Time-dependent Fields on Rectangular Substrates: Modeling Anisotropic Diffusion and Growth

P. Yang[1] and J. A. Venables[1,2]
[1]Department of Physics & Astronomy, Arizona State University, Tempe AZ 85287-1504, U.S.A
[2]School of Science and Technology, University of Sussex, Brighton, U.K

ABSTRACT

In models of nucleation and growth on surfaces, it has often been assumed that surface diffusion is isotropic; however, many cases are known (e.g. Ge clusters or silicide nanowires grown on Si(001)2x1) when anisotropic diffusion, and possibly also attachment, would be more appropriate. Here we explore a method of using the discrete FFT for solving the time-dependent diffusion equation for ad-particles. This method is tested against the known result of diffusion from a 2D point source, with periodic boundary conditions. The FFT method gives a perfect match to the analytical solution for multiple point sources arranged on a lattice, except near (initial) field discontinuities. We solve the time-dependent anisotropic diffusion field with rectangular block sinks to simulate the growth of a regular array of islands using analogous methods. Programs are coded in Matlab®6.5; the calculation is fast, and the evolving diffusion field and island shapes can be seen visually in the form of MatLab movies. Capture numbers are calculated for rectangular islands in the case of restricted corner diffusion.

INTRODUCTION

Diffusion and capture processes involved in nucleation and growth on surfaces have received widespread attention over the last thirty-five years [1]. There is also a large literature on modeling crystal growth shapes, both in 2- and 3- dimensions (2D and 3D) [2-4]. The 3D case (volume growth) is bound to present considerable analytic and computational challenges, but the 2D case (area growth) though identical in principle, is more tractable in practice. Early solid-on-solid models typically used a square geometry, suitable for unreconstructed (001) substrates [5]. Here we examine rectangular geometry, suitable for (110) and reconstructed (001) substrates [6].

Many different computational models have been developed. As noted in a recent review [7], such models are typically either wholly deterministic (e.g. mean field rate equation (RE) or continuum partial differential equation (PDE) solutions) or wholly stochastic (e.g. Molecular Dynamics (MD) or Kinetic Monte Carlo (KMC) simulations). These models have been joined by various hybrid methods, notably Level Set (LS) [7], Phase Field and Configurational Continuum methods. Apart from REs, all these approaches involve time-consuming calculations on specialist, high-end computers. One of our aims is to show that these 2D problems can be approached on modern Personal Computers (PCs), using software that addresses matrices in a compact manner. We concentrate on the 2D Fast Fourier Transform (FFT), the use of logical masks, and the ability to make movies, as implemented in the MatLab® 6.5.

ANISOTROPIC DIFFUSION FROM A 2D POINT SOURCE

We are interested in solutions of 2D diffusion equations for the ad-particle concentration $C(x,y,t)$, when there may be sources and sinks at various positions. Here we explore a simple case, the point source with a constant amount of diffusant, and compare analytical and FFT solutions. The equation to be solved is the standard linear diffusion equation

$$\frac{\partial C}{\partial t} = D_x \frac{\partial^2 C}{\partial x^2} + D_y \frac{\partial^2 C}{\partial y^2}, \quad (1)$$

where $C(x, y, t)$ is the concentration field and D_x and D_y are diffusion coefficients. For a fixed amount of diffusant M_0 that is released at $(x_0, y_0) = (0, 0)$ at $t=0$, the known analytical solution is:

$$C(x, y, t) = M_0 \exp\left(-\left(\frac{x^2}{4D_x t} + \frac{y^2}{4D_y t}\right)\right)/4\pi D_{xy} t, \quad (2)$$

where $D_{xy} = (D_x D_y)^{1/2}$. This solution is a product of x and y 1D solutions [8].

A 2D matrix is used as the concentration field C. The matrix size is M-by-N. The point source is at (x_0, y_0). Therefore, the corresponding analytical solution at the grid points is

$$C(x, y, t) = M_0 \exp\left(-((x- x_0)^2/4D_x t + (y- y_0)^2/4D_y t)\right)/4\pi D_{xy} t, \quad (3)$$

where (x, y) is the coordinate for each element of the matrix. The point source (x_0, y_0) can be at any element of the matrix, but is most conveniently placed at the center.

It is arguably well known [9], but perhaps not generally appreciated, that the full solution to equation (1) can be obtained on a lattice of step size $a(b)$ in the $x(y)$ directions, by the use of a propagator P that is applied in Fourier (reciprocal) space, before conversion back to the original (real) space. For this problem, the propagator matrix at reciprocal lattice points (k, l) is

$$P = \left[\frac{2D_x}{a^2}\left(1 - \cos\left(\frac{2\pi(k-1)}{M}\right)\right) + \frac{2D_y}{b^2}\left(1 - \cos\left(\frac{2\pi(l-1)}{N}\right)\right)\right] \quad (4)$$

For a linear problem, such as this one, the propagator can be applied in a single step from $t = 0$ to t, yielding for the Fourier transform of C, $\overline{C_{k,l}}(t) = \overline{C_{k,l}}(0)\exp(-Pt)$. However, since we may wish to apply this to non-linear problems, with frequent updates every Δt, this is written as

$$\overline{C_{k,l}}(t+\Delta t) = \overline{C_{k,l}}(t)\exp(-P\Delta t). \quad (5)$$

This solution shows that the propagator P determines the results of $\overline{C_{k,l}}(t+\Delta t)$ based on the result at its previous time step $\overline{C_{k,l}}(t)$. Now we can apply the inverse FFT on $\overline{C_{k,l}}(t+\Delta t)$ to get the concentration field at time $(t+\Delta t)$, i.e. $C_{n,m}(t+\Delta t)$.

$$C_{n,m}(t+\Delta t) = \frac{1}{MN}\sum_{k=1}^{M}\sum_{l=1}^{N}\overline{C_{k,l}}(t+\Delta t)\exp\left(-\frac{i2\pi m(k-1)}{M}\right)\exp\left(-\frac{i2\pi n(l-1)}{N}\right) \quad (6)$$

Given the initial and boundary condition for $C_{n,m}(t)$, we can loop through the time steps Δt to calculate the concentration field at each grid point as a function of time t. In MatLab®, the Field matrix (C) can be incremented by the propagator matrix Pmat ($\exp(-P\Delta t)$ in equation 5) as

Field = ifft2(fft2(Field).*Pmat), (7)

where fft2 and ifft2 are MatLab® instructions for the 2D Fourier and inverse Fourier transforms, and .* implies element by element multiplication. For constant time steps Δt, the matrix Pmat can be calculated prior to the time loop, which saves considerable computing time [10].

To compare the FFT results with the analytical solution, we plotted the concentration field along the axes in x and y directions through the point source, as indicated in Figure 1(a). Diffusion coefficients were kept at $D_x = 5$ and $D_y = 10$, exploring moderate anisotropy. The total amount of the diffusant $M_0 = 10$ was released at the center of the matrix at time $t = 0$.

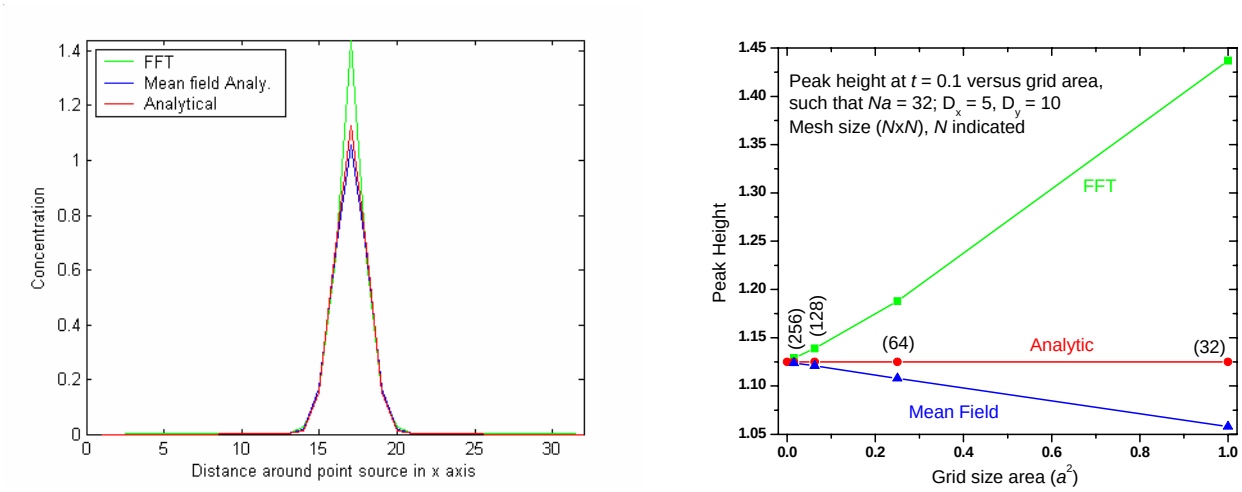


Figure 1 a) left: Concentration field after annealing a point source for $t = 0.1$ with $D_x = 5$, $D_y = 10$, on a 32x32 grid with step size $a = 1$; b) right: Peak height as function of grid area for $t = 0.1$. The "Mean Field" solution is obtained by integrating the analytic solution over each grid square.

We experimented with matrix sizes (32x32) up to (256x256), varying parameters by factors of two, as shown in Figure 1(b). The FFT solutions agree perfectly with the analytical solution except for small times and grid sizes. In these limits, we do not have enough Fourier coefficients in the solution to map out the function (3) near the initial slope discontinuity at (x_0, y_0). At long times, multiple analytic point sources are needed to compare with the FFT solution, which has periodic boundary conditions. A larger set of examples is available from the authors [11].

RECTANGULAR BLOCK SINKS, MASKS AND BOUNDARY CONDITIONS

The main purpose of this work is to develop and test simple algorithms for visualizing the growth of (crystalline) islands on rectangular symmetry substrates, maintaining the full time-dependence of the solution. We choose to study annealing of an initial field $C = 1$ into an existing block sink, on which the concentration is initially zero. Annealing is preferred over deposition for a test case, since it is known that time-dependent effects are more marked [7, 12]. It also avoids the need to consider the parameter (D/F), where F is the deposition flux, which introduces both competing length scales and the possibility of new nucleation. Deposition and new nucleation are the topic of many papers [2-5], and of ongoing current work.

Ad-particles diffuse into the sink, and this capture of particles make the island grow in size. Thus this becomes a moving-boundary problem; we control the speed of the moving boundary by assuming a particular height, h, of the island. For small h the island will grow quickly laterally, while for large h it will grow slowly, and the boundaries are quasi-stationary, as is often assumed in crystal growth calculations. Since this problem is linear, increasing h is equivalent to reducing C, often to $<< 1$, in real epitaxial growth situations. Thus monolayer growth, with $h = 1$, could equally be simulated by say $C \approx 10^{-3}$ [7, 12], or as here with $C = 1$, $h = 10^3$. Moving boundary effects are accentuated by small values of h. Here we illustrate trends using two values, $h = 5$ and 20. With non-linear effects we would need to be careful about the absolute scale of the variables.

The first point is that incorporation processes at the island edge determine the subsequent shape of the island, so the algorithm for these processes has to be determined. At the risk of reinventing the last half-century of crystal growth modeling, we can see that if there is no desorption from the edges (irreversible growth), and if edge diffusion round the island before incorporation is rapid, ad-particles incorporate as a 'rectangular shell', in which both *x*- and *y*-edges grow at the same rate. This means that, if the initial island has lengths characterized by l_x and l_y, then the difference (l_y - l_x) remains the same, and the island becomes squarer with annealing time. The island was chosen to be of size ($2l_x$+1, $2l_y$+1), which allowed islands to be placed around their center at (x_0, y_0). A key point is now how can we update l_x and l_y; as we shall see, there are many options. The reasoning for our initial choice is as follows.

In the absence of a potential field, we can assume that Fick's first law holds, so that the diffusion flux in our anisotropic 2D rectangular case, with a diagonal diffusion tensor *D*, is

$$j_x(x, y) = -D_x\partial/\partial x[C(x, y)],\ \ j_y(x, y) = -D_y\partial/\partial y[C(x, y)]. \qquad (8)$$

Numerical derivatives such as these may be difficult to calculate accurately, especially so as the grid size is decreased, which is otherwise desirable for increased accuracy. However, if we are interested only in the growth of the whole island, rather than the detailed position dependence of equation (9), then the continuity equation gives us an integral method via

$$\oint_{\mathbf{L}} \mathbf{j}.\mathrm{d}\mathbf{L} = \oint_{A} (\partial C / \partial t)\mathrm{dA}\ , \qquad (9)$$

where the line integral is over the line edge **L** of the island, and the surface integral is over the island area *A*. The flux to the island in one update time step Δt is thus expressed simply as the change in concentration ΔC integrated over the island; this method is very stable numerically.

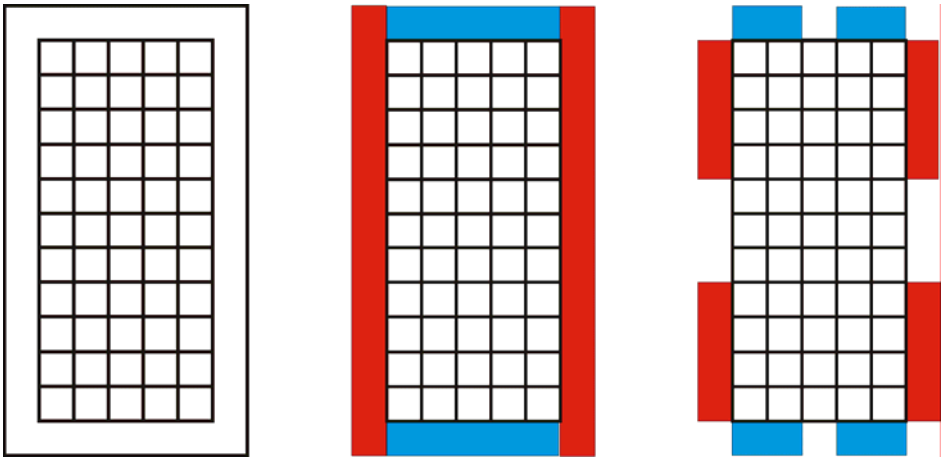
RESTRICTED CORNER DIFFUSION: MOVIES AND CAPTURE NUMBERS

The interesting cases are those where ad-particles diffuse to a limited extent, due to freezing-out of some particular processes. In the limit of particles adhering where they arrive, then we have fractal-like behavior [3]. An intermediate case that has received much attention recently [2, 4] is restricted corner diffusion. We investigated this case using logical island masks as follows.

To calculate the flux to the island using equations (8) and (9), we need to be able to attribute the arriving flux to the *x*- and *y*-sides of the island separately. The FFT solution is for the entire 2D space including the island, but the boundary conditions are reset after each time step, Δt. Thus the integrated concentration within the island during Δt is obtained as a growth increment. This island growth increment is partitioned into *x*- and *y*- growth components, using logical masks to select relevant matrix elements [10].

To monitor the growth, we need to count up how many ad-particles have arrived and to check sum rules. Using the above schemes, the diffusion field and the island shape have been visualized using standard plotting functions and have been built into movies using simple routines available within MatLab® 6.5. In initial work we have updated the island size in a line-by-line fashion, while visualizing the growth pixel-by-pixel, assuming that ad-particles that arrive migrate along the edges to the corners, but not around them, as indicated in Figure 2.

The above procedure produces very realistic faceted shapes, acknowledging that the field concentration is highest at the corners. Two examples are shown in Figure 3, where the contour plot is used to show both features: figure 3(a), for h = 20, shows moderate anisotropy in the diffusion field and smaller lateral island growth than in figure 3(b) for h = 5.



a) 5x11 island & surround b) Increment y, then x c) Increment by pixel

Figure 2: Updating island size and shape. In the examples shown here, a 5x11 island (a) grows by adding ad-particles to the surrounding ring. The calculation is incremented line by line (b), but the display is incremented pixel by pixel (c). See text for discussion.

The island has a larger area, and the anisotropy of the concentration field is even more marked. Note that there is less material in the field and more in the island in the second case, even though the diffusion constants and the diffusion time are the same. This is due to the greater capture efficiency of the larger lateral island size, consequent on the smaller island height.

The calculation of anisotropic capture numbers, σ_x and σ_y, follow straightforwardly, and graphs as a function of annealing time are shown in Figure 4. In these plots we notice the sharp initial drop, due to the time-dependence, and the "spikes" caused by the growth steps corresponding to figure 2(b). Further work is underway to implement the pixel-by-pixel scheme, figure 2(c), which should result in smoother capture number curves [11]. Quantitative studies of capture numbers are discussed in companion papers [7, 12].

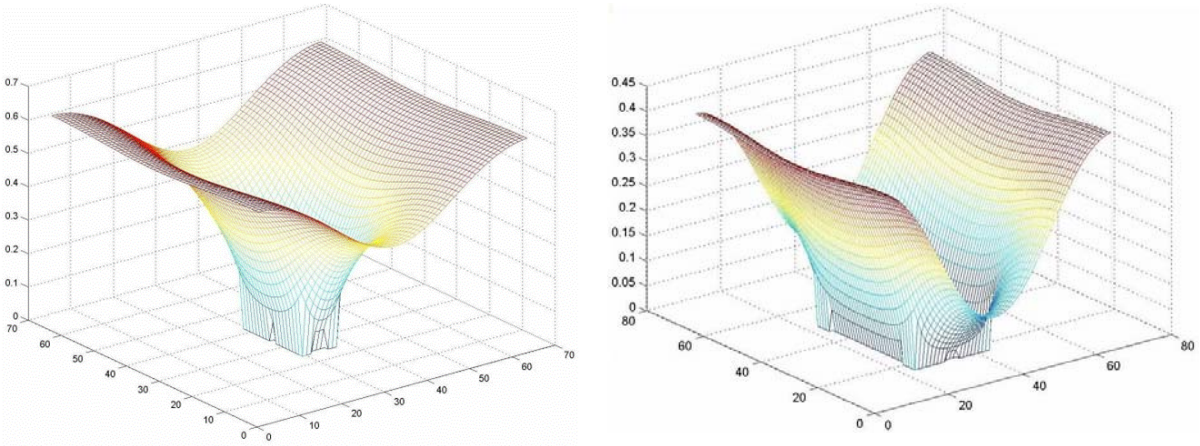


Figure 3: Contour plots of the concentration field after annealing for t = 90 (900 steps, Δt = 0.1) with D_x = 5, D_y = 10, on a 64*64 grid with step size a=b=1: a) left, h = 20; b) right, h = 5.

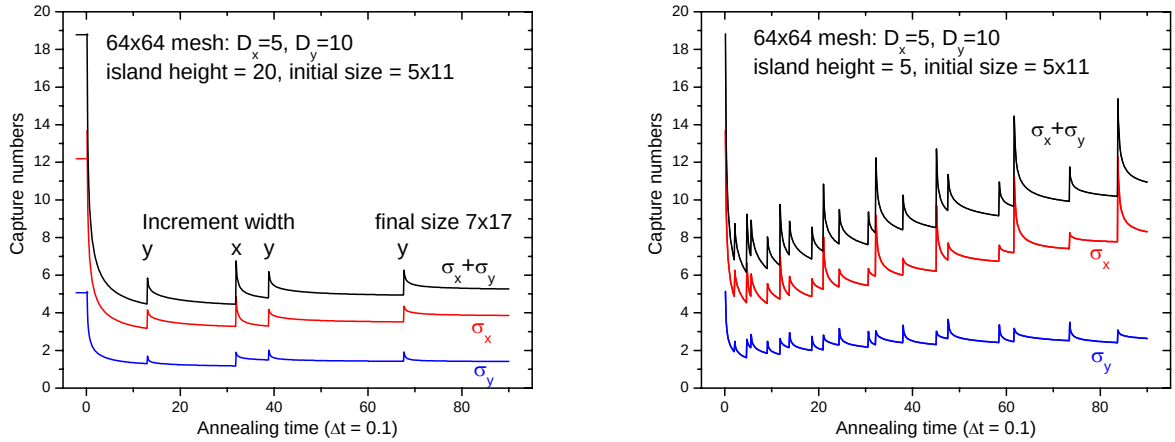


Figure 4: Capture numbers for the growth of the island: conditions as figure 4; a) left, h = 20; b) right, h = 5. The jumps in the capture numbers correspond to incrementing the island size.

REFERENCES

1. For reviews, see J.A. Venables, Surface Sci. **299**, 798 (1994); *Introduction to Surface and Thin Film Processes* (Cambridge University Press: Cambridge, 2000), especially chapter 5.
2. For facetted shapes, see J.M. Bermond & J.A. Venables, J. Cryst. Growth **64**, 239 (1983); M. Odzemir & A. Zangwill, Phys. Rev. B **45**, 3718 (1992); N. Israeli & D. Kandel, Phys. Rev. Lett. **88**, 116103 (2002).
3. For fractal shapes, see the large literature started by T.A. Witten & L.M. Sander, Phys. Rev. Lett. **47**, 1400 (1981); Phys. Rev. B **27**, 5686 (1983).
4. For recent experimental and theoretical work see H. Brune, Surface Sci. Rep. **31**, 121 (1998), H.C. Jeong & E.D. Williams, Surface Sci. Rep. **34**, 175 (1999), H.P. Bonzel, Physics Reports **385**, 1 (2003); T. Michely & J. Krug, *Islands, Mounds and Atoms: Patterns and Processes in Crystal Growth far from Equilibrium* (Springer Verlag: Berlin, Heidelberg, New York 2004).
5. For Solid-on-Solid (001) studies see e.g. J.D. Weeks & G.H. Gilmer, Adv. Chem. Phys. **40**, 157 (1979); S. Clarke, M.R. Wilby and D.D. Vvedensky, Surface Sci. **255**, 91 (1991); M. Siegert & M. Plischke, Phys. Rev. E. **50**, 917 (1994); J. Krug, Adv. Phys. **46**, 139 (1997).
6. M.C. Bartelt & J.W. Evans, Europhys. Lett. **21**, 99 (1993); M. Siegert & M. Plischke, Phys. Rev. Lett. **73**, 1517 (1994); M. Siegert, Phys. Rev. Lett. **81**, 5481 (1998); Y.Li, M.C. Bartelt, J.W.Evans, N. Waelchli, E. Kampshoff & K. Kern, Phys. Rev. B **56**, 12539 (1997); see also J. Krug, Physica A **313**, 47 (2002); T. Michely & J. Krug, ref [4] above, pages 220-221.
7. C. Ratsch & J.A. Venables, J. Vac. Sci. Tech. A **21**, S96 (2003).
8. See e.g. M. E. Glicksman, *Diffusion in Solids: Field Theory, Solid-State Principles, and Applications* (John Wiley, New York, 2000), page 64.
9. J.M. Thijssen *Computational Physics* (Cambridge University Press, Cambridge, 1999), appendices A7 and A9. We thank T.J. Newman for introducing us to the FFT method.
10. S.J. Chapman *MatLab® Programming for Engineers* (Brooks/ Cole, Pacific Grove, 2002) was helpful here (Chapter 4). JAV acknowledges useful conversations with D.J. Waxman.
11. P. Yang and J.A. Venables, internal report (2004), and to be published.
12. See J.A. Venables and P. Yang, these proceedings, and J.A. Venables & H. Brune, Phys. Rev. B **66**, 195404 (2002).