopping energy barrier, $k_B$ is Boltzmann's constant, and $T$ is the substrate temperature. The prefactor $K_0$ is usually taken either as $K_0 = 2k_B T/h$ (where $h$ is Planck's constant) or sometimes simply assigned the constant value $10^{13}\text{s}^{-1}$. The energy barrier to diffusion is calculated as the combination of the surface energy barrier $E_S$ and the bond energy $E_N$ corresponding to each nearest-neighbor atom. Thus, the barrier to hopping of an $n$-fold coordinated atom ($n = 0, 1, \ldots, 4$) is given by

$$(2.3) \qquad E_D = E_S + nE_N.$$

The barrier is assumed to depend only on the local environment of the migrating atom right before its hopping (Figure 2.1).

To summarize, the following parameters need to be given for the KMC simulations: (i) the deposition flux rate $F$, (ii) the substrate area $A$, (iii) the substrate temperature $T$, (iv) the surface energy barrier $E_S$, and (v) the nearest-neighbor bond energy $E_N$. The typical values of these parameters can be found in [7]. In the KMC simulations shown later in this paper, we set the product of $F$ and $A$ to be the number of atoms in one monolayer which are deposited onto the surface in one second, i.e., the size of the substrate. The values of other parameters are given as $T = 850\text{K}$, $E_S = 0.6\text{eV}$, and $E_N = 0.5\text{eV}$, with which there is practically no nucleation in the layer-by-layer growth.

The KMC simulations are based on the so-called rejection-free method of Bortz, Kalos, and Lebowitz (BKL) [3]. The BKL algorithm is built on the assumption that the model features $N$ independent Poisson processes with rates $r_i$ that sum to give the total rate $R = \sum_{i=1}^{N} r_i$. We use the probabilities $r_i/R$ to decide which event to execute and use $R$ to randomly select the time it takes for that event to occur from a Poisson distribution. After generating a random number $\xi \in (0, R)$, a binary search can locate the event in $O(\log_2 N)$ operations [2]. Alternately, for models based on nearest neighbors which feature a finite number of distinct rates, another fast list-based algorithm [20] is constructed with a computational complexity that is essentially independent of $N$. In the KMC simulations, we use a combination of the BKL algorithm and the algorithm in [20].

**2.2. The island dynamics model.** This model describes epitaxial growth by creating and moving island boundaries [11, 5, 6, 19]. It was developed from the BCF model [4], which is discrete in the height resolving the atomic layers in the growth direction, but coarse grained to continuum in the lateral direction. Therefore, we can couple other continuum equations for any field variable to the growth by applying the appropriate boundary conditions for the field and computing the local velocity of the island boundaries with local values of this field. Since this model requires tracking the motion of a large number of individual interfaces, such as coalescent and nucleation, we choose the level set method [17, 24, 15] for numerical implementation of this model. In particular, in a layer-by-layer growth mode, the quantitative agreement with the KMC simulations could be demonstrated [19].

In brief, the basic idea of the level set method is that an interface or boundary curve $\Gamma$ of an open region $\Omega$ can be represented as the zero level set of a function $\phi(\mathbf{x}, t)$; i.e., $\Gamma = \{\mathbf{x} : \phi(\mathbf{x}, t) = 0\}$. For a given velocity field $\mathbf{v}$, the motion of the interface $\Gamma$ can be related to the motion of the zero level set of $\phi$ via the equation

$$(2.4) \qquad \frac{\partial \phi}{\partial t} + \mathbf{v} \cdot \nabla \phi = 0.$$

The normal component of the velocity $v_n = \mathbf{n} \cdot \mathbf{v}$ contains all the physical information of the simulated system, where $\mathbf{n}$ is the outward normal of the moving boundary and $\mathbf{v} \cdot \nabla \phi = v_n |\nabla \phi|$. In this approach, the interface $\Gamma$ is captured by merely locating the set for which $\phi(\mathbf{x}, t) = 0$ (or $\phi(\mathbf{x}, t) = m$, $m = 1, 2, 3, \ldots$ for multilayer growth).

Before the boundary velocity $v_n$ is computed, we need to solve the diffusion equation for the adatom concentration $\rho(x, y, t)$ [6]

$$(2.5) \qquad \frac{\partial \rho}{\partial t} = \nabla \cdot (D \nabla \rho) + F - 2\frac{dN}{dt},$$

where $D$ is the diffusion coefficient and $F$ is the deposition flux of atoms to the surface. Here we take the flux $F$ constant on the spatial scale in our simulations, although it can be spatially varying. The length scale is of the order of the lattice constant, and the time scale is of the order of the substrate coverage.

The last term in (2.5), $-2\frac{dN}{dt}$, stands for the loss of two adatoms due to nucleation as we assume that a dimer (consisting of two atoms) is the smallest stable island; i.e., the critical island size is one. The total number of islands $N(t)$ nucleated up to a time $t$ can be computed with the following assumption that nucleation occurs at a continuous rate given by

$$(2.6) \qquad \frac{dN}{dt} = D\sigma \langle \rho^2 \rangle,$$

where the coefficient $\sigma$ is the so-called capture number for nucleation [1] and $\langle \cdot \rangle$ denotes the spatial average. When we update $N(t)$ from (2.6) during the simulation, if $N(t)$ has increased to the next integer value, a new island is seeded. This event is reflected by the appropriate modification of $\phi$ at grid points near the nucleation site. We remark that since the present work focuses on the layer-by-layer growth without nucleation, the term $-2\frac{dN}{dt}$ is dismissed in our HMM algorithm in section 3.

Then, the island boundary velocity is computed by the flux of adatoms attaching to the island boundaries, which is simply the surface diffusive flux of adatoms in the case of irreversible aggregation; i.e., $-D\nabla\rho$. Let $a$ denote the lattice constant, and $a^2$ the area per atom. The outward normal velocity is given by

$$(2.7) \qquad v_n = a^2[-D\nabla\rho \cdot \mathbf{n}] = -a^2 D \left[\frac{\partial \rho}{\partial \mathbf{n}}\right],$$

where $[\cdot]$ denotes the jump across island boundaries in the normal direction; i.e., $[f] = f_+ - f_-$ in which the subscripts $\pm$ denote the two sides of the interface with the normal $\mathbf{n}$ pointing from $(-)$ to $(+)$ and with $(+)$ denoting the upper terrace.

In order to solve the diffusion equation (2.5) for $\rho$, numerical boundary conditions need to be specified. At the outer simulation lattice boundaries, we take periodic boundary conditions for both $\phi$ and $\rho$. On the other hand, the internal island boundary condition for $\rho$ is determined by the physics of the model. The standard Dirichlet boundary condition is given by

$$(2.8) \qquad \rho(x, y, t) = \rho_b \quad \text{for} \quad (x, y) \quad \text{with} \quad \phi(x, y, t) = 0, 1, 2, \ldots$$

There are two choices for $\rho_b$. In the case of irreversible aggregation, where any adatom hitting an island boundary will attach irreversibly to that island, $\rho_b = 0$, which corresponds to the absorbing boundary condition. For reversible aggregation, $\rho_b = \rho_{\text{eq}}$, in which $\rho_{\text{eq}}$ is the adatom density for which there is local equilibrium between the step and the terrace [4]. More details of several numerical issues regarding this model can be found in [6].

**2.3. HMM framework.** A general framework of multiscale methods was proposed by E and Engquist in [8]. The name "heterogeneous multiscale method" was suggested to emphasize the multiphysics nature of the problems that it intends to handle. The following description is mostly based on the original paper [8] and the review article [9]. Suppose we are interested in the macroscopic state of a system, with state variables denoted by $U$. However, we have at our disposal only a microscopic model for the microscale state variable $u$. The variables $U$ and $u$ are linked together by a compression operator $\mathcal{Q}$ that maps $u$ to $U$, and any operator $\mathcal{R}$ that reconstructs $u$ from $U$:

$$(2.9) \qquad \mathcal{Q}u = U, \qquad \mathcal{R}U = u.$$

$\mathcal{Q}$ and $\mathcal{R}$ should satisfy $\mathcal{Q}\mathcal{R} = \mathcal{I}$, where $\mathcal{I}$ is the identity operator. Examples of $\mathcal{Q}$ and $\mathcal{R}$ were given in [8].

There are two main components in HMM (Figure 2.2):
1. Selection of a macroscopic solver. Even though the macroscopic model is not available completely or is invalid on part of the computational domain, one uses whatever knowledge that is available on the form of the macroscopic model to select a suitable macroscale solver.
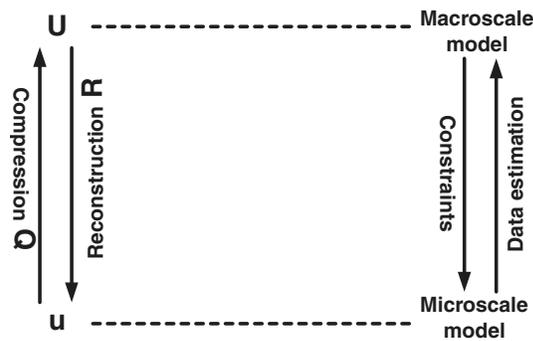
FIG. 2.2. *Schematics of the HMM framework (adapted from Figure 2 in [9]).*

2. Estimating the missing macroscale data using the microscale model. This is usually done in two steps:

(a) Constrained microscale evolution: at each point where some macroscale data is needed, perform a series of microscopic simulations, subject to appropriate constraints. The microscale solution needs to be constrained so that it is consistent with the local macroscopic state. In practice, this is often the most important technical step.

(b) Data processing: use the microscale data generated from the microscopic simulations to extract the needed macroscale data.

The data estimation can either be done in a preprocessing step as in a serial coupling method, which is often advantageous if the needed data depends on very few variables. Or, the estimation can be performed "on the fly" in a concurrent coupling method, where the microscale and the macroscale models are linked together throughout the computation. Our HMM technique for interface tracking is a concurrent coupling method.

We emphasize that HMM is not a specific method; it is a framework for designing methods. For any particular problem, there is usually a considerable amount of work, such as designing the constrained microscopic solvers, that is necessary in order to turn HMM into a specific numerical method.

In terms of interface tracking problems, the basic setup is the following [26, 27]. Suppose the macroscale process that we are interested in is described by the macroscale model with a set of variables $U$ as

$$(2.10) \qquad U_t = F(\nabla_x, U, x, t, \Gamma) \quad \text{and} \quad \Gamma_t = G(\nabla_x, U, x, t, \Gamma),$$

where $\Gamma(t)$ is a representation of the interface. The functions $F$ and $G$ are not defined or accurate everywhere. Instead we have a microscale model with the state variables $u$. The idea of HMM is to solve (2.10) with a standard macroscale method for interface tracking and to estimate the missing data such as the interfacial velocity from the microscale model. It is possible to solve the microscale model over a small region only near the interface where data estimation is carried out instead of the whole computational domain; see Figure 3.1. This is a key to the feasibility and efficiency of our approach, especially when there is a separation of the macroscopic and microscopic scales of the system.

In the next section, we will use the level set method to track the island boundaries for epitaxial growth with their normal velocities and some other information estimated from the KMC simulations.

**3. HMM for epitaxial growth.** After introducing two different numerical methods for epitaxial growth and the HMM framework in the last section, we now present how to couple them together under this framework. The KMC simulations for the discrete model are used as the microscale solver. The level set method for the island dynamics model on the continuum level is taken as the macroscale solver (Figure 3.1).
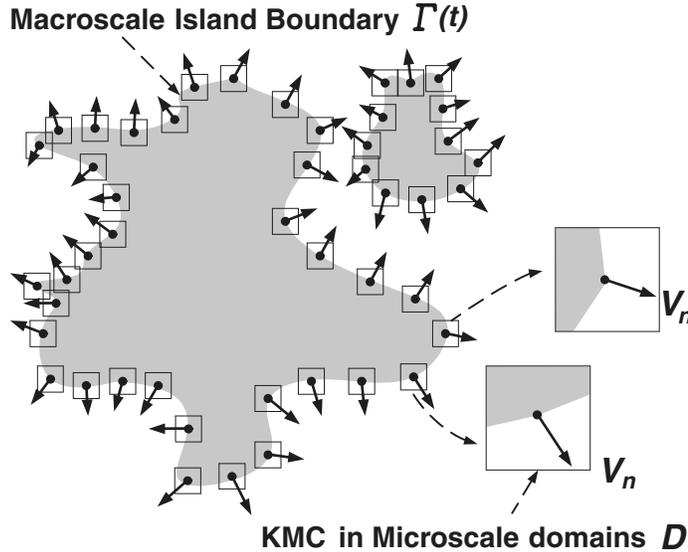
**Macroscale Island Boundary $\Gamma(t)$**



**KMC in Microscale domains $D$**

FIG. 3.1. *Schematic representation of the HMM for epitaxial growth. The KMC simulations for the discrete model are used as the microscale solver. The level set method for the island dynamics model on the continuum level is taken as the macroscale solver.*

The key part is that we will directly perform the KMC simulations in the microscale domains to estimate the normal velocity $v_n$ of the island boundary, instead of using (2.7) in the island dynamics model. Meanwhile, we also estimate the local adatom density $\rho_b$ from the KMC simulations. Here we assume that the normal velocity depends only on (a) the normal vector, and (b) the adatom density nearby. Following the two-dimensional (2-D) HMM structure for interface tracking [26, 27], we outline the algorithm of HMM for epitaxial growth as follows:

ALGORITHM 3.1. *(HMM for epitaxial growth)*

*Step 1: The island boundaries $\Gamma(t)$ are represented by the level set function $\phi$.*

*Step 2: Along the island boundaries, choose the microscale domains $\mathcal{D}$ and reconstruct the initial states for the microscale model with the local information of the level set $\phi$ and the adatom density $\rho$ from the macroscale model.*

*Step 3: Advance the microscale model via the KMC simulations for a number of microsteps in the microscale domains $\mathcal{D}$. A suitable boundary condition is necessary. Time or ensemble averaging may be used to estimate the normal velocities $v_n$ and the local adatom density $\rho_b$.*

*Step 4: Advance the macroscale model: update $\phi$ by solving (2.4); update $\rho$ by solving the diffusion equation (2.5), with the internal boundary condition $\rho = \rho_b$ which is estimated from the KMC simulations; the nucleation step is dismissed due to the layer-by-layer growth.*

*Step 5: The simulation is advanced by a macroscale time step $\Delta t$. Return to step 2.* $\square$

There are several numerical issues regarding this HMM algorithm. We present the details for several key steps as follows:

(i) *Microscale domains and their initial conditions.*

In step 2 of Algorithm 3.1, the microscale domains $\mathcal{D}$ are chosen along the island boundaries. In each macroscale cell which contains a segment of island boundaries, we set up a small box as the microscale domain; see Figure 3.2.
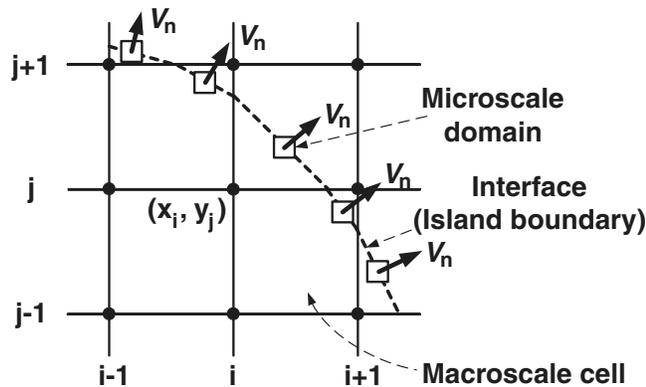


FIG. 3.2. *In each macroscale cell which contains a segment of island boundaries, we set up a small box (labelled by □) as the microscale domain.*

From the level set $\phi$, we can find the local intersections between the boundary segments and the macroscale grids by using interpolation and computing the direction of each segment, which is perpendicular to the direction of the normal velocity. To reduce the numerical errors caused by corner effects and to apply periodic boundary conditions in the direction parallel to the segment in each microscale domain, we rotate the small box to make its sides coincide with the normal direction, as the authors of [27] did in HMM for the combustion front problem (compare Figure 3.3 here and Figure 9 in [27]). Then the segment divides the box into two parts. One part is the substrate and the other is covered by one monolayer epitaxy.
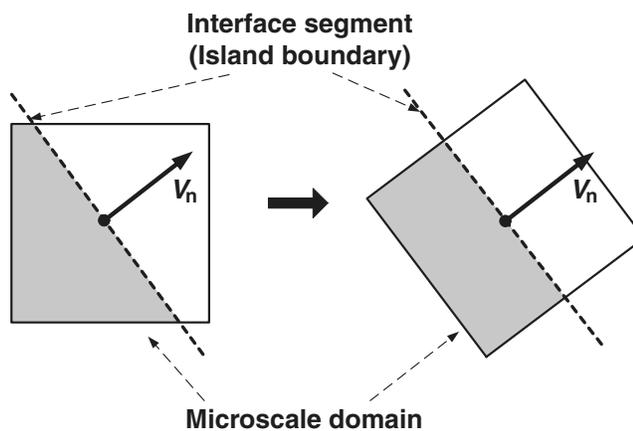


FIG. 3.3. *We rotate the small box to make its sides coincide with the direction of the normal velocity for periodic boundary conditions in the direction parallel to the segment. The region in gray represents one monolayer epitaxy.*

For the initial state of the adatoms in the microscale model, we compute the average value $\rho_0$ of the adatom density $\rho$ at four corner grid points of one macroscale cell, and multiply $\rho_0$ by the size of the microscale box to get the initial number of adatoms. Then, these adatoms are randomly distributed in the box before the KMC simulations start. Moreover, we include a possibility of roughness of the island boundary by adding atoms randomly along the boundary.

(ii) *Boundary conditions for the microscale model.*

In step 3 of Algorithm 3.1, the boundary conditions for the microscale model are needed in order to reduce finite size effects. For the two sides of the microscale domain parallel to the direction of the normal velocity, we use standard periodic boundary conditions (sides (b) and (d) in Figure 3.4). For the sides on the epitaxy and on the substrate (sides (a) and (c) in Figure 3.4), we consider a special boundary condition. If an adatom at these boundaries hops out, we let it leave the box. But we will randomly input adatoms back into the box through this side from time to time. The frequency of input or the number of influx adatoms is proportional to the average adatom density $\rho_0$ in the corresponding macroscale cell [21].
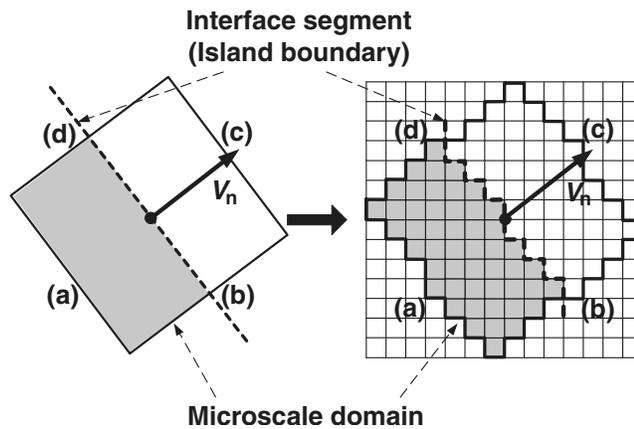


FIG. 3.4. *The small box is rotated, but the orientation of the microscale grids is fixed. This makes the boundaries in a zigzag pattern.*

The microscale box is rotated to coincide with the direction of the normal velocity. However, we should not change the orientation of the microscale grids. The reason is that the growth of epitaxy, which is solid, is supposed to have a fixed lattice orientation. This is a speciality different from computational fluid dynamics, for which we can rotate the computational grids, e.g., in the simulation of combustion fronts [27]. However, the conflict here between the rotated box and the fixed grids makes the boundaries have a zigzag pattern (right panel in Figure 3.4). Therefore, we need to carefully match the periodicity between the zigzag boundaries (b) and (d), especially for the points around the corners.

Figures 3.5 and 3.6 show several KMC simulation results of the microscale solvers. In Figure 3.5, we show a case in which the angle $\theta$ between the $x$-axis and the normal vector $v_n$ is about $40°$. The top right panel presents the zigzag boundaries and the rotated microscale box with the lattice size being 1076, which is between $32 \times 32$ and $33 \times 33$. Then the evolution of the surface morphology for the total time of $t = 0.5$s is given in the middle and bottom panels. We can clearly observe that the island boundary propagates to the northeast of the box. During the simulation, the number
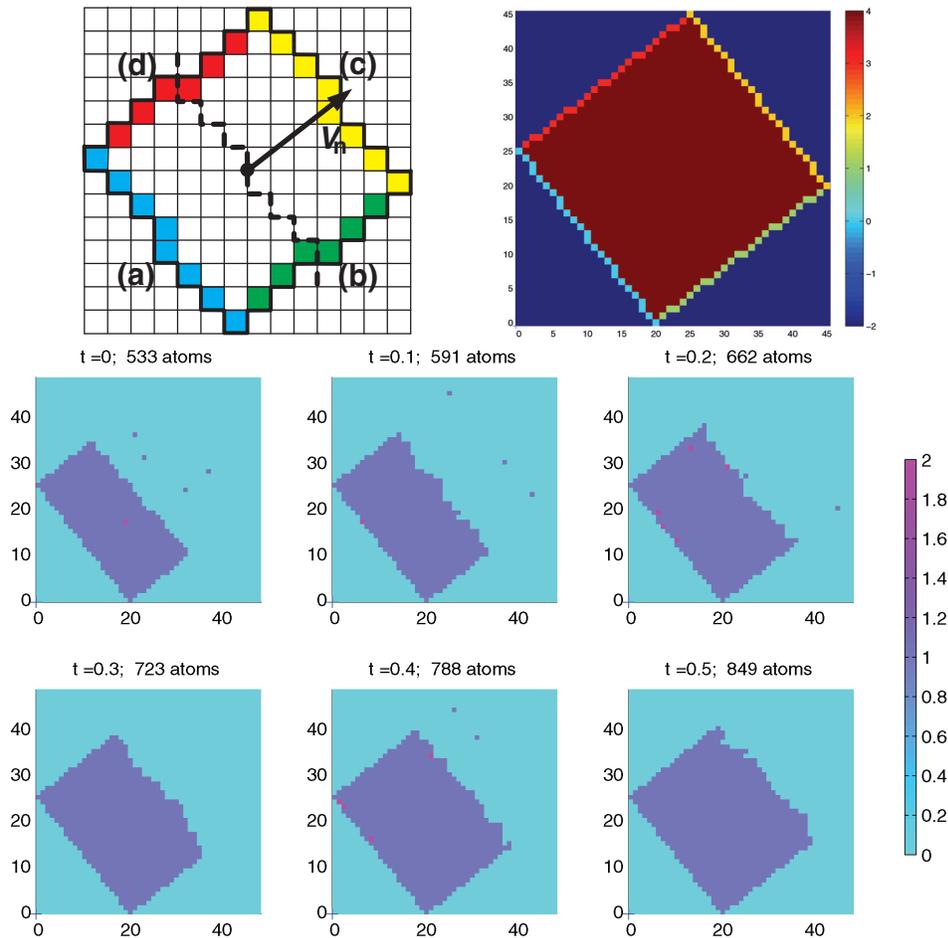
t =0; 533 atoms     t =0.1; 591 atoms     t =0.2; 662 atoms

t =0.3; 723 atoms     t =0.4; 788 atoms     t =0.5; 849 atoms

FIG. 3.5. *Top left: a schematic plot of the microscale grids. Top right: a computational plot of the microscale grids; the lattice size of the rotated microscale box is* 1076, *and the angle* $\theta$ *between the x-axis and the normal vector* $v_n$ *is about* $40°$. *Middle and bottom panels: evolution of the surface morphology in a KMC simulation at the substrate temperature* $T = 850K$ *during the total time of* $t = 0.5s$ *based on the lattice in the top right panel. There are* 316 (= 849 − 533) *atoms added, due to the deposition flux and the incoming/outgoing fluxes at sides* (a) *and* (c).

of atoms is increased by 316, which is the result due to the total number of atoms from the deposition flux and the incoming flux at sides (a) and (c) minus the number of atoms escaping from the box through sides (a) and (c). The substrate temperature $T$ is 850K.

(iii) *Estimation for the velocity of the island boundary.*

After advancing the microscale model for a short time interval, we estimate the normal velocity of the island boundary. The estimation of $|v_n|$ is given by the formula

$$(3.1) \qquad |v_n| = \frac{N(t_0 + \Delta t) - N(t_0)}{l_s \Delta t},$$

where $N(t_0)$ is the number of the atoms which form the biggest island in the first monolayer at the initial time $t_0$, $N(t_0 + \Delta t)$ is the number at a later time $t_0 + \Delta t$, and $l_s$ is the side length of the rotated simulation box. The formula (3.1) means that
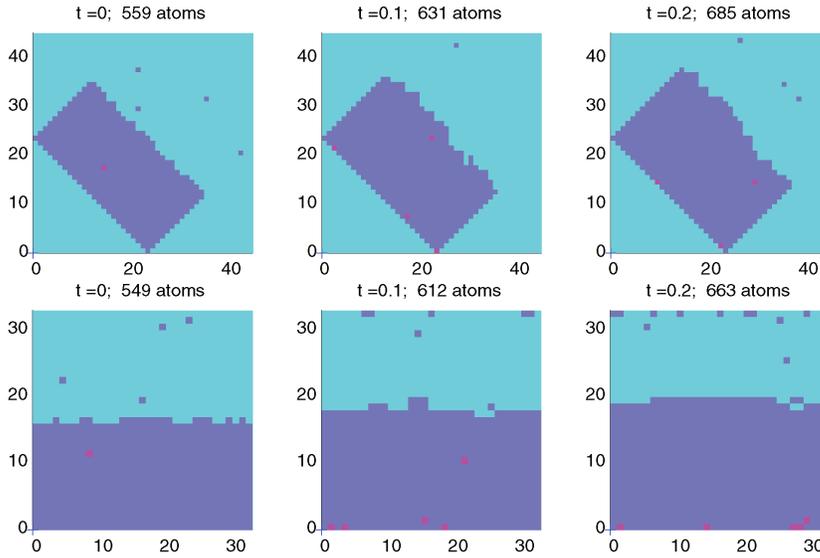
FIG. 3.6. *The angle $\theta$ between the x-axis and the normal vector $v_n$ is $\pi/4$ (upper panel) and $\pi/2$ (lower panel). The substrate temperature $T = 850K$ and the total simulation time of each result is $t = 0.2s$.*

the increased area of the biggest island (i.e., $N(t_0 + \Delta t) - N(t_0)$) divided by the side length $l_s$ gives the average distance over which the boundary has propagated. In the results shown here and the microsolvers used in the HMM tests later, the value of $l_s$ is approximately around 32 or 33, which depends on the rotation angle $\theta$.

Figure 3.6 shows two special cases, in which the angle $\theta$ between the x-axis and the normal vector $v_n$ is $\pi/4$ (upper panel) and $\pi/2$ (lower panel), respectively. In the upper panel, the island boundary is steeply zigzagged and has more kink locations, which makes the adatoms attach to the boundary more easily since they can have two bonds connecting with the nearest neighbors in kink locations. In contrast, the island boundary in the lower panel has fewer kink locations and the adatoms have only one bond to attach to the boundary, which makes it possible for the adatoms to detach from the boundary and diffuse away. Therefore, we can observe that the island boundary propagates slightly slower in the lower panel. These phenomena are also illustrated in Figure 3.7.



FIG. 3.7. (a) *Zigzag boundary case: at kink locations along the island boundary, adatoms have two bonds to connect with nearest neighbors.* (b) *Straight boundary case: adatoms have only one bond to connect with nearest neighbors at the island boundary.*

More quantitative results are shown in Figure 3.8, in which two polar plots of the estimated values of normal velocity for the total simulation time of $t = 0.2s$ and
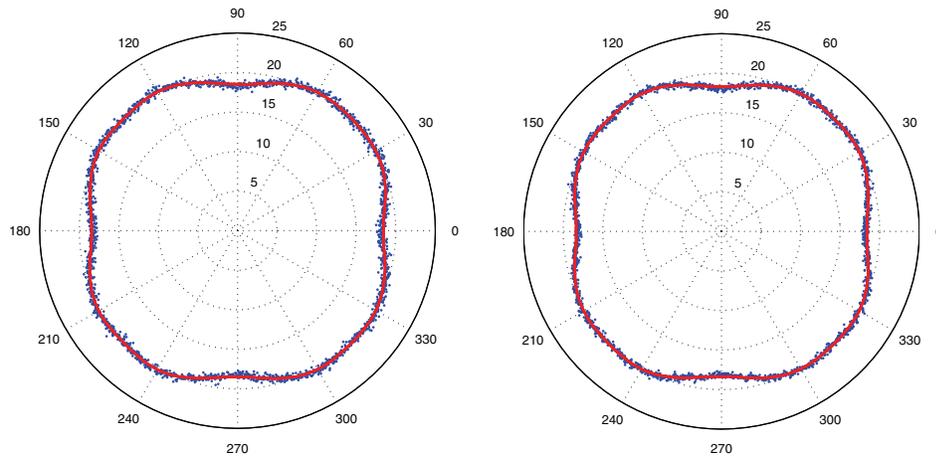
Fɪɢ. 3.8. *Polar plots of the estimated values of the normal velocity along* 2000 *different directions from* $-\pi$ *to* $\pi$. *The blue dots represent* 2000 *values. The red curve is a polynomial fit to the data points. The substrate temperature* $T = 850K$ *and the lattice size in the KMC simulations is approximately* $32 \times 32$. *Left panel: the total simulation time is* $t = 0.2s$. *Right panel:* $t = 0.3s$.

$t = 0.3$s are given in the left and right panels, respectively. In each plot there are 2000 values of $|v_n|$ (represented by blue dots) estimated from the KMC simulations with different direction angles $\theta$ ranging from $-\pi$ to $\pi$. In order to reduce statistical fluctuations, we performed 32 KMC runs with different random number seeds to get an averaged value of $|v_n|$ for each angle. The red curve is a polynomial fit to these averaged values. It is interesting to find that the fitting curve is an anisotropic Wulff shape [16, 25] due to the larger values of velocity along the directions around $\langle \pm 1, \pm 1 \rangle$ orientations. Moreover, if the simulation time is longer ($t = 0.3$s in the right panel), the anisotropic effect becomes slightly weaker. This is because the island boundary in the KMC simulations will become a little rougher at later times, even if it is straight at the beginning, so that more adatoms can attach to it. To catch this anisotropic effect and keep lower computational cost, we will perform the KMC simulations for short time ($t = 0.2$s) in the microsolver of HMM in section 4.

(iv) *Estimation for the adatom density in microscale domain.*

From Figures 3.5 and 3.6, we find that sometimes there are few adatoms at the end of each simulation. At some moments there is not even one adatom. How to calculate the local adatom density $\rho_b$ in this situation is quite a difficult issue, and is a hot topic in multiscale simulations for epitaxial growth. In the present work, we use time averaging to overcome this difficulty. In fact the average number of adatoms is typically small, but it should be invariant (on average) throughout the evolution. Therefore, we count the total number of all adatoms that have appeared during the whole simulation time $T_s$ and average it by the time, and then divide it by the substrate area $l_s \times l_s$ to get the local adatom density $\rho_b$. The formula is

$$(3.2) \qquad\qquad \rho_b = \frac{1}{l_s^2 T_s} \int_{t_0}^{t_0 + T_s} N_{adatom}(t) dt$$

in which $N_{adatom}(t)$ is the number of adatoms at time $t$.

(v) *Velocity and density extension from microscale to macroscale.*

After estimating the normal velocities of the island boundaries and the adatom densities in the microscale domains, we need to apply them to the macroscale solver in step 4 of Algorithm 3.1. For the velocities, first we extend them to the macroscale grid points bordering the island boundary.

We use Figure 3.9 to illustrate the idea. For some grid points bordering the island boundary, each of them belongs to only one cell which contains a boundary segment. We take the velocity estimated from the microscale solver for this segment as the velocity at the grid point. However, some grid points, e.g., $(x_i, y_j)$ in Figure 3.9, belong to more than one cell which contains a segment. The choice for the velocity at these grid points depends on the answer to a question: to which segment is the in-cell distance from this grid point the shortest? For example, there are three in-cell distances, namely $d_1$, $d_2$, and $d_3$, from $(x_i, y_j)$ to the segments in three different cells. We choose $v_2$ as the velocity on $(x_i, y_j)$ since the distance $d_2$ is the shortest one. A fast algorithm for computing the distance function can be found in [28]. Once $v_n$ has been defined adjacent to the interface, we extend the velocities on these bordering grid points to the whole macroscale domain by using fast sweeping techniques [29], and update $\phi$ by solving (2.4).
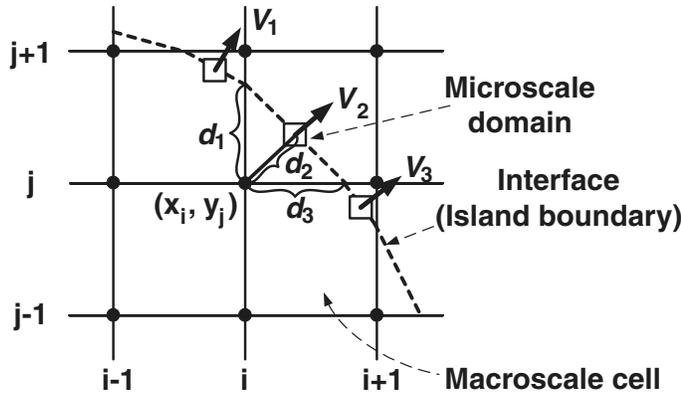


FIG. 3.9. $d_2$ *is the shortest in-cell distance from* $(x_i, y_j)$ *to the segments in three different cells.*

For the adatom densities $\rho_b$, we extend them to the macroscale intersections between the boundary segments and the macroscale grids because we need to discretize $\frac{\partial^2 \rho}{\partial x^2}$ and $\frac{\partial^2 \rho}{\partial y^2}$ on the bordering grid points and these intersections. Usually there are two segments connecting to one intersection, which means that there are two values of $\rho_b$ estimated from two different microscale domains to be chosen as the density at the intersection. Since a microscale domain for estimating the velocity of one segment may be settled down at any location along this segment, we cannot tell which one of these two microscale domains is closer to the macroscale intersection between them. Therefore, the present choice is to take the average of the two values of $\rho_b$ as the adatom density at this intersection.

For other numerical issues of Algorithm 3.1, such as time step restrictions on the macroscale, we use the same idea in the corresponding procedures of the island dynamics model. More details can be found in [6, 26].

**4. Applications.** In this section, we present two applications of the HMM interface tracking technique to epitaxial growth. We first study submonolayer island

growth, which serves as an illustrative example of the numerical method. In the second application, we use the method to track the step edge evolution.

**4.1. Submonolayer island growth.** To illustrate the procedure, we apply the HMM interface tracking technique to study submonolayer island growth without nucleation, for which we limit the number of islands to a small number, say 10. Here we can observe the islands growing, merging, and eventually covering the substrate. In Figure 4.1 the size of the macroscale grid is $128 \times 128$. At the beginning 10 small islands appear at random locations in the domain, and grow into larger islands of approximatively circular shapes. Later they develop into anisotropic shapes because the boundary velocities calculated from different microscale KMC domains depend upon the local normal **n** of the boundary, as we discussed in section 3 (Figure 3.7 and 3.8). Moreover, due to another effect from the nonuniformity of the macroscale adatom density in space, when islands approach each other they will grow more slowly than an island with no nearby neighbors since the adatom density is lower between two nearby islands. Therefore, the boundaries of islands are irregular and some nearby islands undergo deformations.
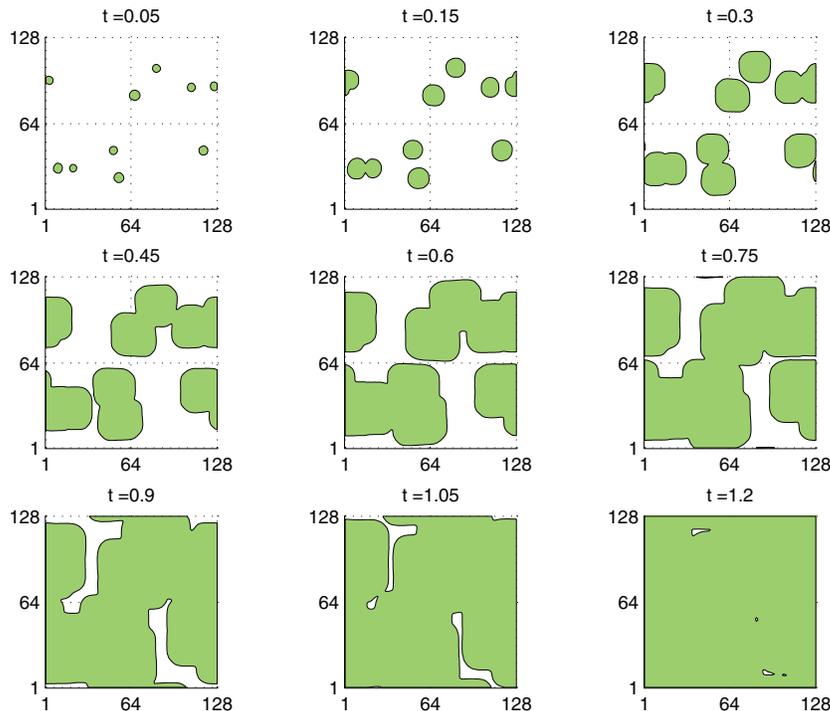


FIG. 4.1. *These plots in 2-D show the growth of 10 islands at nine different times. The HMM simulation couples the island dynamics model on the macroscale with the KMC simulations in the microscale domains along the island boundaries. The total simulation time is $t = 1.2s$. The parameters are $D = 9.8127 \times 10^9$ and $F = 1$. For the microscale grids, the lattice size of each rotated box is approximately $32 \times 32$. The size of the macroscale grid is $n \times n$ with $n = 128$. Therefore, the total size of physical substrate corresponds to $L \times L$ with $L = 4096a_0$ and $a_0$ is the atomic size.*

**4.2. Step edge evolution.** In the second application, we study the step edge evolution without nucleation. In the first example, shown in Figure 4.2, the initial
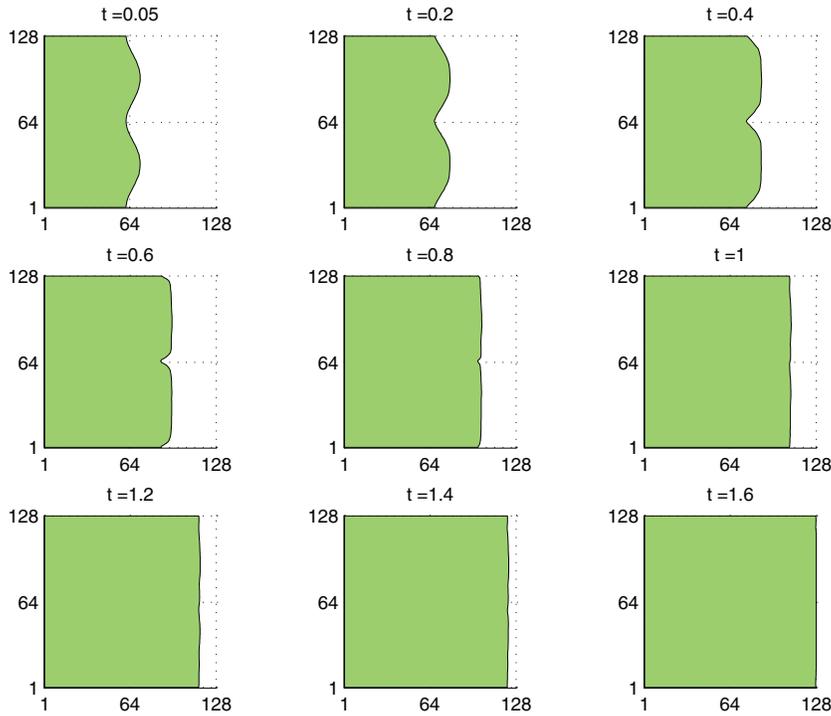
FIG. 4.2. *These plots in* 2-*D show the evolution of a step edge at nine different times. The HMM simulation couples the island dynamics model on the macroscale with the KMC simulations in the microscale domains along the edge. The initial step edge is a sinusoidal curve in the middle of the domain with the amplitude $M = 5\Delta x$ in formula* (4.1). *The total simulation time is $t = 1.6s$. The values of other parameters and the sizes are the same as in Figure* 4.1.

step edge is a sinusoidal curve in the middle of the domain, which is given by

$$(4.1) \qquad\qquad x = n/2 - M \cos(4\pi y/n),$$

where $n$ is the size of the macroscale grid and $M$ is the amplitude of oscillation. The step edge propagates from the left to the right and covers the substrate at the end. Again, the anisotropic effect makes the calculated velocities from the KMC simulations smaller in the direction parallel to the $x$-axis than those in other directions. Then, the concave parts of the edge propagate faster than the other parts and the step edge is eventually changed into a straight line.

For the last example, we consider the evolution of multilayer step edges. Figure 4.3 shows the evolution of three step edges from an HMM simulation. The initial profiles of the step edges are given by a formula similar to (4.1). In both examples the deposition rate was set to $F = 1$ monolayer per second and the diffusion coefficient to $D = 9.8127 \times 10^9$ hops per second. We use periodic boundary conditions at the macroscale boundaries of the simulation lattice for both $\phi$ and $\rho$, and a special treatment is given for $\phi$ at the left- and right-hand side of the lattice. If an edge propagates out of the lattice at the right-hand side, it will come back from the left side with the level set $\phi$ increased by the total number of initial step edges; here it is three. We again observe that the anisotropic effect makes all curved edges develop into straight lines. Since the resolution on the macroscale is $64 \times 64$, the HMM introduces jagged edges in step
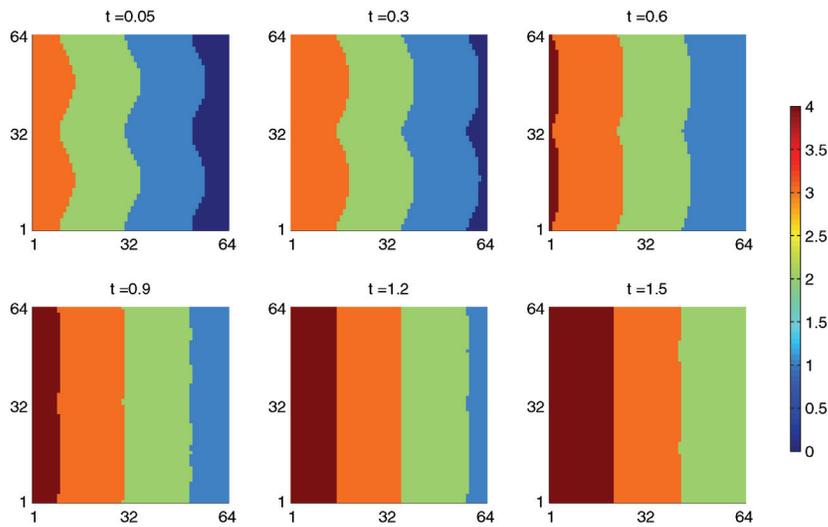
FIG. 4.3. *The six plots in* 2-*D show the evolution of the surface morphology in the HMM simulation during the total time of* $t = 1.5s$. *For the microscale grids, the lattice size of each rotated box is approximately* $16 \times 16$. *The size of the macroscale grid is* $n \times n$ *with* $n = 64$. *Therefore, the total size of physical substrate corresponds to* $L \times L$ *with* $L = 64 \times 16 = 1024a_0$.

edges. Increasing the resolution can reduce this jagged effect. But the computational cost will be increased since more KMC simulations will be needed in more microscale boxes along the step edges.

For comparison, we also show a full KMC simulation result in Figure 4.4, where 8 KMC runs with different random number seeds were performed to reduce statistical fluctuations. The lattice size of the full KMC simulation is $n \times n$ with $n = 1024$, which
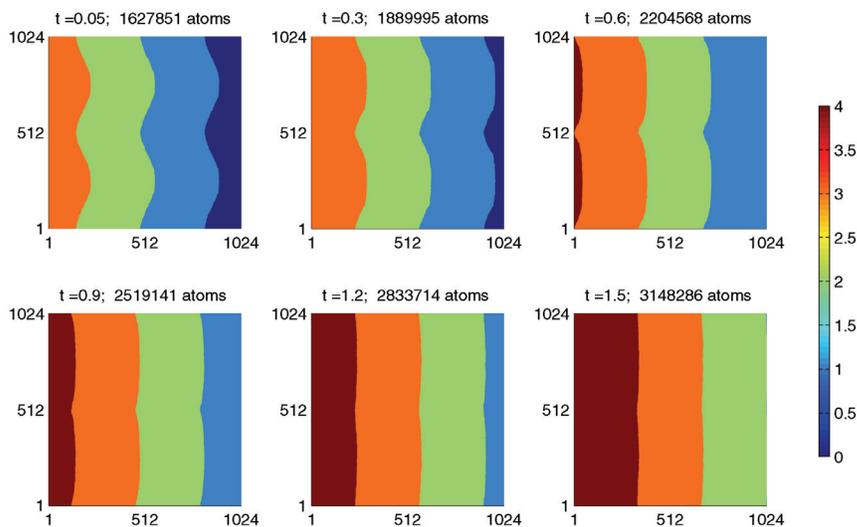


FIG. 4.4. *The six plots in* 2-*D show the evolution of the surface morphology in the full KMC simulations during the total time of* $t = 1.5s$. *The lattice size of the full KMC simulation is* $n \times n$ *with* $n = 1024$, *which makes the same corresponding size of physical substrate* $L \times L$ *with* $L = 1024a_0$.
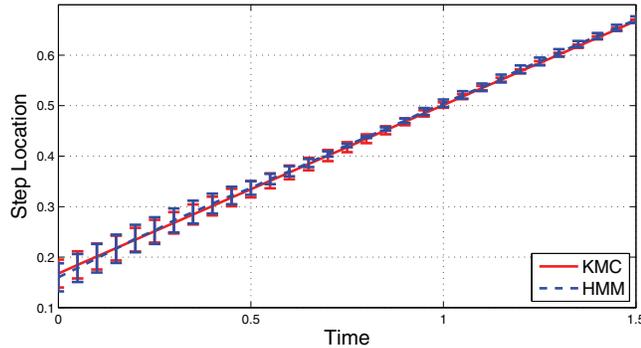
FIG. 4.5. *The evolution of the mean location of the step edge with the height of three in the KMC (solid line) and HMM (dashed line) simulations during the total time of $t = 1.5s$. The superimposed bars represent the maximum and the minimum in the horizonal location of the step edge in Figures 4.3 and 4.4. The lengths of the bars decrease with the time since the curved edges develop into straight lines. The differences between two bars together with the mean locations can be seen as the measures of errors in the HMM approximations.*
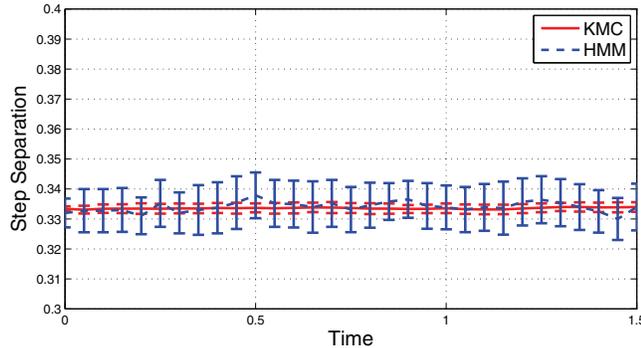


FIG. 4.6. *The evolution of the mean separation between the step edges with the height of two and three in the KMC (solid line) and HMM (dashed line) simulations during the total time of $t = 1.5s$. The superimposed bars represent the maximum and the minimum in the step separation.*

makes the same corresponding size of the physical substrate $L \times L$ with $L = 1024a_0$ and it is consistent with the full size of the HMM simulation ($64 \times 16 = 1024a_0$ in Figure 4.3). We observe that the locations of macroscale step edges from the HMM simulation basically match those of the full KMC result. Moreover, in Figure 4.5, we present the comparison result of the mean location of the step edge with the height of three (i.e., the step between the terraces in orange and green, shown in Figures 4.3 and 4.4). We can see that both of the results are in good agreement most of time. Since almost all atoms attach to the step edges, the motion of the average step edge position is mainly determined by the total mass flux. Therefore, the comparison result again indicates that the mass is generally well conserved for the step edge evolution.

In Figure 4.6, we show the comparison result of the mean separation between the step edges with the height of two and three (i.e., the width of the terrace in green, shown in Figures 4.3 and 4.4). Although there are more fluctuations in the result of HMM, both of the results show a good agreement.

In Figure 4.7, we show the evolution of the surface roughness $w$. This quantity is defined by the surface width, $w^2 = \langle (h_{i,j} - \langle h \rangle)^2 \rangle$, where the index $(i, j)$ specifies the
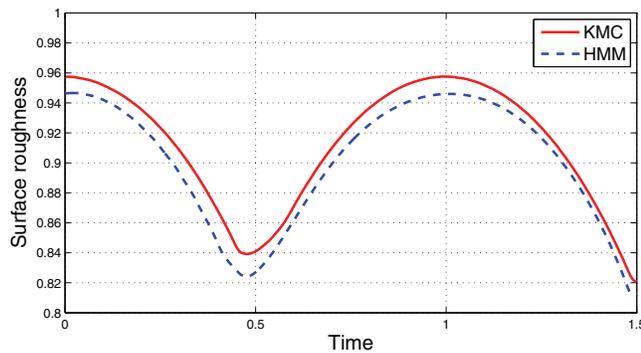
FIG. 4.7. *The evolution of the surface roughness in the KMC (solid line) and HMM (dashed line) simulations during the total time of $t = 1.5s$.*
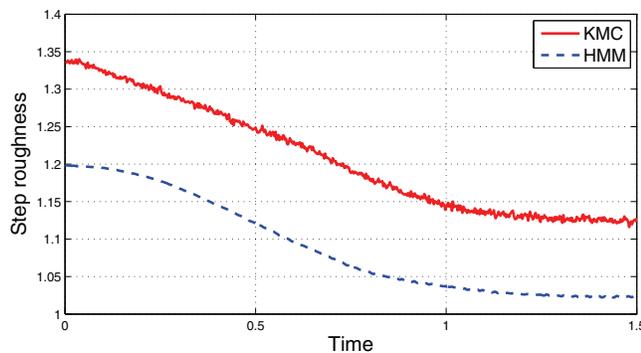


FIG. 4.8. *The evolution of the step roughness in the KMC (solid line) and HMM (dashed line) simulations during the total time of $t = 1.5s$.*

lattice site and $\langle h \rangle$ is the average of the heights. We can observe that the differences between two results are small. In Figure 4.8, we compare the step roughness $r_s$. This quantity is defined as $r_s = L_s/L$, where $L_s$ is the average length of the three step edges and $L$ is the lattice size. Although there are differences between the two results, their profiles basically agree with each other.

We run the code on a Linux platform using an Intel Pentium IV 2.3 MHz processor. For the result shown in Figure 4.3, the HMM method requires about 3 hours to run the simulation on the macroscale grid size $64 \times 64$ with the microscale lattice size $16 \times 16$ in each of the rotated microscale boxes along the step edge, whereas it takes nearly 28 hours to run a full KMC simulation on the lattice $1024 \times 1024$ in Figure 4.4. Thus, the HMM simulation is about 10 times faster than the full KMC simulation. The computational cost of the HMM simulation is dominated by the KMC simulations in the microscale boxes, so that the improvement in performance can be achieved by reducing the number of microscale domains along the step edge or the size of the microscale boxes. On the other hand, as the lattice size approaches a large value, the full KMC simulation is prohibitive due to large memory cost or very long CPU times using the same computer hardware; the HMM simulation can still be an efficient choice.

It is important to remark that there are limitations of the current form of HMM for epitaxial growth. In particular, the method does not work ideally when nucleation

takes place. The nucleation model is the one used for macroscale simulations in [11, 5, 6, 19] and thus has the same limitations. The KMC simulations mainly focus on the layer-by-layer growth by limiting the energy barrier parameters in certain regimes and by dismissing the Ehrlich–Schwoebel barriers. For example, if we take $E_S = 1.0\text{eV}$ instead of $E_S = 0.6\text{eV}$, there should be many nucleations on the terraces between steps in Figure 4.4, which the current method fails to simulate. If we include the nucleation in the island dynamics model under the HMM framework, the nucleation size for efficiency should be larger than that in original island dynamics code since here in HMM the macroscale grid is coarser. Therefore, a blind application of HMM will not be fully reliable. In summary, the current method is well suitable for the step flow problem. How to accurately simulate the nucleation process in our framework is still a challenging problem and needs to be explored in the future.

**5. Conclusions.** In this paper we have applied the HMM methodology to interface tracking in the simulation of epitaxial growth. Two numerical methods, the KMC simulation for the discrete microscale model and the level set method for the island dynamics model on the continuum scale, are described. Under the HMM framework, we design a multiscale method that couples these tools together. Our study shows that HMM is a powerful technique in the study of multiscale phenomena in complex interface problems. Our aim is to get comparably accurate solutions at potentially lower computational cost than for full KMC simulations, especially for the step flow problem. The present work gives the principles of the technique and shows the potential for future applications.

## REFERENCES

[1] G. S. BALES AND D. C. CHRZAN, *Dynamics of irreversible island growth during submonolayer epitaxy*, Phys. Rev. B, 50 (1994), 6057.

[2] J. L. BLUE, I. BEICHL, AND F. SULLIVAN, *Faster Monte Carlo simulations*, Phys. Rev. E, 51 (1995), R867.

[3] A. B. BORTZ, M. H. KALOS, AND J. L. LEBOWITZ, *A new algorithm for Monte Carlo simulation of Ising spin systems*, J. Comput. Phys., 17 (1975), pp. 10–18.

[4] W. K. BURTON, N. CABRERA, AND F. C. FRANK, *The growth of crystals and the equilibrium structure of their surfaces*, Philos. Trans. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci., 243 (1951), pp. 299–358.

[5] R. E. CAFLISCH, M. F. GYURE, B. MERRIMAN, S. OSHER, C. RATSCH, D. D. VVEDENSKY, AND J. J. ZINCK, *Island dynamics and the level set method for epitaxial growth*, Appl. Math Lett., 12 (1999), pp. 13–22.

[6] S. CHEN, M. KANG, B. MERRIMAN, R. E. CAFLISCH, C. RATSCH, FEDKIW, M. F. GYURE, AND S. OSHER, *Level set method for film epitaxial growth*, J. Comput. Phys., 167 (2001), pp. 475–500.

[7] S. CLARKE AND D. D. VVEDENSKY, *Growth kinetics and step density in reflection high-energy electron diffraction during molecular beam epitaxy*, J. Appl. Phys., 63 (1988), pp. 2272–2283.

[8] W. E AND B. ENGQUIST, *The heterogeneous multiscale methods*, Commun. Math. Sci., 1 (2003), pp. 87–132.

[9] W. E, B. ENGQUIST, X. LI, W. REN, AND E. VANDEN-EIJNDEN, *The heterogeneous multiscale method: A review*, Commun. Comput. Phys., 2 (2007), pp. 367–450.

[10] G. Ehrlich and F. Hudda, *Atomic view of surface diffusion: Tungsten on tungsten*, J. Chem. Phys., 44 (1966), pp. 1039–1099.

[11] M. F. Gyure, C. Ratsch, B. Merriman, R. E. Caflisch, S. Osher, J. J. Zinck, and D. D. Vvedensky, *Level set methods for the simulation of epitaxial phenomena*, Phys. Rev. E, 58 (1998), R6927.

[12] M. A. Herman and H. Sitter, *Molecular Beam Epitaxy: Fundamentals and Current Status* 2nd ed., Springer, Berlin, 1996.

[13] A. C. Levi and M. Kotrla, *Theory and simulation of crystal growth*, J. Phys. Condens. Matter, 9 (1997), pp. 299–344.

[14] T. Michely and J. Krug, *Islands, Mounds and Atoms. Patterns and processes in crystal growth far from equilibrium*, Springer, Berlin, 2004.

[15] S. Osher and R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*, Springer, New York, 2003.

[16] S. Osher and B. Merriman, *The Wulff shape as the asymptotic limit of a growing crystalline interface*, Asian J. Math, 1 (1997), pp. 560–571.

[17] S. Osher and J. Sethian, *Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations*, J. Comput. Phys., 79 (1988), pp. 12–49.

[18] A. Pimpinelli and J. Villain, *Physics of Crystal Growth*, Cambridge University Press, Cambridge, 1998.

[19] C. Ratsch, M. F. Gyure, R. E. Caflisch, F. Gibou, M. Petersen, M. Kang, J. R. Garcia, and D. D. Vvedensky, *Level set method for island dynamics in epitaxial growth*, Phys. Rev. B, 65 (2002), 195403.

[20] T.P. Schulze, *Kinetic Monte Carlo simulations with minimal searching*, Phys. Rev. E, 65 (2002), 036704.

[21] T.P. Schulze, *A hybrid scheme for simulating epitaxial growth*, J. Crystal Growth, 263 (2004), pp. 605–615.

[22] T.P. Schulze, P. Smereka, and W. E, *Coupling kinetic Monte-Carlo and continuum models with application to epitaxial growth*, J. Comput. Phys., 189 (2003), pp. 197–211.

[23] R. L. Schwoebel and E. J. Shipsey, *Step motion on crystal surfaces*, J. Appl. Phys., 37 (1966), pp. 3682–3686.

[24] J. A. Sethian, *Level Set Methods and Fast Marching Methods, Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision and Materials Science*, Cambridge University Press, Cambridge, 1999.

[25] R. F. Sekerka, *Equilibrium and growth shapes of crystals: How do they differ and why should we care?*, Crystal Res. Tech., 40 (2005), pp. 291–306.

[26] Y. Sun, *The Heterogeneous Multiscale Methods for Interface Tracking*, Ph.D. thesis, Princeton University, Princeton, 2006.

[27] Y. Sun and B. Engquist, *Heterogeneous multiscale methods for interface tracking of combustion fronts*, SIAM Multiscale Model. Simul., 5 (2006), pp. 532–563.

[28] R. Tsai, *Rapid and accurate computation of the distance function using grids*, J. Comput. Phys., 178 (2002), pp. 175–195.

[29] R. Tsai, L.-T. Cheng, S. Osher, and H.-K. Zhao, *Fast sweeping algorithms for a class of Hamilton-Jacobi equations*, SIAM J. Numer. Anal., 41 (2003), pp. 673-694.

[30] A. Voigt, ed., *Multiscale Modeling in Epitaxial Growth*, Birkhäuser-Verlag, Basel, 2005.