On Interacting Particles in 1D and 2D

Joshua DM Hellier

Doctor of Philosophy
The University of Edinburgh
July 2018

Abstract

Interface growth, and in particular the prediction of its rate, has long been a tough problem in statistical physics. In this thesis, I will outline my personal take on the matter, and will showcase a possible approach to it consisting of constructing a microscopic model on a lattice and using this to parametrise a large-scale model of the phenomenon. I will then discuss how to do this with multiple interacting particle species in play.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

Parts of this work have been published in .

(*Joshua DM Hellier, July 2018*)

Acknowledgements

Insert people you want to thank here.

Contents

List of Figures

List of Tables

Chapter 1

Preliminary Work, Background and Motivation

Here we need to talk about the original intent of the project.

The TiO_2/Ti Interface System

A description of the initial problem upon which the project was based.

Initial Attempts to Model the TiO_2/Ti Interface System

The Difficulties of Nonequilibrium Statistical Mechanics

Dynamics of Ionic Crystals

Maybe mention Ewald sums, and the other issues with computations about materials.

Initial Work Done with MD

I used some LAMMPS code to try to work with MD initially; melts and things.

The Problems with MD

Need to explain why issues with using MD, and why I eventually decided it was not a useful technique for this problem; in particular, why MD is fundamentally flawed as a concept.

Simple Large-Scale Models of the Ti/O/Nb Interacting System

I had a think about various methods I could use to tackle the system in question, and decided that the approach would would be most likely to bear fruit would be a continuum-modelled bulk PDE system with appropriate boundary conditions between phases.

Proposed Linear System

Simplest possible model, and why it failed.

Attempts to create a Suitable Nonlinear System

Talk about why nonlinearity is necessary (as in, it just spits out the previous system again), and the difficulties of parametrising it.

Parametrisation from a Microscopic Model

Talk about the Dresden conference and what I learned from it.

The Sticky Particle Model

Model Motivation

As in, why this is a good start in 1d.

Model Definition

Model Properties

Including Detailed Balance, symmetry, "locality". Also mention that it is a Markov process.

Relation to Existing Literature

Generalisation to Higher Dimensions

Including a proof of detailed balance in arbitrary dimensions (on square lattice).

Implications of Initial Work for the PhD Direction

Why the Change of Direction?

Essentially, why trying to solve this particular problem is actually kind of silly, and why having a better theory of driven lattice flows would be more useful.

Why Investigate Flow in the SPM?

Talk about how boundary-condition-induced flow on systems that would otherwise obey detailed balance hasn't really been done before. Bring it around to the question: "Can we have interesting dynamics in a model whose bulk motion is symmetric and obeys detailed balance?"

Chapter 2

Analytical Results about the SPM

We now have a model, the SPM, which should represent the kind of behaviour we are interested in. In this chapter we will attempt to derive analytic results about how material flows in the model. Initially this was all done with the aim of producing an approximation to the behaviour in the hydrodynamic limit and thus informing us about the surface layer formation; however, as you will see the analytic predictions suggest that the flows could be quite interesting in their own right.

Solving Problems in Nonequlibrium Statistical Mechanics

Models in nonequlibrium statistical mechanics which contain nontrivial interactions between components often produce interesting behaviour, hence the wide interest in these models. However, they usually prove to be difficult to "solve" in any concrete sense. In this section I will give a brief overview of solution methods in equilibrium statistical mechanics, why nonequilibrium statistical mechanics problems tend to be harder to solve, and how this affects the way we approach the SPM.

Equilibrium Statistical Mechanics

Equilibrium statistical mechanics is a bread and butter part of undergraduate physics, and there are a great many texts on the subject [2]. When we speak of "solving" an equilibrium statistical mechanics system, the gold standard is to be able to calculate relationships between the statistics of large-scale quantities as a function of the system constraints or their conjugates. This allows one to classify the system's behaviour by making equations of state and identifying phase transitions (situations where at least some large-scale quantity statistics vary with respect to each other in a discontinuous manner). As you will see, the SPM itself is isomorphic to an equilibrium statistical mechanics model so long as we do not drive the system using boundary conditions (e.g. particle reservoirs with different concentrations).

Exact Solutions

A quantity of key interest in equilibrium statistical mechanics is the partition function, usually denoted by Z. Say we have a closed classical mechanical system maintained at constant temperature T by a heat bath, so only energy can enter and leave the system (the canonical ensemble). Let its state space be Ξ, and denote an individual microstate (specific configuration of the system) by ξ. Such a system must of course have a Hamiltonian $H : \Xi \to \mathbb{R}$. The canonical partition function for this system is defined to be

$$Z(\beta) = \int_{\Xi} \mathrm{d}\xi \;\; e^{-\beta H(\xi)}, \tag{2.1}$$

with $\beta T = 1$, where the integrand on the right hand side is the familiar Boltzmann weighting. This quantity is extremely useful, because itself and its derivatives are directly related to the statistics of large-scale quantities. For example, the ensemble-averaged total energy $\langle E \rangle$ satisfies

$$\langle E \rangle = -\frac{\partial \log Z}{\partial \beta} \tag{2.2}$$

If one is able to obtain an expression for the canonical partition function by analytic means, you can calculate essentially any statistical moment of any large-scale quantity you desire, and thus the system is "solved" in the sense we used above.

Approximations

Nonequlibrium Statistical Mechanics

Exact Solutions

Talk about stuff like ASEP. Remember to mention that only very specific models seem to be analytically solvable, in particular you can't have interactions and range in the current models.

Approximations

Approximations in noneq statmech

Similarities and Differences Between Nonequlibrium and Equilibrium Statistical Mechanics

Where does the SPM stand?

Basically, why we can't analytically solve it, and so why performing mean-field approximation is a decent start.

Similarities between the SPM and Established Models in 1D

In the previous section we have discussed the various approaches one might use when attempted to derive properties of a nonequilibrium statistical mechanical system. We will now try to put these ideas into practise on the SPM.

Relationship with the Ising Model

If we implement the rules of the SPM on a periodic domain, we no longer have to deal with boundary conditions. In this special circumstance, we can find an

isomorphism between this model and the Ising model with fixed magnetisation. One does this by associating the Ising spins $\sigma_i \in \{-1, 1\}$ with $\rho_i \in \{0, 1\}$ via

$$\rho_i = \frac{1}{2}(1 + \sigma_i). \tag{2.3}$$

Recalling our proof that the SPM obeys detailed balance, we saw that the equilibrium probability of finding the SPM in a state containing N particle-particle adjacencies is proportional to λ^{-N}. If our Ising Hamiltonian is defined via

$$H = \frac{1}{2}\sum_{i=1}^{L} J\sigma_i\sigma_{i+1 \pmod L}, \tag{2.4}$$

the probability of finding ourselves in a state with N paired spins is $e^{-\beta NJ}$, with $\beta T = 1$. The comparison with the SPM is now obvious; we set $\log\lambda = \beta J$. Thus λ in the SPM is simultaneously playing the of the binding energy and temperature in the Ising model. !Try to compute average energy!

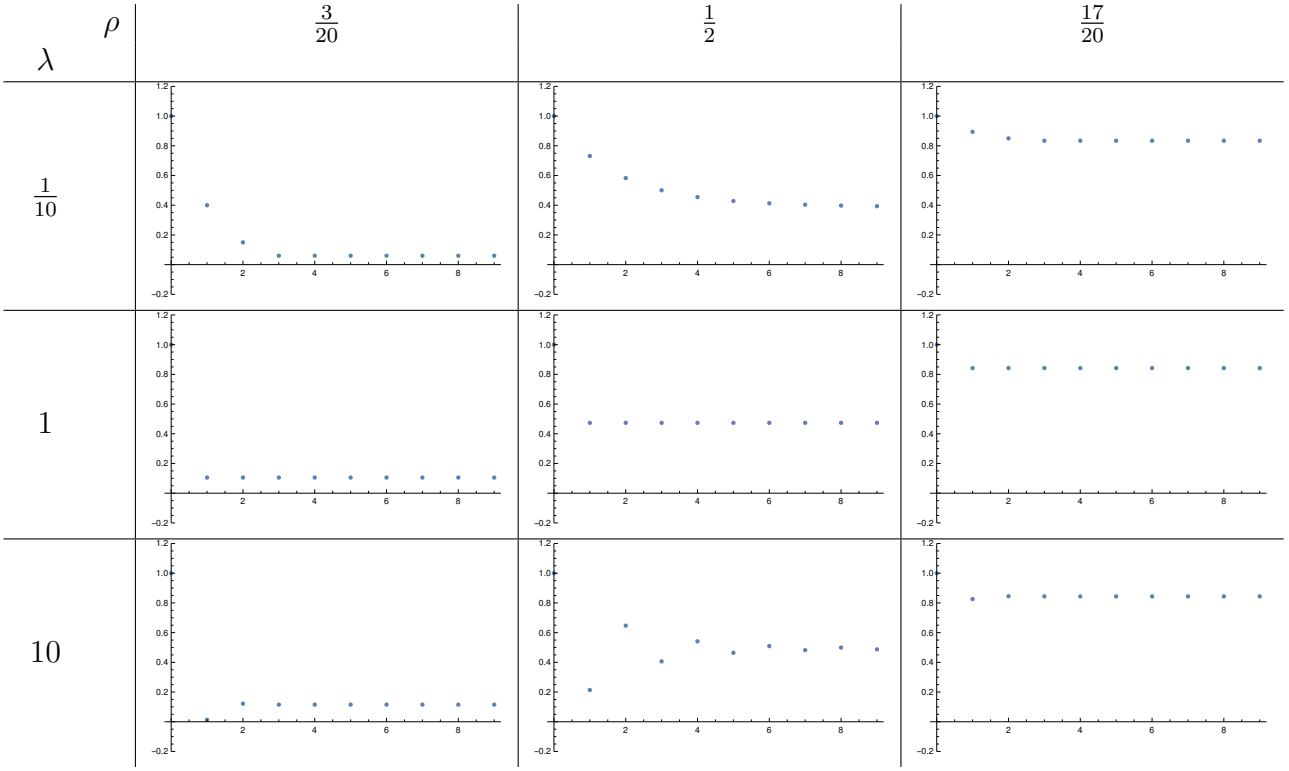
Correlation Functions

For relatively small systems, given a system size L and a number of particles N, we can analytically compute the pairwise correlation function $C(l) = \langle\rho_i\rho_{i+l}\rangle$, or "the probability that site $i + l$ is occupied given that i is (the system is clearly homogeneous in i, so its value is irrelevant). A `Python` code which performs this calculations may be found in Sec. A.1

This is quite a nice result, as we can use simple recursion to perform a calculation which would otherwise be quite difficult to code. Unfortunately the time complexity of the calculation grows exponentially in and L, so the largest L I can reasonably run for is 20. In the table below I have plotted the occupation probability of sites shifted from the origin (assuming the origin is occupied) for a selection of $\lambda = e^{-b}$ and particle densities.

Clearly, as l becomes large, the correlation function tends to the density (note that the way we have define the correlation function it does not subtract this background probability; hence why many definitions do). Very small λ-values cause particles to tend to cluster together, whilst large *lambda* values cause particles and vacancies to tend to alternate. In theory we could use the equivalence with the Ising model to compute correlation lengths as a function of ρ and λ by using the magnetic field in the original Ising model as a Lagrange

Figure 2.1



multiplier in order to fix the total magnetisation (corresponding to particle number in the SPM). However, due to the fact that we cannot accurately compute correlation functions to any decent accuracy using our numerics (see Chap. **??**), we concluded that it was not worth the time to perform the calculation as we would have nothing to compare it to.

Equivalence with the Misanthrope Process

Keeping the SPM on a ring, there is a again a correspondence between it and the Misanthrope Process [1]. The Misanthrope Process is, like the SPM, defined by its rates. This time, however, there can be arbitrarily many particles on a single lattice site. We can choose to consider the symmetric version, in which particles hop in either direction. The defining feature of the process is that particles hop from sites with occupation m to adjacent sites with occupation n with some rate $u(m, n)$.

The equivalence between this and the SPM is made by identifying the **number** of particles on a site in the Misanthrope Process with the **length** of the gap between two particles in the SPM. In this way, one can see that a particle moving, for

example, one step right in the SPM corresponds to a particle moving from one stack to the adjacent one on the left in the Misanthrope Process. To complete the equivalence we set

$$u(m,n) = \begin{cases} \lambda, & n = 0 \\ 1, & \text{otherwise.} \end{cases}$$

Using the result from [1], that the probability weighting of a configuration $\{m_i\}$ may be factorised as

$$P(\{m_i\}) \propto \prod_{i+1}^{N} f(m_i)\delta_{L+N,\sum_{j=1}^{N} m_j}, \tag{2.5}$$

where $f(m)$ is a weighting dependant on the occupation of a site, we see that for the SPM $f(m) = \lambda^{-m}$. Thus, for finite λ

$$\frac{f(m)}{f(m-1)} = \lambda^{-1} \tag{2.6}$$

which remains bounded as $m \to \infty$, therefore this model does not exhibit explosive condensation, again by using [1].

Differences Between SEP and the SPM

Using the Mean-Field Approximation on the SPM

For the reasons discussed above, we cannot analytically solve the SPM on a nonperiodic bounded domain in the same way as SEP. It could be the case that a complete analytic solution method exists, but if it does, we do not know of it, so we will proceed on the assumption that the model is not analytically solvable. Therefore, it would be useful to at least possess approximate solutions, as this can help us by giving us something to test our numerics against, and point us in the direction of interesting behaviours which might occur. We will start by deriving the MFT on a lattice, and will then take the continuum limit (as the lattice spacing tends to zero relative to our scale of interest), as that should predict the dominant behaviour on the macroscopic scale.

Lattice MFT Derivation

As usual, in an MFT approximation, we will be saying that the equal-time probability of the $(i+1)^{\text{th}}$ site being occupied is independent of the probability that the ith site is occupied. More formally, let the probability distribution over system configurations at time t be $p(\xi;t) : \forall t \int_{\Xi} \mathrm{d}\xi \; p(\xi;t) = 1$, and let the operator ρ_i have eigenvalue 1 on configurations with the ith site occupied and 0 on those with it unoccupied. In this language, the mean occupation of site i is

$$\overline{\rho} = \int_{\Xi} \mathrm{d}\xi \;\; \rho_i \xi(t) \tag{2.7}$$

Continuum Limit MFT Derivation

Negative Diffusion Coefficients

When do they happen? What do they mean?

Continuum Limit MFT Solutions

There's a bunch of these.

Continuum MFT Breakdown

The SPM in Higher Dimensions

Kinda repeat the earlier stuff in higher dimensions, particularly 2 where we actually have data. Maybe less need for elaborate sections structure here; just write freely and see how it goes.

Chapter 3

Numerical Results about the SPM

Numerical Simulations of Continuous-Time Markov Processes

Known Methods

Discuss commonly-used methods, and how they work. Eventually, talk about why we are using the n-fold way.

`KMCLib`

Talk about how it works, why I picked it over other implementations.

Running `KMCLib` on `Eddie3`

How calculations are managed day-to-day.

Calculation Results

1D

2D

Chapter 4

Conclusions

Need to summarise the key results of the research here, and give an overview.

Appendix A

Code Listings

1d Ising Correlation Functions

This `Python` script computes the probability of a site being occupied l lattice spacings away from an occupied site. It requires the system size L and the number of particles N as inputs. The output is saved in a file called `corrFnResults.m`, which is formatted so that it may be used by `Mathematica`.

```
import copy
import sys

def configMake(L, N, prevList, totList):
    if L==1:
        endList = [copy.deepcopy(prevList), N]
        totList.append(unfold(endList))
        return [N]
    if N==0:
       return configMake(L-1, 0, [copy.deepcopy(prevList), 0], totList)
    if L==N:
        return configMake(L-1, N-1, [copy.deepcopy(prevList), 1], totList)
    return [configMake(L-1, N, [copy.deepcopy(prevList), 0], totList),
    configMake(L-1, N-1, [copy.deepcopy(prevList), 1], totList)]

def adjSum(candList):
    listLen = len(candList)
    total = 0
    for index in range(0, listLen):
        total += candList[index-1]*candList[index]
    return total

def unfold(candList):
    if isinstance(candList, list):
        if len(candList)==2:
            return unfold(candList[0])+unfold(candList[1])
```

```
        if len(candList)==1:
            return candList
        if len(candList)==0:
            return []
    return [candList]

def listCollate(candList):
    maxItem = 0
    for index in candList:
        if index > maxItem:
            maxItem = index
    outPut = []
    for size in range(0, maxItem+1):
        numCounts = 0
        for index in candList:
            if index == size:
                numCounts += 1
        outPut.append((size, numCounts))
    return outPut

def genCorrFn(L, N):
    totList = []
    allStates = configMake(L, N, [], totList)
    restStates = []
    weightList = []
    maxAdj = 0
    for state in totList:
        if state[0]==1:
            restStates.append((state, adjSum(state)))
            if restStates[-1][1]>maxAdj:
                maxAdj = restStates[-1][1]
            weightList.append(restStates[-1][1])
    partFnList = listCollate(weightList)
    print(partFnList)
    partitionFn = "("
    for pair in partFnList:
        partitionFn += str(pair[1])+"␣Exp["+str(pair[0]-maxAdj)+"b]␣+␣"
    partitionFn += "0)"
    print(partitionFn)
    finalOut = "{"
    for shift in range(0, L-L/2):
        tempList = []
        for config in restStates:
            if config[0][shift] == 1:
                tempList.append(config[1])
        stateDist = listCollate(tempList)
        outSum = "{"+str(shift)+",␣("
        for pair in stateDist:
            outSum += str(pair[1])+"␣Exp["+str(pair[0]-maxAdj)+"b]␣+␣"
        outSum += "0)/"+partitionFn+"}"
        finalOut += outSum
        if shift != L-L/2-1:
            finalOut += ",␣"
    finalOut+="}"
    return finalOut

L = int(sys.argv[1])
```

```
with open("corrFnResults.m", 'w') as f:
    f.write("{")
    for n in range(2, L-2):
        f.write("{"+str(n)+"/"+str(L)+", "+genCorrFn(L, n)+"}, ")
    f.write(genCorrFn(L, L-2) + "}")
```

Bibliography

[1] Evans, M. R., and B. Waclaw. "Condensation in stochastic mass transport models: beyond the zero-range process." *Journal of Physics A: Mathematical and Theoretical* 47, 9: (2014) 095,001. http://stacks.iop.org/1751-8121/47/i=9/a=095001.

[2] Landau, L. D., E. M. Lifshitz, and L. Pitaevskii. "Statistical physics, part I.", 1980.