

On Interacting Particles in 1D and 2D

Joshua DM Hellier



Doctor of Philosophy
The University of Edinburgh
February 2019

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, February 2019)

Acknowledgements

Insert people you want to thank here.

Contents

| | |
|--|------|
| Abstract | i |
| Declaration | ii |
| Acknowledgements | iii |
| Contents | iv |
| List of Figures | viii |
| List of Tables | x |
| 1 Preliminary Work, Background and Motivation | 1 |
| 1.1 The TiO ₂ /Ti Interface System | 1 |
| 1.2 Initial Attempts to Model the TiO ₂ /Ti Interface System..... | 1 |
| 1.2.1 Molecular Dynamics..... | 1 |
| 1.3 Large-Scale Models of the Ti/O/Nb Interacting System | 2 |
| 1.3.1 Proposed Linear System..... | 2 |
| 1.3.2 Attempts to create a Suitable Nonlinear System | 2 |
| 1.3.3 Parametrisation from a Microscopic Model | 2 |
| 1.4 The Sticky Particle Model..... | 2 |
| 1.4.1 Model Motivation | 2 |

| | | |
|----------|---|----------|
| 1.4.2 | Model Definition | 3 |
| 1.4.3 | Model Properties | 3 |
| 1.4.4 | Relation to Existing Literature | 3 |
| 1.4.5 | Generalisation to Higher Dimensions..... | 3 |
| 1.5 | Implications of Initial Work for the PhD Direction..... | 3 |
| 1.5.1 | Why the Change of Direction?..... | 3 |
| 1.5.2 | Why Investigate Flow in the SPM?..... | 3 |
| 2 | Analytical Results about the SPM | 4 |
| 2.1 | Solving Problems in Nonequilibrium Statistical Mechanics..... | 4 |
| 2.1.1 | Equilibrium Statistical Mechanics..... | 5 |
| 2.1.2 | Nonequilibrium Statistical Mechanics | 6 |
| 2.1.3 | Where does the SPM stand? | 8 |
| 2.2 | Similarities between the SPM and Established Models in 1D..... | 9 |
| 2.2.1 | Relationship with the Ising Model | 9 |
| 2.2.2 | Correlation Functions | 10 |
| 2.2.3 | Equivalence with the Misanthrope Process | 11 |
| 2.2.4 | Calculation of the Partition Function of the SPM on a Closed Ring..... | 12 |
| 2.3 | Using the Mean-Field Approximation on the SPM | 16 |
| 2.3.1 | Lattice MFT Derivation | 16 |
| 2.3.2 | Continuum Limit MFT Derivation..... | 18 |
| 2.3.3 | Negative Diffusion Coefficients..... | 20 |
| 2.3.4 | Continuum Limit MFT Solutions | 22 |
| 2.3.5 | Implications of Continuum MFT Breakdown..... | 29 |

| | | |
|----------|---|-----------|
| 2.4 | The SPM in Higher Dimensions..... | 29 |
| 2.4.1 | Symmetry + Locality + Detailed Balance = Unique 1-Parameter SPM | 30 |
| 2.4.2 | MFT of the n -Dimensional SPM | 35 |
| 2.5 | Conclusions About the SPM MFT | 39 |
| 3 | Transition Rate Matrix Analysis | 41 |
| 3.1 | The Transition Rate Operator for the SPM..... | 41 |
| 3.1.1 | A Small Worked Example: Closed System | 43 |
| 3.2 | Forming the TRM for Systems with Dirichlet Boundary Conditions | 45 |
| 3.2.1 | Dirichlet Boundary Conditions..... | 45 |
| 3.2.2 | Another Small Worked Example: Open System | 46 |
| 3.2.3 | Formation of the TRM in Sparse Format | 49 |
| 3.3 | The Eigenspectrum of the TRM | 51 |
| 3.3.1 | The Computation of the TRM Eigenspectrum..... | 51 |
| 3.3.2 | The Structure of the TRM Eigenspectrum..... | 51 |
| 3.3.3 | Current and Density in the Steady State..... | 62 |
| 3.4 | Time-Dependent Properties of Small SPM Systems..... | 69 |
| 3.4.1 | The Relaxation Time for the SPM..... | 69 |
| 3.4.2 | Time-Evolution of States..... | 70 |
| 3.5 | Conclusions | 71 |
| 4 | Monte-Carlo Simulations of the SPM | 74 |
| 4.1 | Numerical Simulations of Continuous-Time Markov Processes | 74 |
| 4.1.1 | Purpose of Monte-Carlo Methods | 75 |
| 4.1.2 | Evenly-Spaced Timesteps | 75 |

| | | |
|---------------------|--|----|
| 4.1.3 | The N-Fold Way, or Gillespie Algorithm | 77 |
| 4.2 | Implementation of Monte Carlo Methods..... | 79 |
| 4.2.1 | Our Implementation of a Metropolis-Hastings Algorithm with Evenly-Space Timesteps | 79 |
| 4.2.2 | KMCLib | 80 |
| 4.2.3 | Managing KMCLib Calculations in Parallel | 83 |
| 4.3 | 1D Calculation Results..... | 85 |
| 4.3.1 | Calculational Choices | 85 |
| 4.3.2 | Flow Patterns | 85 |
| 4.3.3 | Current | 85 |
| 4.3.4 | Density | 85 |
| 4.3.5 | Diffusion Coefficient | 85 |
| 4.4 | 2D Calculation Results..... | 85 |
| 4.4.1 | Calculational Choices | 85 |
| 4.4.2 | Results..... | 85 |
| 4.5 | Conclusions | 85 |
| 5 | Conclusions | 86 |
| A | Code Listings | 87 |
| A.1 | 1d Ising Correlation Functions | 87 |
| A.2 | <i>n</i> -Dimensional Continuum-Limit MFT | 89 |
| Bibliography | | 90 |

List of Figures

| | |
|--|----|
| (2.1) Plots of the equal-time particle density correlation function on a ring. | 10 |
| (2.2) The free energy density of the SPM on a closed ring, as a function of density and λ | 14 |
| (2.3) The chemical potential of the SPM on a closed ring, as a function of density and λ | 15 |
| (2.4) Some plots of the variation of the MFT diffusion coefficient with density, for some selected λ | 19 |
| (2.5) A contour plot of the variation of the MFT diffusion coefficient with density and stickiness. | 20 |
| (2.6) The variation in the current at fixed λ with respect to the boundary densities. | 24 |
| (2.7) The variation of flow rate with respect to λ in the MFT, with fixed boundary densities. | 25 |
| (2.8) The dependence of the critial value of λ required for backward diffusion on the boundary conditions. | 26 |
| (2.9) The variation of the system-wide average density with respect to λ in the MFT, with fixed boundary densities. | 27 |
| (2.10)The possibe transitions which may occur in a symmetric local 2-dimensional hopping model. | 31 |
| (2.11)Figure demonstrating that a local system obeying detailed balance and symmetry cannot have interesting local configuration-dependent dynamics. | 32 |
| (2.12)Figure demonstrating that particle motion out of a hyperplane with m adjacent particles away from an empty space moves with rate λ^m | 34 |

| | |
|--|----|
| (2.13) Figure demonstrating that particle motion out of a hyperplane with m adjacent particles away from an empty space moves with rate λ^m | 34 |
| (2.14) The variation of the density of maximal/minimal flow with λ | 38 |
| (2.15) 2d SPM MFT current flow due to boundaries, for a selection of λ . . | 40 |
| (2.16) 2d SPM MFT current flow as a function of λ , with fixed boundaries. | 40 |
| | |
| (3.1) The variation of configuration probabilities with λ for a small open system. | 48 |
| (3.2) The variation of the eigenspectrum of the TRM with λ for a small open system. | 49 |
| (3.3) The TRM eigenspectrum for a system with $L = 5$, $b = 1000$ | 54 |
| (3.4) The lower part of the TRM eigenspectrum for a system with $L = \{5, 10\}$, $b = \{100, 1000\}$ | 56 |
| (3.5) A graph of the scaling of the number of slow modes with system size. | 58 |
| (3.6) The upper TRM eigenspectrum for a system with $L = \{5, 10\}$, $b = \{100, 1000\}$ | 60 |
| (3.7) The “full” TRM eigenspectrum for a system with $L = 8$, $b = 1000$. | 61 |
| (3.8) The variation of the density profile with λ for a system of size $L = 12$ and boundaries ($\rho_0 = 0.6$, $\rho_L = 0.4$). | 63 |
| (3.9) The variation of the current (measured flowing from high density boundary to low) with λ for a system of size $L = 10$ and boundaries ($\rho_0 = 0.6$, $\rho_L = 0.4$). | 65 |
| (3.10) The variation of the diffusion coefficient of a system of size $L = 10$ with respect to λ and ρ | 67 |
| (3.11) The variation of the order parameter χ for a system of size $L = 10$ with respect to λ and ρ | 68 |
| (3.12) The dependence of the relaxation time on λ for three sets of boundary conditions. | 70 |
| (3.13) The time-evolution of uniform distributions to equilibrium. | 72 |
| | |
| (4.1) Illustration of the method for choosing successor states in the n-fold way. | 79 |

List of Tables

| | |
|---|----|
| (2.1) The dependence of MFT current upon dimension in the SPM. . . | 36 |
| (3.1) Tabulated values for the variation of the width of the slow band with system size. | 57 |

Chapter 1

Preliminary Work, Background and Motivation

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

1.1 The TiO₂/Ti Interface System

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

1.2 Initial Attempts to Model the TiO₂/Ti Interface System

1.2.1 Molecular Dynamics

Need to explain why issues with using MD, and why I eventually decided it was not a useful technique for this problem.

1.3 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 be most likely to bear fruit would be a continuum-modelled bulk PDE system with appropriate boundary conditions between phases.

1.3.1 Proposed Linear System

Simplest possible model, and why it failed.

1.3.2 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.

1.3.3 Parametrisation from a Microscopic Model

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

1.4 The Sticky Particle Model

1.4.1 Model Motivation

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

1.4.2 Model Definition

1.4.3 Model Properties

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

1.4.4 Relation to Existing Literature

1.4.5 Generalisation to Higher Dimensions

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

1.5 Implications of Initial Work for the PhD Direction

1.5.1 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.

1.5.2 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.

2.1 Solving Problems in Nonequilibrium Statistical Mechanics

Models in nonequilibrium 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.

2.1.1 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 [6]. 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). Once we introduce such driving forces however, we find that we can no longer use equilibrium analysis, and things get a bit more difficult.

Exact Solutions

A quantity of key interest in equilibrium statistical mechanics is the partition function [8], 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 $H : \Xi \rightarrow \mathbb{R}$. The canonical partition function for this system is defined to be

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

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

$$\langle E \rangle = -\frac{\partial \log Z}{\partial \beta} \quad (2.2)$$

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 (**state variable**) you desire, and thus the system is “solved” in the sense we used above.

Approximate Methods for Analysing Equilibrium Statmech Systems

Of course, the situation in which one can simply evaluate the partition function exactly is extremely rare in equilibrium statistical mechanics, at least in the large-scale limit (the one of principle interest, as it is required for most interesting phenomena such as phase transitions). More often, one might approximate the partition function itself, perhaps by converting the required integral (Eq. 2.1) into an asymptotic series in one of the thermodynamic variables; this is effectively what one does when analysing a system in terms of instanton transitions.

Another approach is to directly deal with the state variables we are interested in themselves, and try to find approximate relationships between them in order to classify their interdependence (an **equation of state**). Equilibrium mean-field theory (**MFT**) is exactly such a method. In mean-field theory, we introduce the means of thermodynamic quantities of interest as independent variables, and then make the assumption that they have no nontrivial correlations. In practical terms, this means that for thermodynamic variables x and y , we assume that

$$\langle xy \rangle = \langle x \rangle \langle y \rangle. \quad (2.3)$$

This is of course quite an assumption to make, although it is often the case that it is true to low order in some asymptotic expansion. Typically, MFT tends to work well in higher dimensions and more weakly correlated systems. In the case of the Ising Model (the Weiss Molecular Field), it incorrectly predicts a phase transition in 1d, but for 2d and above it is at least qualitatively correct.

Finally, one can attempt to sample directly from the space of microstates numerically via Monte Carlo methods, and in doing so build up information about state variables that way. We will say more about this in Ch. 4.

2.1.2 Nonequilibrium Statistical Mechanics

Nonequilibrium statistical mechanics differs from equilibrium statistical mechanics in the sense that it is “out of equilibrium”. The actual meaning of this

is that there are nontrivial currents (be them of energy, matter, or otherwise) flowing through the system. Note that it is perfectly possible for a nonequilibrium statistical mechanical system to be out of equilibrium but still in a steady state, in the sense that the system's microstate can be constantly changing but still maintaining a fixed time-averaged ensemble distribution allowing currents to flow.

In general, nonequilibrium statistical mechanics problems tend to be quite a lot harder to get a handle on than equilibrium statistical mechanics problems. The principle reason for this is the issue of time-dependence. In true equilibrium, where there is no flow and therefore no overall motion, there is no passage of time, which effectively reduces the number of variables under consideration. This, coupled with the second law of thermodynamics, means that whatever state we are in must be an allowed state with maximal entropy; in this case, entropy ends up acting a little bit like a Lyapunov function, which we can use to find candidate final states.

This is of no help to us at all out of equilibrium; asking a question such as “how much current flows across a given concentration gradient?” inherently involves time, and so we can’t just make a maximal-entropy argument to get the answer. There are occasions in the literature where a principle of “maximal entropy production” is invoked in order to close sets of equations, but as far as we have seen there is not yet a proof of why such a thing should in general be true.

Exact Solutions

There are some problems in nonequilibrium statistical mechanics known which have exact solutions. These include the Ising model in 1 and 2 dimensions, as well as a class of problems which can be solved in steady by matrix products or Bethe ansatz. These are the subject of much interest at the moment, not least because one can often use isomorphisms between systems which are known to be integrable in order to discover new integrable systems. Unfortunately, building up such an exact solution to a specific problem from scratch does not seem to be possible in every case, so it remains to be seen how useful these methods really are for tackling problems of actual physical interest.

Approximations

As with equilibrium statistical mechanics, there are a bunch of approximate methods which can be invoked to probe system behaviour, some of which we have used some of these in our research. Instead of solving the full system as an integrable system, there are approaches in which instantons are used to “patch together” local solutions to try to get a grasp of the whole picture. One can, as always, simply simulate a system by numerical means and try to derive useful statistics from the output. In order to do this more efficiently, one can possibly use results from Large Deviation Theory (LDT) in order to boost the strength of the tails of the distributions and so capture the essence of some useful large deviation function. However, again this doesn’t seem to be something that can be done completely generically.

Of course, mean-field theory is still an option, although now that time dependence has been added it tends to produce a set of coupled ODEs for the mean variables under consideration. It is also in principle possible to study small-scale systems essentially exactly by numerical analysis of the relevant transition state matrix (TRM). Of course, this has the disadvantage that it is only a small finite system, so some of the phenomena observed in it will differ from larger versions of the same system.

2.1.3 Where does the SPM stand?

The issue of how to analyse the SPM leaves us in a slightly awkward position. It is almost as simple as the Symmetric Exclusion Process; however, the extra interaction in the SPM makes it quite different, as it is in fact longer-ranged. Thus, it’s a bit difficult to see immediately how to modify a SEP matrix-product steady state in order to solve the SPM. In a similar vein, it might in fact be possible to solve the SPM exactly using thermodynamic Bethe ansatz, but we have neither the expertise nor the time to properly investigate that possibility.

Instead, we have gone for a somewhat more pedestrian, traditional approach. In the rest of this chapter, we will discuss the relationships enjoyed by the SPM and some well-known models, and will then perform mean-field analysis, in the hope of obtaining some at least qualitative results about how the SPM behaves. In Ch. 3 we will then consider some semianalytic solutions to the SPM in finite systems in 1d via TRM analysis. Finally, in Ch. 4 we will compare the TRM

and MFT results to numerical simulations performed using Monte-Carlo (MC) methods. We have also performed MC calculations and have MFT results for the 2-dimensional situation.

2.2 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.

2.2.1 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

$$\rho_i = \frac{1}{2} (1 + \sigma_i). \quad (2.4)$$

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

$$H = \frac{1}{2} \sum_{i=1}^L J \sigma_i \sigma_{i+1} \pmod{L}, \quad (2.5)$$

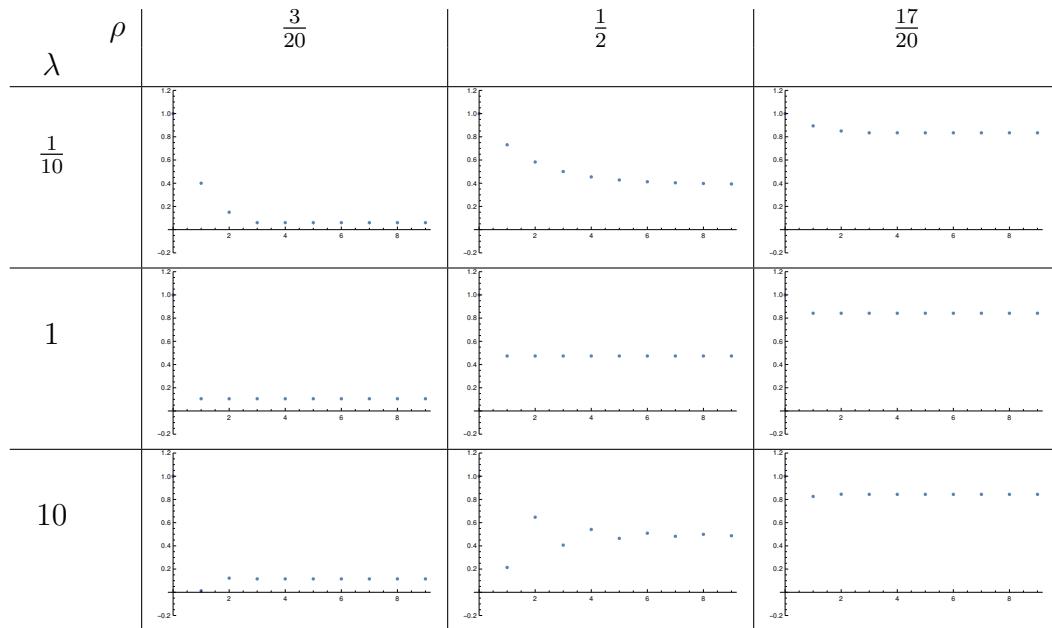
the probability of finding ourselves in a state with N paired spins is $e^{-\beta N J}$, with $\beta T = 1$. The comparison with the SPM is now obvious; we set $\log \lambda = \beta J$. Thus λ in the SPM has a one-to-one mapping to the ratio of the binding energy to the temperature in the Ising model. !Try to compute average energy, if there's time!

2.2.2 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.

Figure 2.1 Here are some particle-particle correlation functions for the SPM on a small closed ring, with the density chosen by choosing how many particles to insert at the start. The system in this case has 20 lattice sites. This was calculated using computer-assisted algebra and the various density and stickiness combinations should give an overall impression as to their structure.



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 λ -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 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.

2.2.3 Equivalence with the Misanthrope Process

Keeping the SPM on a ring, there is again a correspondence between it and the Misanthrope Process [2]. 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 [2], that the probability weighting of a configuration $\{m_i\}$ may be factorised as

$$P(\{m_i\}) \propto \prod_{i=1}^N f(m_i) \delta_{L+N, \sum_{j=1}^N m_j}, \quad (2.6)$$

where $f(m)$ is a weighting dependant on the occupation of a site, we see that for

the SPM

$$f(m) = \begin{cases} \lambda, & n = 0 \\ 1, & \text{otherwise} \end{cases}. \quad (2.7)$$

Thus, for finite λ

$$\frac{f(m)}{f(m-1)} = 1, \forall m > 2, \quad (2.8)$$

which remains bounded as $m \rightarrow \infty$, therefore this model does not exhibit explosive condensation, again by using [2].

2.2.4 Calculation of the Partition Function of the SPM on a Closed Ring

Whilst we are on the subject of the duality between the SPM and the Misanthrope Process, we can observe that the probability weighting of any configuration is proportional to λ^{-k} , where k is the number of particle-particle adjacencies in the SPM and the number of empty slots in the Misanthrope Process. This begs the question: if we know the weightings, can we calculate the partition function, and therefore other quantities such as the free energy or chemical potential, for the SPM on a closed ring with particle density ρ ?

In order to attempt this, we must first make two observations from the field of combinatorics:

- The number of possible ways to select, without ordering or replacement, M objects from N is

$$\binom{M}{N} = \frac{N!}{M!(N-M)!}. \quad (2.9)$$

- The number of ways to insert M unlabelled balls into N boxes is

$$\binom{N+M-1}{M-1} = \frac{(N+M-1)!}{N!(M-1)!}. \quad (2.10)$$

Now let us consider a Misanthrope Process on a closed ring of N slots, into which we wish to distribute $L - N$ particles; this corresponds to the SPM on a ring with L slots containing N particles. Say we wish to distribute the particles so that only M of the slots contain any particles at all; then the above two combinatorial results suggest that there are $\frac{N!}{M!(N-M)!}$ ways to choose which M slots, into which

we insert one particle each, leaving $\frac{(L-N-1)!}{(L-N)!(M-1)!}$ ways to insert the remainder. Thus, the overall number of possible ways to distribute the particles so that only M slots contain any at all is

$$C_M = \frac{N!(L-N-1)!}{M!(N-M)!(L-N)!(M-1)!}. \quad (2.11)$$

As we know, the weighting of a configuration in which only M of the slots are nonempty is $\lambda^{-(N-M)}$; therefore, the partition function $Z_L(\lambda, N)$ for a system of size L is

$$Z_L(\lambda, N) = \sum_{M=1}^{M=\min\{N-1, L-N\}} \left[\frac{N!(L-N-1)!}{M!(N-M)!(L-N)!(M-1)!} \lambda^{-(N-M)} \right]. \quad (2.12)$$

For statistical mechanics, we are really interested in the situation where the system size is very large. Therefore, let us define $\rho, m \in (0, 1)$ so that $N = \rho L$ and $M = mL$, and invoke the Stirling approximation $\log x! \sim x \log x$ for large x . Regarding L as a large constant, and keeping only leading order behaviour in L , we find that

$$\begin{aligned} Z_L(\lambda, \rho) \sim & \int_0^{\min\{\rho, 1-\rho\}} dm \exp L [-2m \log m + (1-\rho) \log(1-\rho) + \rho \log \rho \\ & - (1-\rho-m) \log(1-\rho-m) - (\rho-m) \log(\rho-m) - (\rho-m) \log \lambda] \end{aligned} \quad (2.13)$$

This integral looks quite intractable, but recall that in the limit $L \rightarrow \infty$ we can evaluate it asymptotically using the Laplace's Method. This requires finding the location of extrema of the exponentiated term as a function of m ; these occur when

$$\lambda(1-m-\rho)(\rho-m) = m^2. \quad (2.14)$$

One of the solutions occurs in $(0, \min\{\rho, 1-\rho\})$, at

$$m_+ = \frac{\lambda + \sqrt{\lambda^2 + 4\lambda(1-\lambda)\rho(1-\rho)}}{2(1-\lambda)}, \quad (2.15)$$

and a little analysis reveals that it is indeed a maximum, as required for the use

of Laplace's Method. Therefore, we find that

$$\begin{aligned} \sqrt[L]{Z_L(\lambda, \rho)} &\sim \sqrt{2\pi} \frac{(1-\rho)^{1-\rho}\rho^\rho}{m_+^{2m_+} (\rho - m_+)^{\rho-m_+} (1 - m_+ - \rho)^{1-m_+-\rho}} \\ &\times \sqrt{\frac{m_+ (\rho - m_+) (1 - m_+ - \rho)}{2(1-\rho)\rho - m_+}} \lambda^{m_+-\rho}. \end{aligned} \quad (2.16)$$

From here one can use computer algebra to obtain the free energy density $F(\rho, \lambda) = -\frac{\log Z_L}{L}$ (Fig. 2.2) and chemical potential $\mu(\rho, \lambda) = \frac{\partial F}{\partial \rho}$ (Fig 2.3) of the SPM system.

Figure 2.2 *The variation of the SPM free energy density on a large closed ring as a function of particle density and stickiness parameter λ . The patches on the line $\lambda = 1$ are due to the way that m_+ is calculated; although a little analysis reveals that it is in fact a well-behaved removable singularity, numerical errors cause the plotting numerics to behave badly in places.*

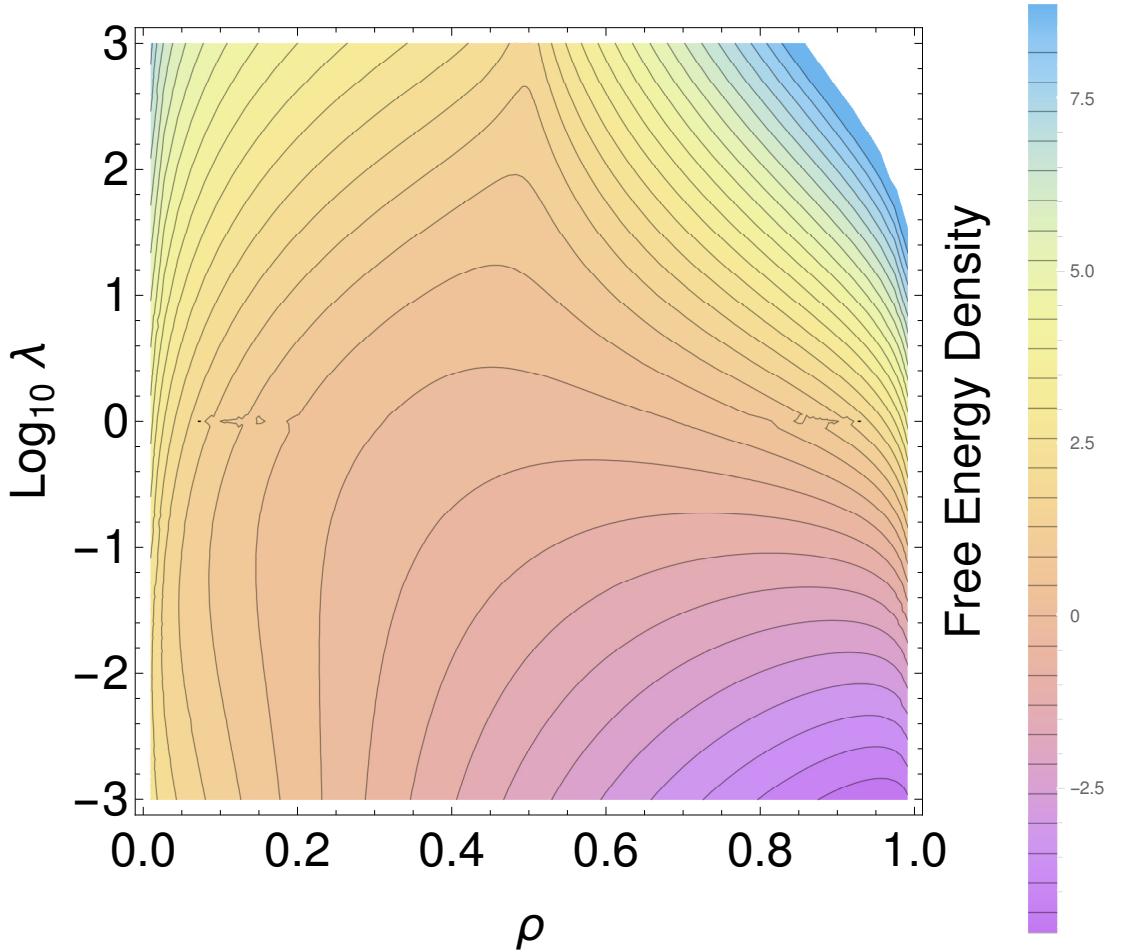
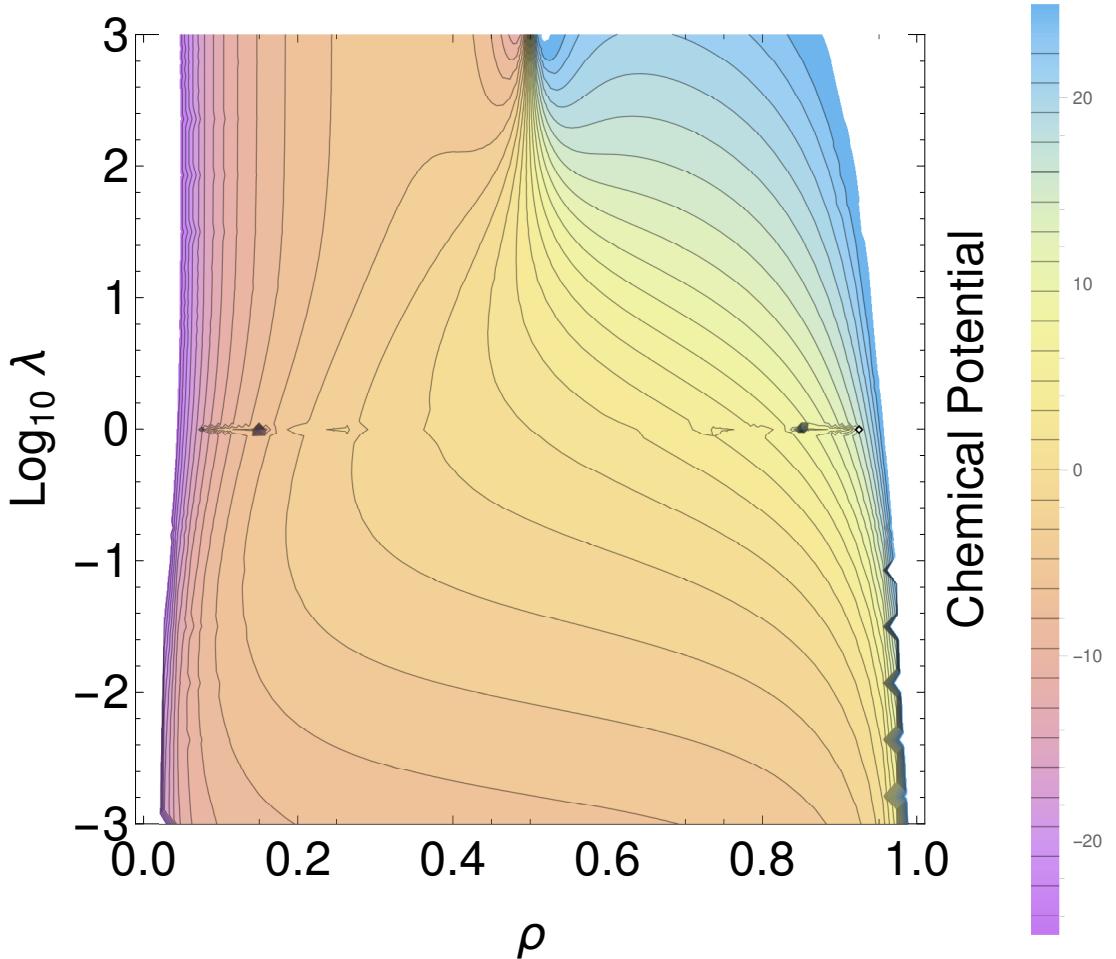


Figure 2.3 The variation of the SPM chemical potential on a large closed ring as a function of particle density and stickiness parameter λ . Again, we see some bad behaviour at $\lambda = 1$, for the same reasons as before.



Of course, in thermodynamic equilibrium a system generally attempts to lower its total free energy to the lowest value allowed by the constraints. If we hold λ constant whilst allowing ρ to vary, this corresponds to the curve $\mu(\rho, \lambda) = 0$, which one can follow on Fig. 2.3; a glance at Fig. 2.2 confirms that this is indeed a minimum. Thus we should expect that a very large system connected to a particle reservoir would tend towards having a total density such that $\mu = 0$, regardless of the particle density in the reservoir, as the particle density in the system would only be pinned to that of the reservoir near to the boundary. Therefore, at large λ we would see systems tend towards a density of $\rho = \frac{1}{2}$, whereas for small λ the system would tend to fill. Of course, if we were to hook a system up to two different boundaries and cause a current to flow, as we have done many times in the course of this project, that might change things, as now the system would

be in a nonequilibrium steady state as opposed to a thermodynamic equilibrium, and those are not quite the same beast.

2.3 Using the Mean-Field Approximation on the SPM

For the reasons discussed in Sec. ??, we do not possess an analytic solution for the SPM on a nonperiodic bounded domain. Such a solution might exist, but we will proceed on the assumption that the model is not analytically solvable. Therefore, it would be useful to at least possess approximate analytic 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.

2.3.1 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 i^{th} site is occupied. More formally, let us denote the mean occupation of the i^{th} site at time t by $\rho_i(t)$. When we invoke the mean-field approximation, we say that the mean occupations of sites at equal times are independent; thus, the probability that site $j \neq i$ occupied given that site i is occupied is $\rho_j(t)$. We can use this to calculate the rate at which $\rho_i(t)$ increases and decreases, and so obtain a system of coupled ODEs for $\rho_i(t)$.

Let us first consider the situation where the i^{th} site is unoccupied. The probability of this being the case is $(1 - \rho_i(t))$. A particle could move from site $(i - 1)$ or site $(i + 1)$, but only if those sites are currently occupied. Assuming that site $(i - 1)$ is occupied (occurring with probability ρ_{i-1} in MFT), the rate at which it would jump to site i would depend on the occupation of site $(i - 2)$, as it would be 1 if it was unoccupied and λ if it was occupied. Phrasing this in MFT terms, and suppressing t -dependence for brevity, the rate at which $\rho_i(t)$ is increased by

particles coming from the left is

$$\tau_0^{-1} (1 - \rho_i) \rho_{i-1} [(1 - \rho_{i-2}) \cdot 1 + \rho_{i-2} \cdot \lambda]. \quad (2.17)$$

By symmetry, the income of particles from the right is

$$\tau_0^{-1} (1 - \rho_i) \rho_{i+1} [(1 - \rho_{i+2}) \cdot 1 + \rho_{i+2} \cdot \lambda]. \quad (2.18)$$

Using similar logic, but shifting things around slightly, the rate at which particles leave site i to go to site $i + 1$ is

$$\tau_0^{-1} (1 - \rho_{i+1}) \rho_i [(1 - \rho_{i-1}) \cdot 1 + \rho_{i-1} \cdot \lambda], \quad (2.19)$$

and similarly

$$\tau_0^{-1} (1 - \rho_{i-1}) \rho_i [(1 - \rho_{i+1}) \cdot 1 + \rho_{i+1} \cdot \lambda] \quad (2.20)$$

is the rate at which particles leave i to go to $i - 1$.

At this point it becomes fairly clear why we introduced the quantity $\zeta = 1 - \lambda$, as it neatens things up in general. The total rate at which particles enter site i is

$$\tau_0^{-1} (1 - \rho_i) [(1 - \zeta \rho_{i-2}) \rho_{i-1} + (1 - \zeta \rho_{i+2}) \rho_{i+1}] \quad (2.21)$$

whilst they leave at rate

$$\tau_0^{-1} \rho_i [(1 - \zeta \rho_{i+1}) (1 - \rho_{i-1}) + (1 - \zeta \rho_{i-1}) (1 - \rho_{i+1})] \quad (2.22)$$

Combining the rates of arriving and leaving, we obtain our main result:

$$\begin{aligned} \tau_0 \frac{\partial \rho_i}{\partial t} &= (1 - \rho_i) [(1 - \zeta \rho_{i-2}) \rho_{i-1} + (1 - \zeta \rho_{i+2}) \rho_{i+1}] \\ &\quad - \rho_i [2\zeta \rho_{i-1} \rho_{i+1} - (3 - \zeta) (\rho_{i-1} + \rho_{i+1}) + 2]. \end{aligned} \quad (2.23)$$

This is a nice result, and in theory we could stop right here and we could make a computational scheme for solving this as a sequence. However, there are a few issues. For one thing, $\rho_i(t)$ isn't the mean of a quantity whose variance is being suppressed by the law of large numbers, as is desired when using the MFT approximation. Thus, it is merely a rough sketch of what might happen, as variances and correlations between sites aren't suppressed. On the other hand, it simply relates the occupations of nearby sites, whereas we would find a description of the bulk flow to be much more useful. Therefore, we may as well take the

continuum limit to see how flow depends on concentration gradient and local density.

2.3.2 Continuum Limit MFT Derivation

To take the continuum limit, let's promote $\rho_i(t)$ to $\rho(x, t)$ so that

$$\rho_{i+m}(t) \rightarrow \rho(x + am, t). \quad (2.24)$$

Now we can Taylor expand for $\rho_{i+m}(t)$, as

$$\rho(x + am, t) = \rho(x, t) + ma \frac{\partial \rho(x, t)}{\partial x} + \frac{1}{2} m^2 a^2 \frac{\partial^2 \rho(x, t)}{\partial x^2} + \mathcal{O}(a^3). \quad (2.25)$$

Preferably with the aid of a computational algebra package (in my case **Wolfram Mathematica**), one may directly substitute Taylor expansions for the required ρ_j into Eq. 2.23, continuing to truncate at $\mathcal{O}(a^3)$. Doing so, and collecting terms, we find that

$$\tau_0 \frac{\partial \rho}{\partial t} = a^2 \left[1 - \zeta \rho (4 - 3\rho) \frac{\partial^2 \rho}{\partial x^2} \right] 2a^2 \zeta (3\rho - 2) \left(\frac{\partial \rho}{\partial x} \right)^2 + \mathcal{O}(a^4), \quad (2.26)$$

which may be factorised into the more convenient form

$$\frac{\partial \rho}{\partial t} = \frac{a^2}{\tau_0} \frac{\partial}{\partial x} \left\{ [1 - \zeta \rho (4 - 3\rho)] \frac{\partial \rho}{\partial x} \right\}, \quad (2.27)$$

which is a continuity equation

$$\frac{\partial \rho}{\partial t} = \frac{\partial J}{\partial x} \quad (2.28)$$

with current

$$J = -\frac{a^2}{\tau_0} [1 - \zeta \rho (4 - 3\rho)] \frac{\partial \rho}{\partial x}. \quad (2.29)$$

Considering Fick's Law

$$J = -D \frac{\partial \rho}{\partial x}, \quad (2.30)$$

we see that our diffusion coefficient is

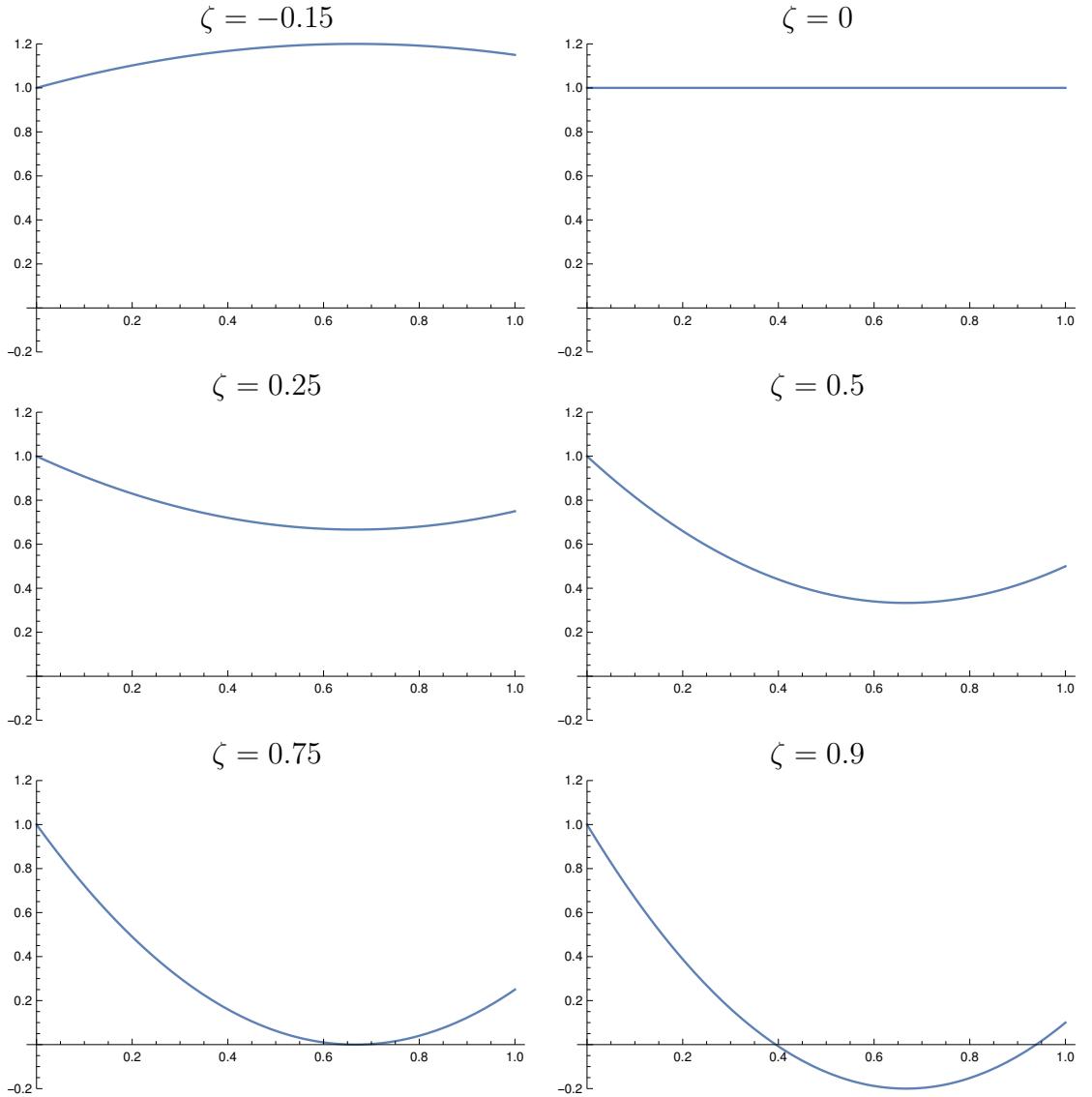
$$D = \frac{a^2}{\tau_0} [1 - \zeta \rho (4 - 3\rho)]. \quad (2.31)$$

Setting $\zeta \rightarrow 0$ (i.e. $\lambda = 1$), we see that $D \rightarrow \frac{a^2}{\tau_0}$, which is consistent with what

we would expect for the Symmetric Exclusion Process.

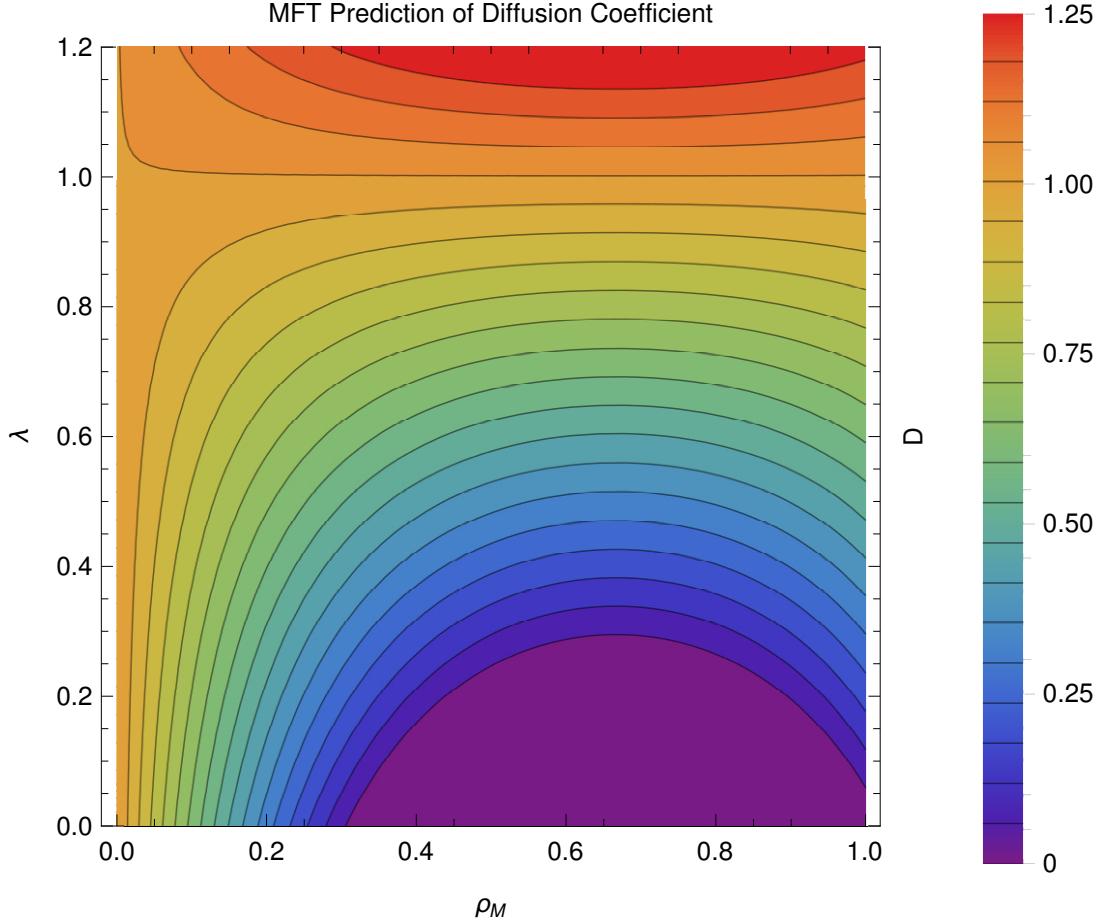
Clearly, the diffusion coefficient varies quadratically with ρ . This is easiest to see via a few graphs, as shown in Fig. 2.4. Note that $D \rightarrow \frac{a^2}{\tau_0}$ as $\rho \rightarrow 0$ and $D \rightarrow \frac{a^2}{\tau_0} \lambda$ as $\rho \rightarrow 1$, so for $\zeta < 0$ ($\lambda > 1$) D is guaranteed to be positive for $\rho \in [0, 1]$ as the diffusion coefficient is an inverted parabola so far as its variation in ρ is concerned.

Figure 2.4 Plots of the variation of $\frac{\tau_0 D}{a^2}$ (y-axis) with respect to ρ (x-axis), evaluated with various values of ζ (above plots).



Note that D has a symmetry in ρ around $\rho = \frac{2}{3}$, in the sense that D is unchanged under $\rho \mapsto \frac{4}{3} - \rho$. Why this symmetry is present in the MFT is a little unclear ($\rho \mapsto 1 - \rho$ would be a much more obvious choice), however as you will see in

Figure 2.5 A contour plot of the variation of $\frac{\tau_0 D}{\alpha^2}$ as a function of ρ and λ . The region with negative diffusion (which is really critically slow or zero diffusion due to our stability argument in 2.3.3) has been highlighted in purple. Note how as we descend in λ with $\lambda < \frac{1}{4}$, it grows from a single point at $\rho = \frac{2}{3}$ to fill most physically realistic density values.



the numerical simulations it does seem to be quite relevant, particularly in the high- λ limit.

2.3.3 Negative Diffusion Coefficients

A quick inspection of the dependence of the diffusion coefficient D upon ζ reveals that it is possible for strange things to happen in this MFT. For a given value of ζ , D is quadratic in ρ ; a natural question to ask is whether D is always positive, and if not, what the physical implications of this would be.

An easy way to do this is by analysing the roots of D . Writing it as a standard

quadratic,

$$D = \frac{a^2}{\tau_0} [3\zeta\rho^2 - 4\zeta\rho + 1] \quad (2.32)$$

which has discriminant $4\zeta\frac{a^4}{\tau_0^2} [4\zeta - 3]$. For a real quadratic, the discriminant changes sign when the solutions switch between being real and complex, which in our case is the difference between having real solutions and not having real solutions. Assuming that $\zeta > 0$ (as we know D is positive for $\rho \in [0, 1]$ for $\zeta < 0$), this change occurs when $\zeta = \frac{3}{4}$ corresponding to $\lambda = \frac{1}{4}$, so there are no real solutions for $\zeta < \frac{3}{4}$ and $\lambda > \frac{1}{4}$, and therefore D is guaranteed to be positive in these regions. Positive- D is the normal situation in physics, and a solution to the MFT PDE Eq.2.27 which contains only positive- D regions is at least self-consistent (although of course is only as good an approximation to the SPM as the continuum MFT assumptions allow).

When $\zeta > \frac{3}{4}$, D is negative so long as

$$\frac{2}{3} - \frac{\sqrt{\zeta(4\zeta - 3)}}{3\zeta} < \rho < \frac{2}{3} + \frac{\sqrt{\zeta(4\zeta - 3)}}{3\zeta}; \quad (2.33)$$

this is like a gap opening up in ρ when $\zeta > \frac{3}{4}$. At its maximal extent (when $\zeta = 1$), negative diffusion occurs for

$$\frac{1}{3} < \rho < 1, \quad (2.34)$$

so there is still a region where ρ is sufficiently low that negative diffusion does not occur.

In terms of what a negative diffusion coefficient actually means, consider a constant solution $\rho(x, t) = \rho_0$. Insertion Eq. 2.27 quickly confirms that this is indeed a solution. Now consider adding a small perturbation $\delta\rho(x, t)$ to ρ_0 . The equation for the time evolution of $\delta\rho$ then reads

$$\frac{\partial\delta\rho}{\partial t} = \frac{a^2}{\tau_0} [1 - \zeta\rho_0 (4 - 3\rho_0)] \frac{\partial^2\delta\rho}{\partial x^2}. \quad (2.35)$$

This becomes a little clearer if one takes a Fourier transform with respect to x , so that $\hat{\delta\rho}(k, t) = \mathcal{F}(\delta\rho(x, t))$; then, the equation of motion for $\hat{\delta\rho}$ is

$$\frac{\partial\hat{\delta\rho}}{\partial t} = -k^2 \frac{a^2}{\tau_0} [1 - \zeta\rho_0 (4 - 3\rho_0)] \hat{\delta\rho}. \quad (2.36)$$

This shows us that so long as $\zeta < \frac{3}{4}$, small perturbations to the density are suppressed by exponential decay in time with increasing ferocity as their wavenumber increases for all wavenumbers, and so the solution is stable; the same applies if $\zeta > \frac{3}{4}$ so long as we do not stray into situations where

$$\frac{2}{3} - \frac{\sqrt{\zeta(4\zeta - 3)}}{3\zeta} = \rho_- < \rho_0 < \rho_+ = \frac{2}{3} + \frac{\sqrt{\zeta(4\zeta - 3)}}{3\zeta}. \quad (2.37)$$

If we do find ourselves in this regime, small perturbations grow exponentially with time in a situation akin to ripening [find some decent references when possible], which, given that the particles are undergoing conserved flow, suggests that we will have a separation into regions with lower and higher densities. Of course, the positive feedback driving this separation stops if the density grows higher or lower than ρ_{\pm} , where we reenter the stable regime. This does suggest that in the MFT a system containing a negative- D region would have a tendency to self-organise itself into alternating domains, with at least the boundaries of these domains having densities of ρ_- or ρ_+ . This is very important: whilst it is no coincidence that these critical values of the density are those densities where our diffusion coefficient is zero, this does suggest that **a solution to the continuum MFT in the $\lambda < \frac{1}{4}$ regime which contains values for ρ in the critical gap $[\rho_-, \rho_+]$ should admit no current.** The search for this predicted effect is in fact the main driving force behind this entire PhD project.

2.3.4 Continuum Limit MFT Solutions

The continuum-limit MFT has given us a partial differential equation for $\rho(x, t)$; therefore, we should try to find some solutions to it, as these may give us clues as to what types of behaviour the SPM might exhibit.

Steady Flow Across a Bounded Domain

It's pretty obvious that $\rho = \rho_0 = \text{const.}$ is a solution to the MFT PDE, and it takes only a little thought to notice that this is in fact the only spatially homogeneous solution available. If we instead look for a solution which lacks time dependence (i.e. $\rho(x, t) = \rho(x)$), the PDE reduces to the ODE

$$-\frac{a^2}{\tau_0} \frac{d}{dx} \left([1 - \zeta\rho(4 - 3\rho)] \frac{d\rho}{dx} \right) = 0. \quad (2.38)$$

Integrating both sides with respect to x , and using the fundamental theorem of calculus, we find that

$$-\frac{a^2}{\tau_0} [1 - \zeta\rho(4 - 3\rho)] \frac{d\rho}{dx} = J_0, \quad (2.39)$$

with J_0 an arbitrary constant, which has been labelled as such in hindsight because it represents the constant current flowing through the system in a steady state. Doing so again, we find that we can invoke the chain rule via

$$J_0(x - x_0) = -\frac{a^2}{\tau_0} \int dx \frac{d\rho}{dx} [1 - \zeta\rho(4 - 3\rho)] \quad (2.40)$$

$$= -\frac{a^2}{\tau_0} \int d\rho [1 - \zeta\rho(4 - 3\rho)] \quad (2.41)$$

$$= -\frac{a^2}{\tau_0} \rho [1 + \zeta\rho(\rho - 2)] \quad (2.42)$$

Thus with a little rearrangement we have x as a function of ρ , with ρ a cubic in x . We can in principle invert this to obtain $\rho(x)$, but let us first consider the appropriate boundary conditions to use.

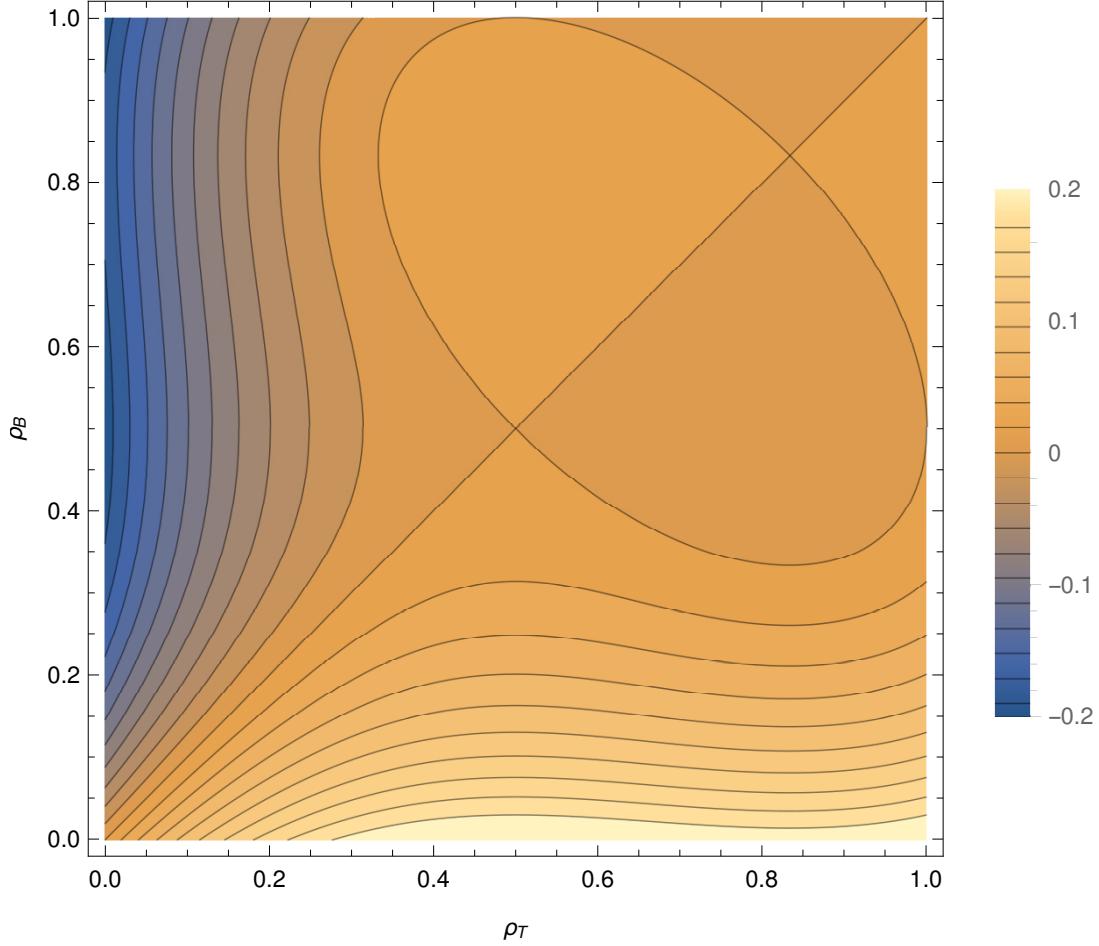
Let us consider solving problems on a bounded domain; we choose to do this as opposed to an infinite one, as one can see that for our cubic $\|x\| \rightarrow \pm\infty \implies \|\rho\| \rightarrow \pm\infty$ for nontrivial J . Therefore let us consider solutions on the domain $[0, L]$ for $L > 0$. With a second order ODE of this kind, we must supply two boundary conditions, which may be Dirichlet, Neumann or some mixture of the two, and must contain at least one piece of Dirichlet information. However, our ODE does not make any special reference to ρ values of 0 or 1, and therefore if we do not fix ρ at both boundaries it is highly likely that the solution will contain unphysical values for ρ . Therefore, let us apply Dirichlet conditions at both boundaries, so that $\rho(0) = \rho_0$ and $\rho(L) = \rho_L$. Inserting this information into Eq. 2.40 we can fix the constants x_0 and J_0 ; in particular we find that

$$J_0 = \frac{a^2}{L\tau_0} [\rho_0 - \rho_L + \zeta (\rho_0 [\rho_0^2 - 2] - \rho_L [\rho_L^2 - 2])], \quad (2.43)$$

which can be reinserted to yield the desired x_0 . An illustrative plot of $J_0(\rho_B, \rho_T)$ is shown in Fig. 2.6.

This solution in particular is extremely useful, as we can use it to predict the flow which should occur (MFT being correct) if we set up a numerical simulation of the SPM with a length of, say, L lattice points. In particular, if we vary λ whilst

Figure 2.6 A contour plot of the variation of the constant current $J_0(\rho_B, \rho_T)$ in a bounded domain with boundary densities ρ_0 and ρ_L at $x = 0$ and L with $\lambda = 0.2$. Notice how the magnitude of J_0 generally grows as the difference between ρ_0 and ρ_L increases, and how there is a region of boundary condition space in which the current takes the opposite sign one would expect.

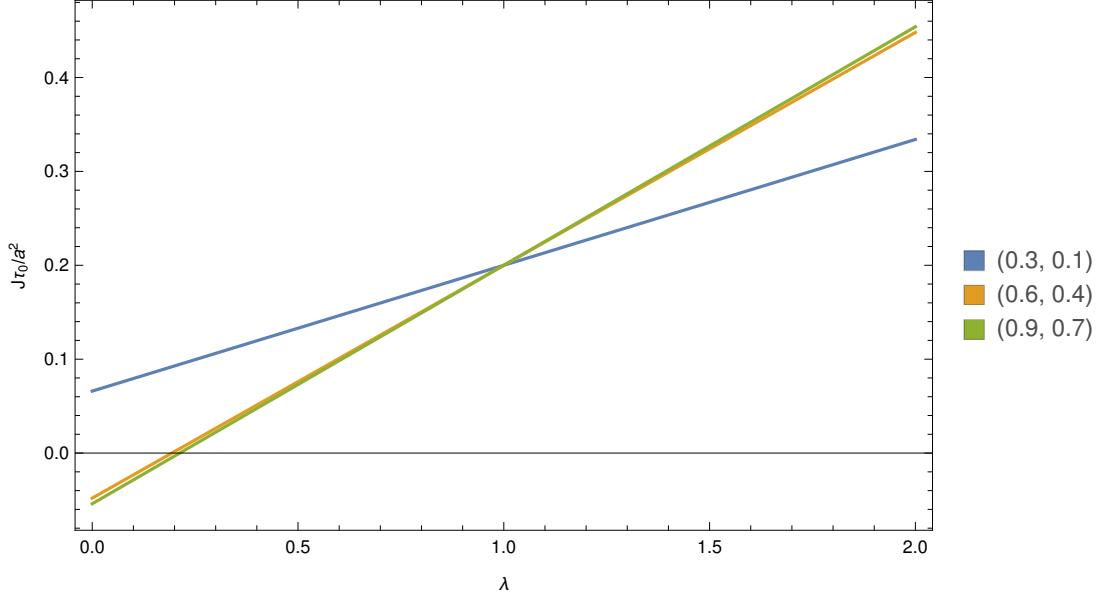


keeping the boundaries constant, the measured current should vary linearly, as depicted in Fig. 2.7. Thus, if we were to run simulations with, say, $(\rho_0, \rho_L) = (0.6, 0.4)$, we should see the transition to a backwards or critically slow flow occur. We can use Eq. 2.43 to find the critical value for λ , λ_c , at which the transition to negative diffusion should occur for given boundary conditions. To do this, we simply set $J_0 = 0$ and solve for λ , and find that

$$\lambda_c = 1 - \frac{1}{2(\rho_0 + \rho_L) - (\rho_0 + \rho_L)^2 + \rho_0 \rho_L}. \quad (2.44)$$

This is shown in Fig 2.8.

Figure 2.7 A plot of the MFT prediction of the dedimensionalised flow rate with varying λ for boundaries $(\rho_0, \rho_L) = \{(0.3, 0.1), (0.6, 0.4), (0.9, 0.7)\}$. Notice how the dependence of J on λ is actually very similar for the high and medium boundary-density-average situations, but is quite different for the low density case. Note that the MFT clearly predicts that the flow should start running backwards when λ becomes sufficiently low, which means that we should be able to see backwards or critically-slow flow in our numerics if we hold the boundaries constant whilst varying λ .



We can also obtain a prediction of the system-wide average density

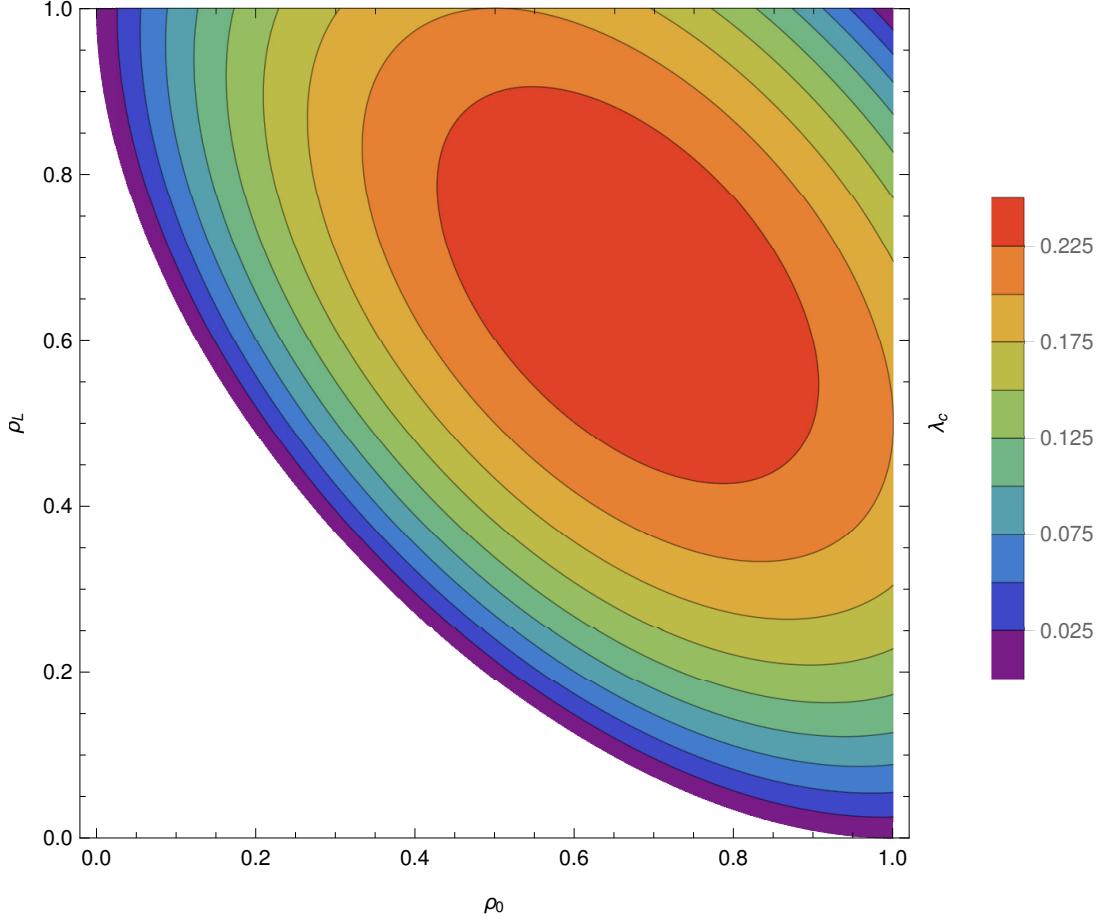
$$\bar{\rho} = \frac{1}{L} \int_0^L dx \rho(x), \quad (2.45)$$

so long as we can invert $x(\rho)$ to find $\rho(x)$ uniquely. The easiest way to do this is using the main result in [5], yielding

$$\bar{\rho} = \frac{6(\rho_0 + \rho_L) + \zeta [9(\rho_0^3 + \rho_L^3) - 16(\rho_0^2 + \rho_L^2) + \rho_0 \rho_L (9[\rho_0 + \rho_L] - 16)]}{12[1 + \zeta(\rho_0^2 + \rho_L^2 + \rho_0 \rho_L - 2[\rho_0 + \rho_L])]} \quad (2.46)$$

The variation of the average density with λ for selected fixed boundary conditions is plotted in Fig. 2.9. In general, this overall density deviates very little from the average of the two boundary densities.

Figure 2.8 A plot of the critical value λ_c , specified in Eq. 2.44 which λ must be smaller than in order to cause backward flow with boundary densities (ρ_0, ρ_L) . The region for which λ_c is negative is not included and marked in white, as $\lambda > 0$ for a physically realistic system. This shows that there are boundary configurations for which flow should still occur for arbitrarily small values of λ .

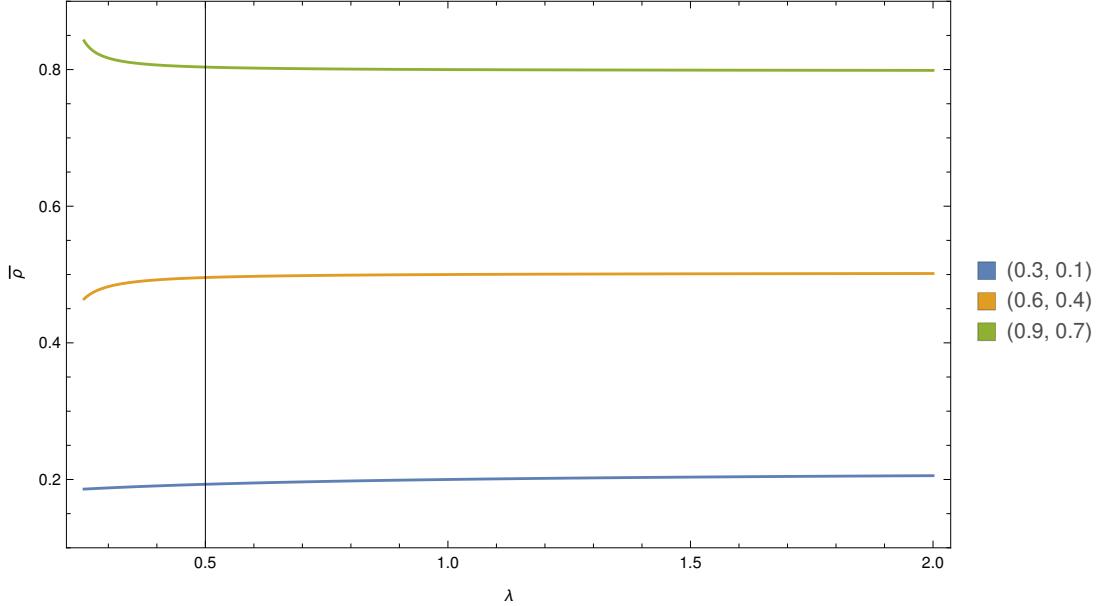


Other Analytic Solutions

Steady flow across a bounded domain is not the only solution for the continuum-limit MFT. We can also attempt to exploit Lie symmetries in the equations to generate solutions. Using the results in [4], we see that there should exist solutions of the form $\rho(x, t) = \phi(\omega)$ with $\omega = x - vt$ for some real v ; intuitively, this corresponds to a solution which simply translates through time with velocity v . After a little rearrangement, this implies that

$$v \frac{d\phi}{d\omega} = -\frac{a^2}{\tau_0} \frac{d}{d\omega} [1 - \zeta\phi(4 - 3\phi)] \quad (2.47)$$

Figure 2.9 A plot of the MFT prediction of the overall system-wide density with varying λ for boundaries $(\rho_0, \rho_L) = \{(0.3, 0.1), (0.6, 0.4), (0.9, 0.7)\}$. We have only plotted for $\lambda > \frac{1}{4}$, as outside this regime the MFT prediction is not unique, and so the inversion formula we need to calculate the density is not valid. In each case the density rarely deviates far from the average of the two boundary densities.



which may be integrated and then solved as a first order ODE to obtain

$$\omega = \frac{a^2}{\tau_0 v} \left[\frac{1}{2} \zeta \phi (8 - 6\mu - 3\phi) - (1 - \zeta [4 - 3\mu] \mu) \log(\phi - \mu) \right] + \omega_0, \quad (2.48)$$

where ω_0 and μ are constants.

Now we need as usual to consider what kind of boundary conditions to use. For simplicity, let us consider a wave of density travelling into an empty region; this implies that $\phi \rightarrow 0$ as $\omega \rightarrow \infty$. The only way to achieve this is by setting $\mu = 0$, leaving us with

$$\omega(\phi) = \frac{a^2}{v\tau_0} \left[\frac{1}{2} \zeta \phi (8 - 3\phi) - \log \phi - \frac{5}{2} \zeta \right] + \omega_0. \quad (2.49)$$

As we can vary ω_0 to shift solutions around in ω essentially arbitrarily, we can choose where ϕ takes a desired value. For convenience, let's make ϕ take the value 1 at $\omega = 0$, which is easily achievable by setting $\omega_0 = 0$. At this point, we are faced with the prospect of trying to invert Eq. 2.49. This would be annoying, although one could be assisted by numerics to lighten the load. However, we can gather plenty of information simply by taking some limits. As $\omega \rightarrow \infty$, $\phi \rightarrow 0$

by design, and so $\phi(\omega) = \mathcal{O}(e^{-\frac{v\omega\tau_0}{a^2}})$; having an exponential tail at the leading edge of the wave, with a thickness proportional to the default diffusion coefficient divided by the wave speed, makes perfect sense. Meanwhile, by considering small variations in ϕ around 1, we may derive that $\phi \sim 1 - \frac{v\tau_0}{a^2\lambda}\omega$ as $\omega \rightarrow 0$. One could see this as being a front, behind which the system is filled by a blast wave moving with velocity v . Notice how both limits suggest that the leading edge of the wave becomes thinner with increasing wavespeed, whilst close to the full region it thickens in proportion to λ .

However, we are left with the problem that we have the free parameter v in the solution. One might hope that we can find the preferred value for v by means of a speed-selection argument as is does with the Fisher-KPP equation [11], but that does not work in this case as the wave tail thickness is monotonic in the wave speed. Therefore, the wave speed seems to be dictated by the initial conditions, which allows some rather bizarre behaviour. For example, in the limit $v \rightarrow +\infty$, the initial condition (in other words, setting $t = 0$ and observing that $\rho(x, 0) = \phi(x)$) resembles an inverted Heaviside step function for $x > 0$; thus this suggests that if we were to start the system with initial condition $\rho(x, 0) = 1 - H(x)$, the high-density region would advance into the low-density region with infinite velocity, regardless of a , τ_0 or λ . This seems somewhat unphysical, and serves as a reminder that the MFT is a guide only, and shouldn't be expected to accurately predict the behaviour of the SPM.

Using [4] there is one last type of solution based upon symmetry. To acquire it, let us define $\xi = \frac{x}{\sqrt{t}}$ and $\rho(x, t) = \chi(\xi)$; then our PDE reduces to

$$\xi \frac{d\chi}{d\xi} = -2 \frac{a^2}{\tau_0} \frac{d}{d\xi} [1 - \zeta\chi(4 - 3\chi)]. \quad (2.50)$$

Taking $\zeta = 0$, this clearly reduces to the standard similarity solution of the diffusion equation as we would expect, so this is the nonlinear analogue of that. Unfortunately, this is a nonlinear second order ODE which isn't particularly amenable to solution, so after some solution attempts we decided to give up at this point and focus on numerics and other analytic avenues.

2.3.5 Implications of Continuum MFT Breakdown

We have already mentioned that the MFT can predict negative diffusion coefficients for $\lambda < \frac{1}{4}$, $\rho \in (\rho_-, \rho_+)$ as defined in Eq. 2.37. However, we should investigate this in a little more detail, as it has testable implications for the SPM.

Treating the MFT at face value, our stability analysis in 2.3.3 suggests that in the event that we have a region with $\rho \in (\rho_-, \rho_+)$ there should be a tendency for the medium to separate into parts which have $\rho \notin (\rho_-, \rho_+)$; of course, the first ρ s for which $\rho \notin (\rho_-, \rho_+)$ are ρ_- and ρ_+ themselves, which are the values for which the diffusion coefficient, and therefore the current resulting from a concentration gradient is zero. So, the process of the medium separating should in general yield a mixture of regions with $\rho = \rho_-$, $\rho = \rho_+$ and other $\rho \notin (\rho_-, \rho_+)$, mixed in such a way that the total number of particles is locally conserved.

It is this nonuniqueness of configuration which causes us some problems if we try to accept the MFT as a good descriptor of SPM phenomenology. In reference to our steady state solution described in 2.3.4, note that $\rho(x)$ is only unique so long as we avoid negative diffusion, otherwise, the cubic inversion we need to perform to transform $x(\rho)$ into $\rho(x)$ is multivalued. One could imagine that we could fix this by patching together sections which cross with $\rho = \rho_-$ or ρ_+ , but then we have essentially unlimited choice of how large to make the sections and how many alternations to include. This means that the MFT makes no prediction of the system-wide average density $\bar{\rho}$ which is unfortunate as this is a quantity which it is easy for us to measure using our numerics.

2.4 The SPM in Higher Dimensions

We initially designed the SPM for use in one dimension, as it was originally intended to represent interacting particles moving along a periodic potential with deep, narrow wells. However, it is only natural to wonder whether a similar model could be constructed in higher dimensions. Recall that the SPM in one dimension was designed to have two properties:

- Left-right symmetry, and
- locality, in the sense that only the presence or absence of a particle in an adjacent lattice site may influence the transition rate.

In addition we also proved that (boundary conditions aside) the SPM also obeys detailed balance. This was not put into the model intentionally, but emerges naturally as the space of possible one-dimensional models which are local and symmetric is very small. Let us consider only square lattices in n dimensions, for simplicity. If we attempt to build a model in two dimensions which is symmetric and local, (i.e. obeys all the point group symmetries of the underlying lattice, and whose transitions are only influenced by the immediate environment around a particle), we find that we now have more freedom in the model construction than we did in one dimension. For example, in two dimensions a moving particle might be leaving any one of six possible unique local configurations, as shown in Fig. 2.10, and so such a model would need to be parametrised by 5 rates, once we take time dilation symmetry into account. The number of possible symmetric local hopping models only grows greater in higher dimensions, and this makes it very difficult to analyse the parameter space of such models using numerics. Therefore, we have chosen to investigate most closely those models which in addition obey detailed balance; as it turns out, there is only one such model, regardless of the number of dimensions.

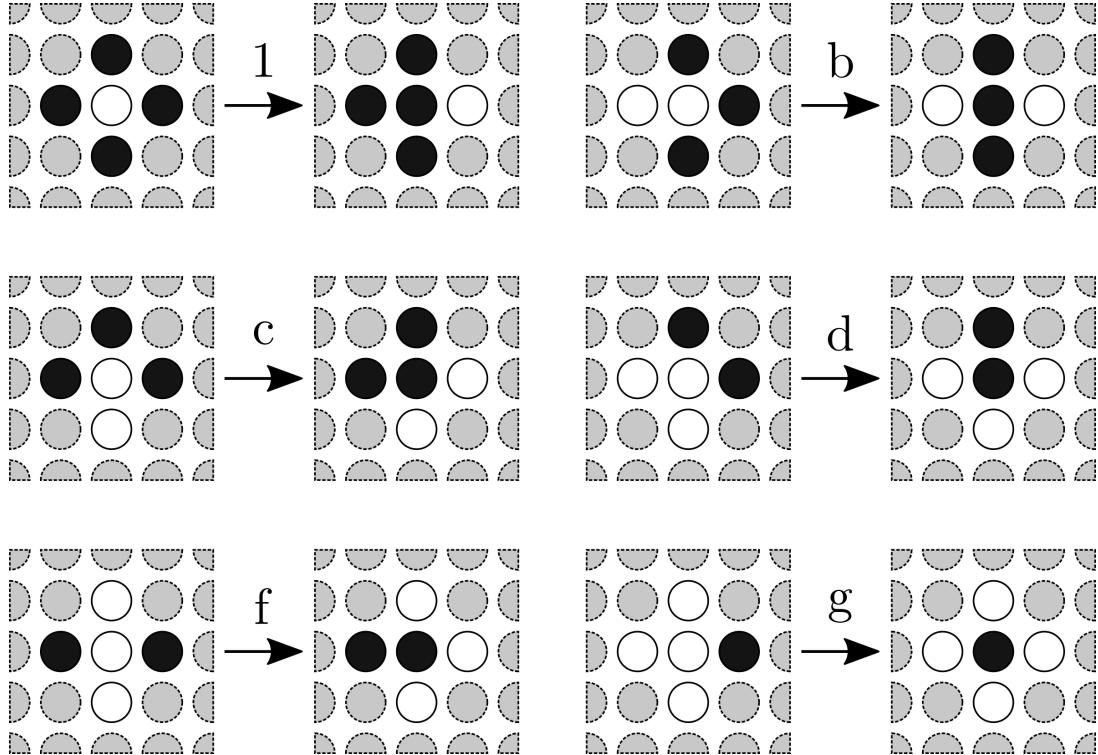
2.4.1 Symmetry + Locality + Detailed Balance = Unique 1-Parameter SPM

When investigating higher-dimensional analogues of the SPM, we initially considered a generic model as shown in Fig. 2.10; however, as the parameter space is so large, we decided to attempt to impose the detailed balance condition and see how much freedom that gave the model. In the end, after some exceedingly tedious casework, we found that in order to obey detailed balance particles needed to move with transition rate specified by the following theorem, which applies in arbitrary numbers of dimensions:

Theorem 1. *Any processes defined on an n -dimensional square lattice in which particles swap places with vacancies and whose transition rates $\sigma(\xi_1 \rightarrow \xi_2)$*

- *are symmetric (invariant under rotations, reflections and translations of the underlying lattice),*
- *local (rate at which a particle swaps with a vacancy only depends upon configuration of particle's immediate neighbours),*

Figure 2.10 The unique available moves in a 2-dimensional symmetric local hopping model. Note that we have rescaled time so that the free particle hopping rate is 1, for consistency with the 1-dimensional SPM. Simultaneous rotations or reflections of both initial and final states are also allowed moves with the same rates. White corresponds to a lattice site occupied by a particle, black to an unoccupied site, and grey to a site which has no effect on the transition rate. In each case a particle moves into an empty space to its right, with its different starting environment determining the transition rate.



- and obey detailed balance, i.e. that \exists probability distribution P over configurations $\xi \in \Xi$ such that $\forall \xi_1, \xi_2 \in \Xi$,

$$P(\xi_1)\sigma(\xi_1 \rightarrow \xi_2) = P(\xi_2)\sigma(\xi_2 \rightarrow \xi_1) \quad (2.51)$$

must have a transition rate of the form

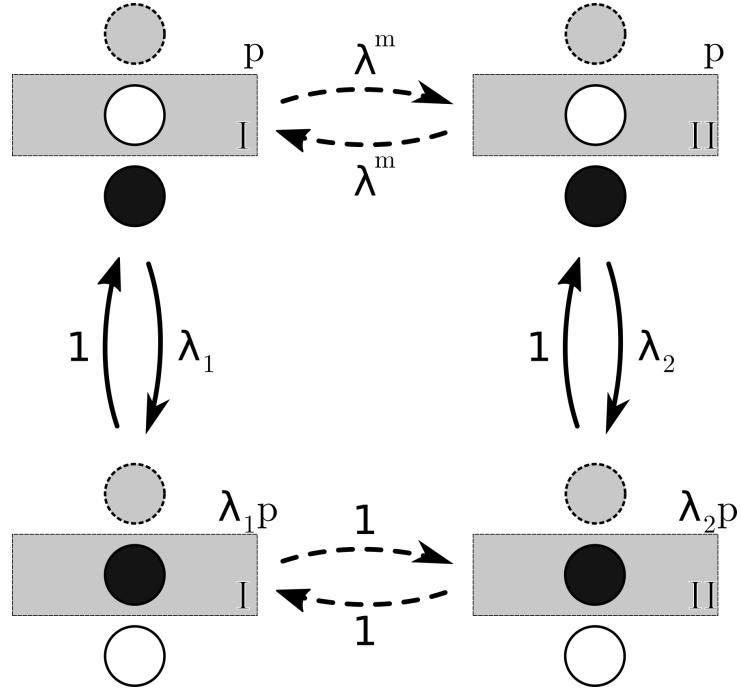
$$\sigma(\xi_1 \rightarrow \xi_2) = \frac{1}{\tau_0} \lambda^m, \quad (2.52)$$

where m is the number of particles directly adjacent to the particle which is attempting to swap with a vacancy and τ_0 is an arbitrary constant.

Proof. We have already shown that the theorem applies in 1-dimension, as there

the space of possible models is so constrained [reference in introduction] that there is only one symmetric local model, which is the SPM. This model, as we showed, obeys detailed balance anyway, so we're done for $n = 1$.

Figure 2.11 *I and II represent two local configurations of m particles, centred around a single site, which may or may not be occupied, in an $(n - 1)$ -dimensional hyperplane intersecting the central site. The dotted lines dashed arrows represent configurational transitions which require multiple steps, all with the same rate (in this case, λ) as indicated. The grey particle slot represents a slot whose occupation is irrelevant for this calculation, although it must be constant throughout. The expressions in the top right corners of the configurations are the probabilities of our finding the system in that configuration.*



We will proceed with the proof via induction. Let us assume that the result is true in $(n - 1)$ dimensions. Firstly, we need to show that the actual configuration of the particles in contact with a particle does not affect its rate of motion into an adjacent free space. Fig. 2.11 shows just such a situation: there are m particles clustered around the central particle in an $(n - 1)$ -dimensional hyperplane, but in two different local configurations, I and II. Because the model in $(n - 1)$ -d obeys detailed balance, we can consider a reversible chain of moves from one configuration to the other. We do this by moving each particle adjacent to the central particle outwards, with rate λ . Once they are all separated from the central particle, we can then move them around at leisure, with each move having

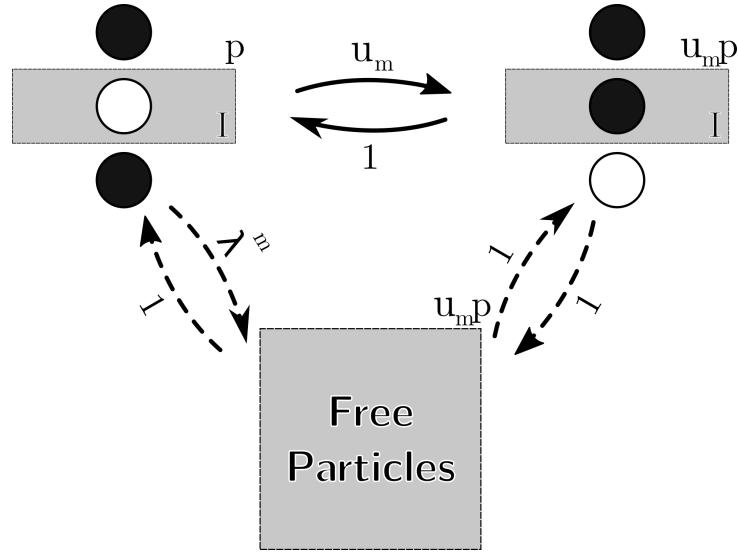
rate 1 so long as we keep them separate; luckily, there is plenty of room to do this, as we can simply move the particles further away using only rate 1 moves. We can then rebuild the particles from, say, configuration I into their exact state in configuration II. Furthermore, we can do exactly the same thing going from configuration II to configuration I, again using only m rate λ moves and the rest with rate 1. Importantly, if a system obeys detailed balance, then if we can move from one state to another and back again with the same rate, they must have the same equilibrium occupation; therefore, the probability of finding the system in configuration I and the probability of finding it in configuration II must be equal, otherwise the system would violate detailed balance (which the model which we are assuming the system obeys in $(n - 1)$ -d satisfies). However, there is another way to go from state I to state II: we can move the central particle, which we will say occurs with rates λ_1 and λ_2 respectively. Once the central particle is moved, we now have a “gas” of free particles, which we can once again move as we like using only moves of rate 1, in order to deform the system into configuration II. As the probability of our system being in state II is the same as it being in state I, the only way detailed balance can be obeyed is if $\lambda_1 = \lambda_2$. Therefore, the rate at which a particle moves into an adjacent empty space cannot depend upon the configuration of its adjacent particles in a given $(n - 1)$ -dimensional hyperplane, only their number.

Now we need to consider two cases, as the slot directly behind the central particle’s proposed direction of motion can be either empty or full. Firstly, the empty case is considered in Fig. 2.12. This time, we are highlighting the fact that if we move the central particle out of a hyperplane configuration of m particles, which occurs with rate u_m , this results in a gas of free particles; equivalently we could’ve produced a gas by moving the adjacent particles one by one as before, which requires m steps with each with rate λ . As the two gases are equivalent, in detailed balance terms, they must have the same realisation probability. If the probability of our being in the initial configuration (top left) was p , then by detailed balance the probability of our being in a free particle state is simultaneously both pu_m and $p\lambda^m$, and so $pu_m = p\lambda^m$ implying that

$$\lambda^m = u_m. \quad (2.53)$$

In Fig. 2.13, we show exactly the same thing, only this time there is a particle behind the central particle, and so an additional move must be made in order to create a gas of free particles. This means that the equivalence is now between

Figure 2.12 Here we've used the same diagram rules as in Fig. 2.11.

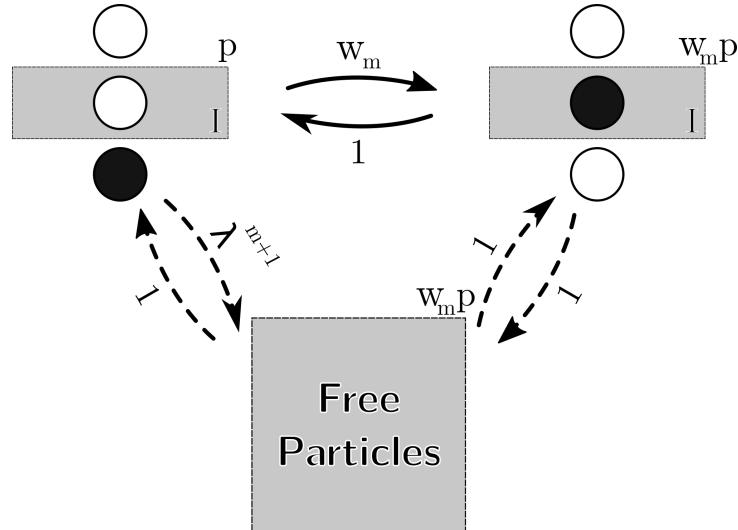


$p u_m$ and $p \lambda^{m+1}$, and so

$$\lambda^{m+1} = w_m. \quad (2.54)$$

With the constraints that detailed balance imposes upon u_m and w_m , we find that the original proposition for dimension n is true so long as it is for dimension $(n - 1)$ for $n > 1$, and as we know it is true for $n = 1$, the theorem is proved.

Figure 2.13 As Fig. 2.12, but now with a particle behind the central one.



□

Theorem 1 implies that an n -dimensional exclusion model which is symmetric,

local and obeys detailed balance must have an energy (defining the equilibrium distribution) which is proportional to the number of particle-particle adjacencies in the system; this therefore limits energy to being contained in the “bond” between adjacent particles, leaving us only with the option of a simple additive bond energy. This means that in order to have a more complicated energy, our only option is to either break symmetry or have a system of rates which considers more than the immediate environment of a particle which is attempting to move, for example considering the environment to which the particle is going.

The advantage of possessing this result is that it highlights that in n -dimensions there is again a special, simple one-parameter model which has many symmetry properties and is therefore a good first target for study. As such, we will perform a mean-field analysis of the n -dimensional SPM, to give us analytical results to compare to later Monte-Carlo numerics.

2.4.2 MFT of the n -Dimensional SPM

In the same manner as in Sec. 2.3.1, we will let $\rho_\chi(t)$ be the mean occupation of the χ^{th} site at time t , where $\chi \in \mathbb{Z}^n$. Let the unit vector in the i^{th} direction be $e_i \in \mathbb{Z}^n$, so that $\rho_{\chi+e_i}$ refers to a lattice site adjacent to the χ^{th} and offset in the i^{th} direction. Then, making the normal MFT assumption (that means of products are products of means), we can say that the rate at which a particle moves from site χ to site $\chi + e_i$ is

$$\begin{aligned} \sigma_{\chi \rightarrow \chi + e_i} &= \rho_\chi(1 - \rho_{\chi + e_i}) [(1 - \rho_{\chi - e_i}) + \lambda \rho_{\chi - e_i}] \\ &\times \prod_{j \neq i}^n [(1 - \rho_{\chi + e_j})(1 - \rho_{\chi - e_j}) + \lambda \rho_{\chi + e_j}(1 - \rho_{\chi - e_j}) + \lambda \rho_{\chi - e_j}(1 - \rho_{\chi + e_j}) + \lambda^2 \rho_{\chi + e_j} \rho_{\chi - e_j}] \\ &= \rho_\chi (1 - \rho_{\chi + e_j}) [1 - \zeta \rho_{\chi - e_j}] \prod_{j \neq i}^n [(1 - \zeta \rho_{\chi - e_j})(1 - \zeta \rho_{\chi + e_j})], \end{aligned} \tag{2.55}$$

with $\zeta = 1 - \lambda$ as usual.

Again, we will move to a continuum formulation. To do this, we are best off considering the overall flow between the site at χ and the site at $\chi + e_i$, and then Taylor expanding $\rho(\mathbf{x}, t)$ as a continuous variable. Doing this analytically in arbitrary dimensions is extremely tedious, and more importantly error-prone; thus, computer assistance is useful. A code which calculates this MFT current

Table 2.1 The MFT current in the SPM, as a function of the density gradient, for the first 4 dimensions.

| n | Simplified Current, $-\mathbf{J} \frac{\tau_0}{a^2}$ |
|---|--|
| 1 | $[1 - \zeta\rho(4 - 3\rho)] \nabla\rho$ |
| 2 | $(1 - \zeta\rho)^2 [1 - \zeta\rho(6 - 5\rho)] \nabla\rho$ |
| 3 | $(1 - \zeta\rho)^4 [1 - \zeta\rho(8 - 7\rho)] \nabla\rho$ |
| 4 | $(1 - \zeta\rho)^6 [1 - \zeta\rho(10 - 9\rho)] \nabla\rho$ |

for given n to $\mathcal{O}(a^3)$ may be found in Sec. A.2.

Computing this current for a few low values of n , a pattern emerges, as one can see in Tab. 2.1. Thus one can see that the MFT current in n -dimensions is

$$\mathbf{J} = -\frac{a^2}{\tau_0} (1 - \zeta\rho)^{2(n-1)} [1 - \zeta\rho((2n+2) - (2n+1)\rho)] \nabla\rho, \quad (2.56)$$

where the current and density obey the usual continuity equation

$$D = \frac{\partial\rho}{\partial t} + \nabla \cdot \mathbf{J} = 0 \quad (2.57)$$

with $\mathbf{J} = -D(\rho)\nabla\rho$, D being the diffusion coefficient

$$D(\rho) = \frac{a^2}{\tau_0} (1 - \zeta\rho)^{2(n-1)} [1 - \zeta\rho((2n+2) - (2n+1)\rho)]. \quad (2.58)$$

The equivalent of our 1-dimensional steady state solution may be found by considering a flow from one hyperplane to a parallel one a distance L away. Taking the planes to be separated in the x direction, we find that we have an equivalent solution, where the homogeneous current J_0 is given by

$$J_0 = \int_{\rho_L}^{\rho_0} d\rho \frac{a^2}{L\tau_0} (1 - \zeta\rho)^{2(n-1)} [1 - \zeta\rho((2n+2) - (2n+1)\rho)], \quad (2.59)$$

where ρ_0 and ρ_L are as usual the densities on the bounding hyperplanes. The density profile is the implicit solution of

$$\int_{\rho}^{\rho_0} d\rho \frac{a^2}{\tau_0} (1 - \zeta\rho)^{2(n-1)} [1 - \zeta\rho((2n+2) - (2n+1)\rho)] = J_0 x. \quad (2.60)$$

The rest of the boundary conditions can be pretty arbitrary. The domain might

extend infinitely parallel to the hyperplanes, or is could be finite, or on a periodic domain; whichever types of boundary are used, the boundary density must be chosen to match the density specified by the homogeneous solution there.

Limiting and Symmetry Properties of the n -Dimensional MFT

Now that we have derived the diffusion coefficient D in the n -dimensional MFT, we can investigate some special cases, in a similar manner to the 1d situation.

As $\rho \rightarrow 0$, $D \rightarrow 1$, which makes sense as in extremely low-density situations the particles do not interact. As $\rho \rightarrow \frac{a^2}{\tau_0}$, $D \rightarrow \lambda^{2(n-1)} \frac{a^2}{\tau_0}$, which again corresponds to the diffusion of vacancies in an almost-full lattice (which depends upon almost fully-bound particles needing to jump in order for a vacancy to move).

When $\lambda \rightarrow 1$, we find that $D \rightarrow \frac{a^2}{\tau_0}$, which makes perfect sense as turning off the interactions gives us normal diffusion again. As $\lambda \rightarrow 0$,

$$D(\rho) \rightarrow (1 - \rho)^{2(m-1)} [1 - (2m + 1)\rho], \quad (2.61)$$

suggesting that in that limit negative diffusion occurs at densities higher than $\frac{1}{2m+1}$. In terms of the behaviour for very large λ , one can show that

$$D(\rho) = \lambda^{2m-1} \rho^{2(m-1)} [(2m + 1)\rho - (2m + 2)] + \mathcal{O}(\lambda^{2(m-1)}), \quad (2.62)$$

so we should expect to see currents of $\mathcal{O}(\lambda^{2m-1})$ for large λ in arbitrary dimensions. Of course, the number of sites adjacent to a particle attempting to move into an adjacent vacancy is $2m - 1$, so in the large λ limit $\mathcal{O}(\lambda^{2m-1})$ is the biggest speedup we could reasonably expect to see, and the MFT reflects that.

Unlike the 1d situation, there is no longer a symmetry in the density dependence of the diffusion coefficient. This is because the $(1 - \zeta\rho)^{2(n-1)}$ bracket is symmetric about $\rho = \zeta^{-1}$, whereas the $[1 - \zeta\rho((2n + 2) - (2n + 1)\rho)]$ bracket's symmetry is about $\rho = \frac{n+1}{2n+1}$; thus, in general they would not share a point of symmetry, and so their product would not be symmetric.

Finding the density at which flow is extremal is annoying, but doable. Differen-

tiating the diffusion coefficient with respect to ρ gives, after factorising,

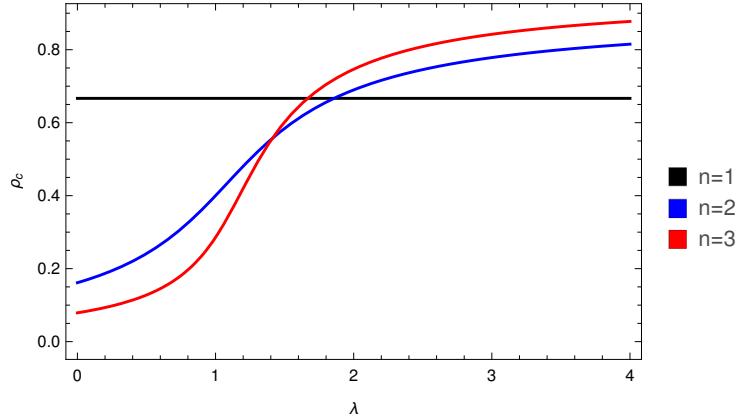
$$\frac{\partial D}{\partial \rho} = -2\zeta(1 - \zeta\rho)^{2n-3} [\zeta\rho(n(2n+1)(\rho-1) + 1) - 2n(\rho-1) - \rho]. \quad (2.63)$$

The first bracket has its zero when $\rho = \zeta^{-1}$, which means that this only causes a turning point for allowed ρ when $\zeta > 1 \implies \lambda < 0$, which is unphysical. The second bracket is a quadratic in ρ . Its root, and therefore the extremal value, occurs at ρ_c where

$$\rho_c = \frac{4}{2\zeta n^2 + (3 - \lambda)n + \lambda + \sqrt{(2\zeta n^2 + (3 - \lambda)n + \lambda)^2 - 8\zeta n(2n+1)}}. \quad (2.64)$$

The extremum is a minimum for $\lambda < 1$ and a maximum for $\lambda > 1$. The variation of ρ_c with λ is shown in Fig. 2.14. In terms of interpretation, this extremum determines where maximal flow occurs for $\lambda > 1$. As $\lambda \rightarrow \infty$, $\rho_c \rightarrow \frac{4n^2+2n-2}{2n(1+2n)}$, which approaches 1 as the number of dimensions becomes very large.

Figure 2.14 *The variation of the ρ for which flow is extremal as a function of λ , for $n = \{1, 2, 3\}$. Notice that it starts low and ends high in the 2 and 3-dimensional cases, but is constant in the 1d case.*



In all numbers of dimensions, the SPM MFT can produce negative flows. To find out which conditions are required for this, we must solve $D(\rho) = 0$ for $\rho \in (0, 1)$. Luckily we already have D in a factorised form in Eq. 2.58. Assuming $\zeta < 1$, the first bracket cannot have a solution on $(0, 1)$, therefore we seek the zeros of the second bracket. The discriminant of the quadratic in the second bracket is

$$\zeta^2(2n+2)^2 - 4\zeta(2n+1), \quad (2.65)$$

thus the crossover between the existence and nonexistence of real solutions to $D(\rho) = 0$ occurs when $\zeta = \frac{2n+1}{(n+1)^2}$. Using our result for the dependence of ρ_c upon λ (Eq. 2.64), we see that as we reduce λ from 1, backwards diffusion first occurs for

$$\rho_c = \frac{4(n+1)}{(n+1)(6n-3) + 3 + (2n+1)\sqrt{n(9n-8)}} \quad (2.66)$$

As we have seen, this “gap” of negative diffusion begins at $\rho_c = \frac{2}{3}$ in 1d, and then at $\rho_c \sim 0.22$ in 2d and $\rho_c \sim 0.13$ in 3d.

The SPM MFT in 2-Dimensions

Specifically, in the 2-dimensional case, J_0 is given by

$$J_0 = \frac{a^2}{L\tau_0} \left[\zeta^3 (\rho_0^5 - \rho_L^5) + \frac{1}{2} \zeta^2 (3\zeta + 5) (\rho_L^4 - \rho_0^4) + \frac{1}{3} \zeta (13\zeta + 5) (\rho_0^3 - \rho_L^3) + 4\zeta (\rho_L^2 - \rho_0^2) + (\rho_0 - \rho_L) \right]. \quad (2.67)$$

The dependence of flow upon the boundary conditions for some given λ is shown in Figure. 2.15, and the dependence of flow upon λ for some fixed boundaries is shown in Fig. 2.16. We can use these in comparison with our Monte-Carlo data in 2-dimensions.

2.5 Conclusions About the SPM MFT

In this Chapter we have derived a whole host of results about the continuum limit Mean-Field Theory of the Sticky Particle Model defined in arbitrary dimensions. The key things to take away from it are:

- Thm. 1 shows us that the n -dimensional SPM (as defined within it) is the only model defined on an n -dimensional square lattice which is symmetric, local and obeys detailed balance.
- The MFT of this model **always** predicts a transition in which the diffusion coefficient becomes negative for some physically-allowed ρ and λ , regardless of dimension.
- Currents for very large λ should scale as λ^{2m-1}

Figure 2.15 Plots of the flow in the MFT of the 2-dimensional SPM, where we are varying the boundary densities. The values of λ used are $\{0.2, 0.4, 0.75, 2\}$, going clockwise from top left.

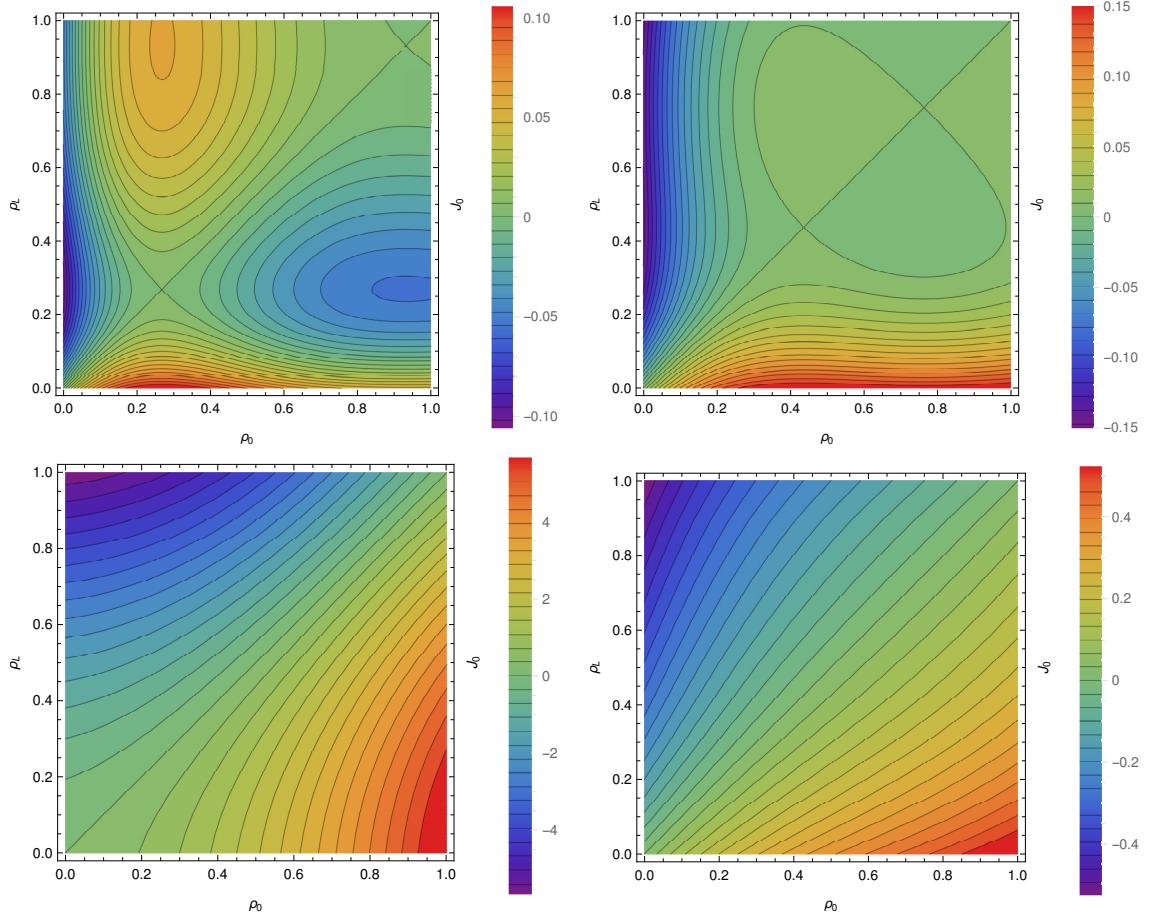
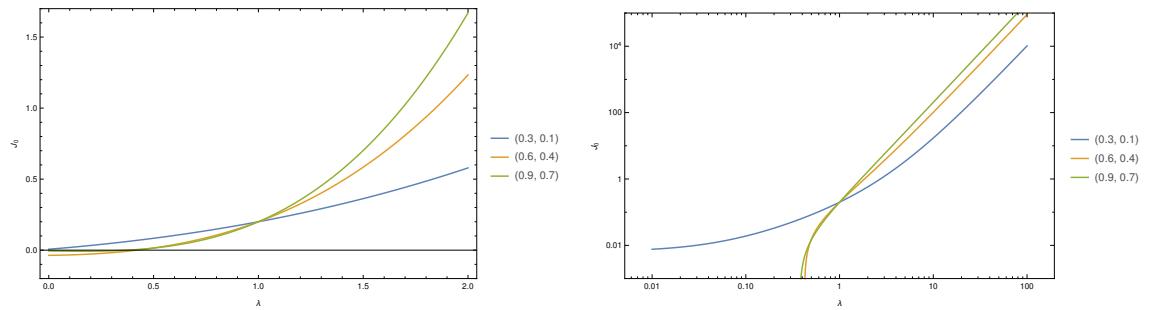


Figure 2.16 Plots of the current variation with respect to λ in the MFT of the 2-dimensional SPM. Boundary conditions are labelled in the legend in the form (ρ_0, ρ_L) .



Chapter 3

Transition Rate Matrix Analysis

Now that we have MFT predictions about the relationship between density difference and current in the SPM, it would be good to try to investigate their validity. In Chapter 4 we will use Monte-Carlo methods to do this in 1d and 2d, but in this chapter we will restrict our attention to 1d.

3.1 The Transition Rate Operator for the SPM

The SPM is an autonomous continuous-time Markov Process, which describes continual transitions between states with transition rates depending only upon the current state. As such, if we call the total space of states Ξ then the probability distribution $P : \Xi \times \mathbb{R} \rightarrow \mathbb{R}$ should obey a **master equation**

$$\frac{\partial P(\xi, t)}{\partial t} = \mathcal{A}P(\xi, t), \quad (3.1)$$

where $\mathcal{A} : \Xi \rightarrow \Xi$ is the **transition rate operator** or TRO. Note that I am going to be using column vectors for probabilities rather than row vectors, as many in the probability community do. Parametrising \mathcal{A} via

$$(\mathcal{A}f)(u) = \int_{\Xi} d\xi \sigma(u, \xi) f(\xi) \quad (3.2)$$

puts it in a more familiar, transport equation-style notation:

$$\frac{\partial P(\xi, t)}{\partial t} = \int_{\Xi} du \sigma(\xi, u) P(u, t). \quad (3.3)$$

Here $f : \Xi \rightarrow \mathbb{R}$ is an arbitrary function, intended to be a probability distribution (nonnegative, unit measure, etc), whilst $\xi, u \in \Xi$ are dummy variables and $\sigma : \Xi \times \Xi \rightarrow \mathbb{R}$ represents \mathcal{A} in the basis determined by how we perform the integration.

We demand that σ satisfies

$$\forall \xi \in \Xi, \sigma(\xi, \xi) \leq 0 \quad (3.4)$$

and

$$\forall \xi_1, \xi_2 \in \Xi : \xi_1 \neq \xi_2, \sigma(\xi_1, \xi_2) \geq 0, \quad (3.5)$$

as well as the constraint

$$\forall u \in \Xi, \int_{\Xi} d\xi \sigma(\xi, u) = 0. \quad (3.6)$$

The last constraint implies that

$$\int_{\Xi} d\xi \frac{\partial P(\xi, t)}{\partial t} = \int_{\Xi} du P(u, t) \left[\int_{\Xi} d\xi \sigma(\xi, u) \right] = 0 \quad (3.7)$$

regardless of the structure of P , which is our probability conservation equation.

The formal (forward-time) solution to Eq. 3.1 is given by

$$P(\xi, t) = e^{(t-t_0)\mathcal{A}} P_0, \quad (3.8)$$

where t_0 is some initial time, P_0 is the starting distribution, and the operator exponential is defined by its Taylor expansion, which should converge fine for bounded \mathcal{A} , satisfied for the finite-system SPM. As $e^{(t-t_0)\mathcal{A}}$ and \mathcal{A} share eigenvectors, we see that the eigenstructure of \mathcal{A} is something well worth investigating, as it should give us information about the time-evolution of the system. An important thing to point out is that \mathcal{A} does not in general have orthogonal eigenvectors because it is not in general symmetric, and so we cannot normally diagonalise it using orthogonal transformations. This means that modes do not “decouple” in the way that states do in the Schrödinger equation, and we instead have to deal with an **adjoint system**.

Luckily, when it comes to the analysis of steady states, there are a few results that can help us. If we consider the operator $\mathcal{G}_T = e^{T\mathcal{A}}$ (in other words, the **propagator** for a period of time T), it is pretty easy to see that this is a standard Markov Operator, as we are essentially reversing the limiting process we would

perform in order to define a continuous time Markov process as a limit of a discrete time one. If we assume that Ξ is finite and \mathcal{G}_T is irreducible then as a Markov operator it must possess a unique eigenvector with corresponding eigenvalue 1. All other eigenvalues must have modulus between 0 and 1. Therefore, \mathcal{A} must share that same unique eigenvector with associated eigenvalue 0, and its other eigenvectors must have negative real part. In physical terms, the system must have a single steady state probability distribution, which it always relaxes towards exponentially quickly, with a rate determined by the nonzero eigenvalue of \mathcal{A} with real part closest to 0.

Of course, for such finite-state systems (such as the SPM on a finite domain) the integrals become sums, and σ a matrix, Q . In such systems, we can arrange to have some labelling scheme which uniquely relates system states to natural numbers, and therefore relates states to basis vectors in a vector space. In the SPM, a site is either full or empty, which means that there is a natural mapping between states and natural numbers based upon binary representation; a string of 1s and 0s can be associated with a natural number as well as a configuration of particles and vacancies.

3.1.1 A Small Worked Example: Closed System

As a concrete example, let us consider the SPM on a cyclic domain of length 3. There are 2^3 possible combinations, and so 8 possible states: 000, 001, and so forth. The transition rate matrix describing this system is:

$$Q = \begin{bmatrix} 0 & & & & & & & \\ -2 & 1 & & & & & & \\ 1 & -2 & 1 & & & & & \\ & & -2\lambda & \lambda & \lambda & & & \\ 1 & 1 & & -2 & & & & \\ & & \lambda & -2\lambda & \lambda & & & \\ & & \lambda & \lambda & -2\lambda & & & \\ & & & & & & & 0 \end{bmatrix}, \quad (3.9)$$

where we have omitted most of the zero entries for clarity. An alert observer will note that Q is reducible, and so by permuting the basis vectors we can rearrange

the matrix into block form:

$$Q' = \begin{bmatrix} -2 & 1 & 1 \\ 1 & -2 & 1 \\ 1 & 1 & -2 \\ & -2\lambda & \lambda & \lambda \\ & \lambda & -2\lambda & \lambda \\ & \lambda & -2\lambda & \lambda \\ & & & 0 \\ & & & 0 \end{bmatrix}. \quad (3.10)$$

According to our recipe, to learn about the solutions to Eq. 3.1 in this case, we need to know about the eigendecomposition of Q . The block structure of Q' means that there are 4 distinct parts of the state space, between which there are no transitions; this partitioning corresponds to the fact that particle number is conserved on the ring. Two of these sectors correspond to the full and empty states, and their dynamics are completely trivial, in the sense that the state space is 1d and there are no dynamics, as the particles/vacancies have nowhere to go! The remaining sectors correspond to the situation where there is one particle or one vacancy. The matrix is symmetric, meaning that its eigenvalues are real, and we find that both nontrivial blocks can be diagonalised to form a multiple of

$$\begin{bmatrix} 0 & & \\ & -3 & \\ & & -3 \end{bmatrix}, \quad (3.11)$$

with eigenvectors $[1, 1, 1]^T$, $[-1, 0, 1]^T$ and $[-1, 1, 0]^T$ respectively, the latter two forming a degenerate eigenspace.

In this particular example then, we find that there 4 steady states:

- All slots full,
- All slots empty,
- One particle present, with equal chance to be in any particular position,
- The same but with a vacancy instead of a particle.

Whilst the first 2 cases are trivial (one-dimensional probability spaces), in the other two cases we relax towards the steady state with rates 3 and 3λ respectively.

In this example, the TRM was actually symmetric, and so all the eigenvalues were real. It was also highly reducible, due to the strict constraints imposed by the particle conservation law. When we have boundary conditions which permit the creation and destruction of particles, that will change; we will consider that situation now.

3.2 Forming the TRM for Systems with Dirichlet Boundary Conditions

3.2.1 Dirichlet Boundary Conditions

In Chap. 2 we made an MFT for the SPM in 1-dimension, and when investigating steady states used Dirichlet boundary conditions when we needed them. In particular, we had a system of length L and sought solutions in which the system density was pinned at ρ_0 at $x = 0$ and ρ_L at $x = L$.

We would like to do a similar thing for the non-MFT SPM. An exact analogue of the situation does not exist, as occupation number is not defined *a priori* in a Markov process, but merely emerges as a result of the rate prescribed. The closest imitation to it we can get is by allowing particles to be created and destroyed in boundary regions at either end of a chain, and then try to set these rates so that the time-averaged occupation probability in the end sites are ρ_0 and ρ_L respectively. Note that this is a little more involved than simply loading and unloading particles as one does in ASEP, as

- loading and unloading really doesn't simulate the boundary condition we are looking for, which is attachment to a particle reservoir of constant particle density, and
- we actually need to consider a two-site boundary layer attached to each end, because the internal dynamics of the particles depend upon their immediate environment.

To simulate a boundary which is attached to a reservoir with occupation ρ , we allow particles to appear in empty boundary sites with rate $B_0\sqrt{\frac{\rho}{1-\rho}}$ and to disappear from full ones with rate $B_0\sqrt{\frac{1-\rho}{\rho}}$, for some positive B_0 . If we switch off

all other dynamics and consider this in isolation, we would have a collection of decoupled two-state systems. Writing a TRM for this with the first basis vector being the empty state and the second the full one, we get

$$Q = \begin{bmatrix} -B_0\sqrt{\frac{\rho}{1-\rho}} & B_0\sqrt{\frac{1-\rho}{\rho}} \\ B_0\sqrt{\frac{\rho}{1-\rho}} & -B_0\sqrt{\frac{1-\rho}{\rho}} \end{bmatrix}. \quad (3.12)$$

There is of course a zero eigenvalue, which corresponds to a stationary distribution $[1 - \rho, \rho]^T$, precisely as desired.

In order to simulate attaching a reservoir with particle density ρ , we apply the creation and annihilation rates above to the outermost two sites in our lattice. The outermost site only performs these operations. The inner boundary site undergoes these creation and annihilation processes, as well as the normal dynamics of the SPM; thus, particles will move in and out of it as normal, and when the outermost site's occupation is required to determine the transition rate for a particle moving inward, that information is available. One could possibly eliminate the need for the outermost boundary site by averaging over occupations, however we have chosen not to do this for consistency with our Monte-Carlo calculations in Chap. 4.

Observant readers will notice that by using these creation and annihilation rates we are left with a free variable, B_0 . This controls the ratio of the creation and annihilation rates to the internal dynamical rates 1 and λ . In general when we're trying to simulate an adjacent reservoir we want the boundary motion to be “fast” compared to the internal dynamics, and so B_0 may be regarded as a regularisation parameter; any choice of B_0 should be good so long as the creation and annihilation rates sufficiently dominate both 1 and λ . In practise, in our larger-scale calculations we used

$$B_0 = b(1 + \lambda), \quad (3.13)$$

with b set to, 100 or 1000.

3.2.2 Another Small Worked Example: Open System

Using full boundary conditions, the smallest system we can consider consists of adjacent boundary layers, in which particles pass directly from the inner layer of

one boundary to the inner layer of the other boundary. Unfortunately the state space of such a system is $2^4 = 16$ -dimensional, and so it would be cumbersome to consider such a system as an example here.

However, in the special case in which one boundary has density $\rho_0 = 1$ and the other has $\rho_L = 0$, the outer boundary layers become nondynamical, and the system simplifies considerably; particles can only enter at the full end with rate λ , and can only leave at the empty end. Therefore, we can consider a system consisting of single boundary layers attached to a single internal slot without needing to consider a state space larger than $2^3 = 8$. Using the same state-labelling convention as in Sec. 3.1.1, the TRM for this new open system is

$$Q = \begin{bmatrix} -\lambda & 1 & & & & & \\ & -\lambda - 2 & 1 & & & & \\ & & 1 & -\lambda - 2 & 1 & \lambda & \\ & & & & -2\lambda - 1 & \lambda & \\ \lambda & & & 1 & & -\lambda & 1 \\ & \lambda & & & \lambda & & -\lambda - 2 & \lambda \\ & & \lambda & & & & 1 & -\lambda & \lambda \\ & & & \lambda & & & & & -\lambda \end{bmatrix}. \quad (3.14)$$

This time the matrix is not symmetric, and there does not exist a basis permutation which puts the TRM into block form. If we calculate the characteristic polynomial $p(q) = |Q - qI|$, we find that

$$\begin{aligned} p(q) = q & [q^7 + (9\lambda + 7)q^6 + (34\lambda^2 + 53\lambda + 17)q^5 + (71\lambda^3 + 165\lambda^2 + 103\lambda + 17)q^4 + \\ & + (89\lambda^4 + 274\lambda^3 + 247\lambda^2 + 76\lambda + 6)q^3 + (67\lambda^5 + 256\lambda^4 + 294\lambda^3 + 126\lambda^2 + 17\lambda)q^2 + \\ & + (28\lambda^6 + 127\lambda^5 + 171\lambda^4 + 91\lambda^3 + 15\lambda^2)]q + (5\lambda^7 + 26\lambda^6 + 37\lambda^5 + 24\lambda^4 + 4\lambda^3). \end{aligned} \quad (3.15)$$

Of course, the roots of $p(q)$ are the eigenvalues of Q . Clearly one of the eigenvalues is $q = 0$, and the others could be found by finding the root of the 7th order polynomial, which we will not be doing analytically because it is extremely tedious and besides the point of this example. The 0-eigenvector, and therefore the steady-state probability distribution over the possible configurations, can be

computed with a little care, and turns out to be

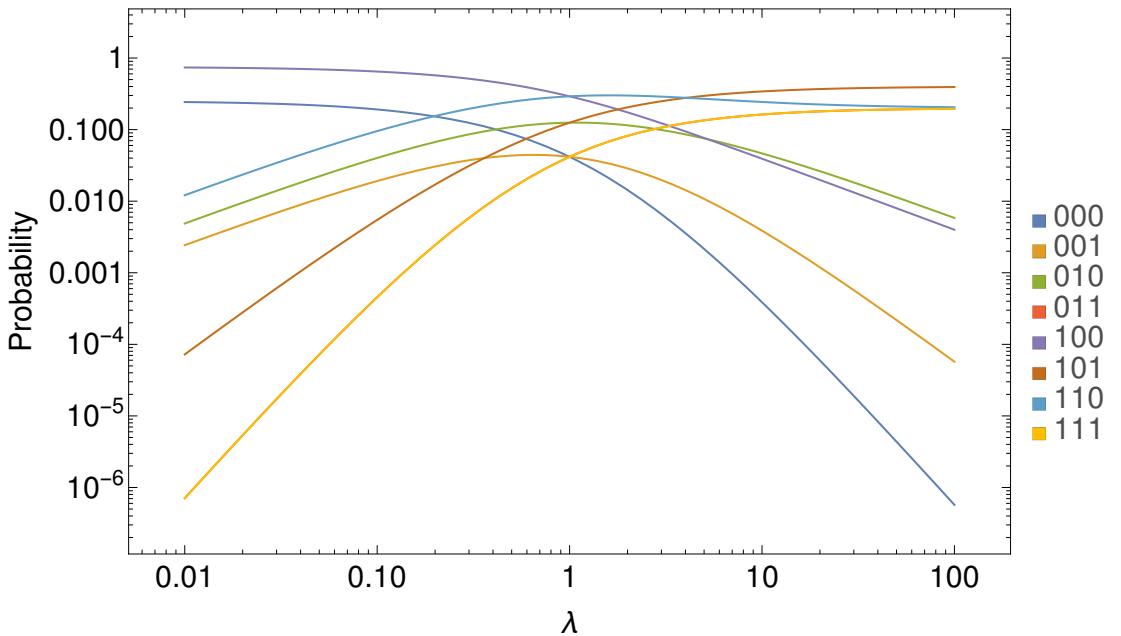
$$q_0 = D^{-1} \begin{bmatrix} 3\lambda + 1, \lambda(3\lambda + 1) \\ \lambda(\lambda + 2)(3\lambda + 1) \\ \lambda^3(\lambda + 3) \\ (\lambda + 3)(2\lambda(\lambda + 2) + 1) \\ \lambda^2(\lambda + 3)(2\lambda + 1) \\ \lambda(\lambda(\lambda(\lambda + 8) + 14) + 5) \\ \lambda^3(\lambda + 3) \end{bmatrix}, \quad (3.16)$$

where

$$D = \lambda(\lambda(\lambda(5\lambda + 26) + 37) + 24) + 4. \quad (3.17)$$

The variation of the occupation probabilities with λ is displayed in Fig. 3.1. Notice that at extremely small λ the system either empties or a single particle

Figure 3.1 *The variation of the steady-state configuration probability distribution with λ for our open system. The coloured curves correspond to specific configurations, as indicated in the legend.*



gets stuck to the full boundary, essentially preventing further flow by blocking it. Meanwhile, at large λ there are a few fairly popular states, with the most dominant being one with alternating particles and vacancies, as one might expect. The steady-state current from left to right in this system can be calculated analytically once we have the steady state probability distribution, and turns

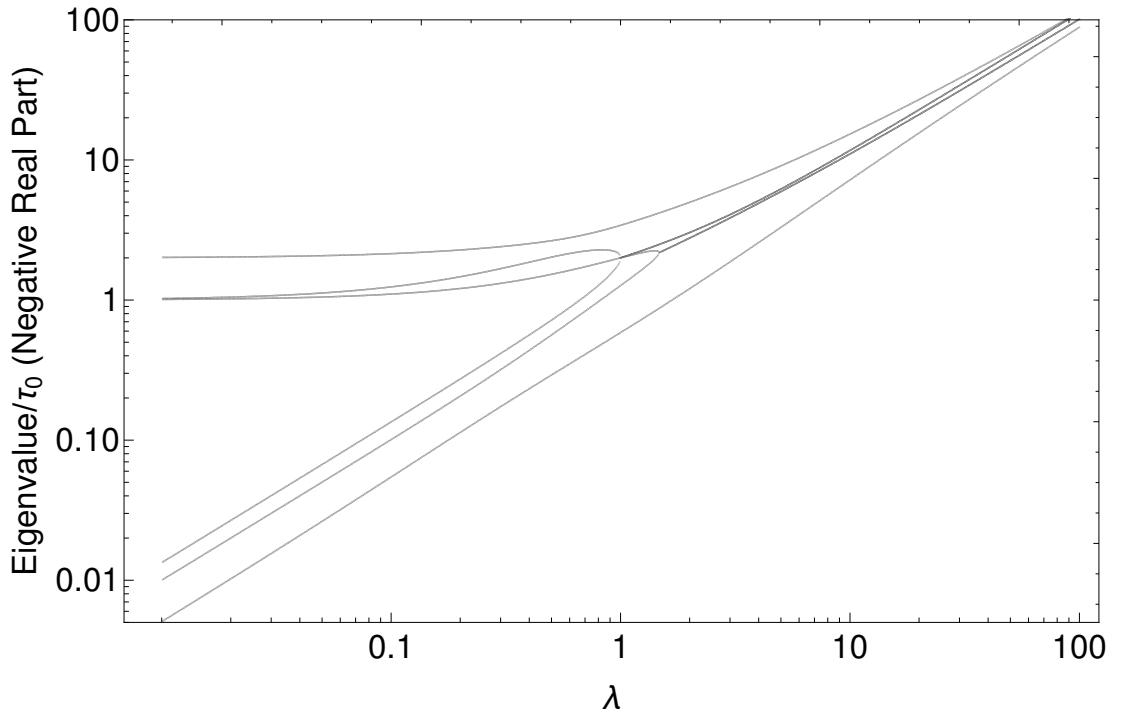
out to be

$$J = \lambda \left[\frac{1}{4} + \frac{(1-\lambda)(3+\lambda)\lambda^2}{4 + \lambda(24 + \lambda[37 + \lambda(26+5)])} \right]. \quad (3.18)$$

The second term in the bracket is generally dominated by the first, so we can say to an excellent approximation that $J \sim \frac{1}{4}\lambda$. Thus, at least for this tiny example with very extreme boundary conditions, there does not appear to be any change in the scaling of the current with λ .

We can also compute the negative real part of the eigenspectrum of the TRM for this small system; this is displayed in Fig. 3.2. It is in many ways rather

Figure 3.2 *The variation of the negative real part of the eigenspectrum of the TRM for our small open system with λ .*



similar to the eigenspectra we find for larger systems, so we see Sec. 3.3.2 for our comments on those.

3.2.3 Formation of the TRM in Sparse Format

As we have seen, it is possible to find the eigendecomposition of the TRM analytically for extremely small systems with special boundary conditions. However, we'd really like to see what happens for somewhat larger systems, with

more diverse boundary conditions. Thus, let us try to develop a method for the numerical analysis of TRMs of arbitrary size.

The important thing to note about transition rate matrices for local lattice models such as the SPM is that whilst the TRM itself grows very aggressively with system size, the TRM is generally extremely sparse. The state space dimension grows as 2^L , and the TRM dimension therefore grows as $2^L \times 2^L$. However, a state containing N particles only has transitions to

- states which differ from the current state by one particle move, of which there are $\mathcal{O}(N)$, and
- states which differ from the current state by a single particle creation or annihilation, of which there are $\mathcal{O}(4)$.

Thus, as $N \leq L$ the number of nonzero entries in the matrix is $\mathcal{O}(2^L L)$, which is not a particularly tight bound. Therefore, the overall density of the TRM is $\mathcal{O}(2^{-L} L)$. Given that there already exist a great many efficient numerical routines for sparse linear algebra operations, there exists the possibility that we could use this to solve the SPM on a finite domain for small systems “exactly” (or at least, up to some nominated numerical tolerance).

To make use of this, we need to assemble the TRM in a suitable sparse format. I have written a Python code which does this. The script itself is stored `jdecide` how precisely, but the gist of the algorithm is to simply run through all possible states, document all the transitions they can perform, and store the resulting entries in a sparse matrix element by element. During construction, the matrix should be stored in coordinate list format, i.e. a list of elements of the form (row, column, value), as this is trivial to update as we only inspect each matrix entry once. We can then convert the matrix to a compressed format such as CSC (Compressed Sparse Column) or CSR (Compressed Sparse Row). We used CSC, but in hindsight CSR would probably have been a better choice as it tends to make matrix-vector multiplication a little more efficient. Once we have the TRM in this format, it is ready for sparse linear algebra operations. Note that whilst these operations generally happen “in place”, this part of the process is the memory-intensive bit, as of course the memory usage scales with the number of nonzero matrix elements, which is $\mathcal{O}(2^L L)$. In terms of actual numbers, we found that 4Gb of memory was more than adequate to solve a system of 16 sites in total. As the memory required to represent the TRM is the main limiting factor

in this kind of calculation, if we were to attempt a larger calculation we would probably switch to using a machine with a very large working memory, such as DiRAC.

3.3 The Eigenspectrum of the TRM

3.3.1 The Computation of the TRM Eigenspectrum

Once we have the TRM Q in CSC or CSR format, we can then use sparse linear algebra. In our code we called the Python routine `scipy.sparse.linalg.eigs` upon it, which itself is a wrapper for C codes which find eigenpairs according to desired criteria; precisely which algorithm is used is determined during runtime, and it may try different methods if it doesn't initially succeed.

In our computations, we typically performed two types of calculation. In the first we merely sought to find the steady state, so which we requested only the eigenvector x_0 associated to the eigenvalue q_0 with smallest absolute value, which should always be numerically zero. We requested that the eigenpair be found to a relative accuracy ϵ accuracy of 1 part in 10^{12} , which amounts to saying that

$$\frac{\|Qx_0 - q_0x_0\|}{\|x_0\|} \leq 10^{-12} = \epsilon \quad (3.19)$$

where $\|\cdot\|$ is some reasonable subordinate matrix norm (in our case, the 1 norm). Because we requested the eigenvalue closest to 0, `eigs` used the shift-invert method `|find article|`, leading to greater accuracy in the computation of eigenvalues near to 0 which is exactly what we wanted. For the other type of computation, we instead requested the k eigenpairs with largest real part, which most likely provoked the code to use a Implicitly Restarted Arnoldi Method (IRAM). This is not as accurate for computing the steady state, but yields vastly more accurate results when computing the other eigenpairs compared to the first method.

3.3.2 The Structure of the TRM Eigenspectrum

Using the code listed at `|place|`, we can compute the eigenspectrum of an SPM system with boundary densities connected at both ends. Such a computation

requires the following parameters to be specified:

- $\rho_0 \in (0, 1)$, the density of the reservoir connected to the left end of the domain.
- $\rho_L \in (0, 1)$, the density of the reservoir connected to the right end of the domain.
- $L \in \mathbb{N}$, the system size. The way we have defined things in the code, we do not count the two sets of two particles representing the boundary; thus, $L = 1$ actually refers to a system which contains 5 lattice sites, of which 4 are busy doing boundary duties.
- $b > 0$, the variable which controls the separation of timescales between the flickering motion on the boundaries and the internal motion within the bulk of the system. This should be set to something large, and compared to other values of it to ensure that it is working as a regularisation parameter (i.e. large changes to b have minimal impact on the relevant internal dynamics of the system, even if they have a great impact on the boundary dynamics).
- $\lambda > 0$, the internal anomalous movement rate in the SPM.
- ϵ , the relative accuracy the calculation aims for in the sense of Eq. 3.19.
- $1 \leq k < 2^{L+4}$, the number of eigenvalues to compute.

For consistency we kept L , b , ϵ and k constant during runs of calculations. This leaves ρ_0 , ρ_L and λ to be varied. In terms of what to vary and how to display the data, we decided to allow the spectrum to depend upon only one variable, and picked λ to be that variable. We generally chose $\rho_0 = 0.6$ and $\rho_L = 0.4$ in order to study a system in which, at least for $\lambda = 1$, the density is middling and a current flows. We then computed the resulting eigenspectrum in a couple different ways. First, we performed a relatively small calculation, in terms of the number of eigenvalues demanded and the number of λ used. We altered the values of L and b between runs, so that we can see what impact they have on the eigenspectrum. We then performed a very large calculation, in order to get a good look at the eigenstructure as a whole. In Chap. 2 our MFT suggested that a transition might occur around $\lambda = \frac{1}{4}$, so we should look out for odd behaviour in that regime.

To actually perform these TRM calculations, we used Edinburgh Compute and Data Facility’s Linux Compute Cluster, known as Eddie3. We will discuss this machine and its capabilities further in Chapter 4. Although we did initially consider the possibility of exploiting the natural parallelisability of the numerical linear algebra routines, we concluded that it was not probably not worthwhile in terms of computational gains, whilst requiring a lot of extra development time to implement. Instead, we simply submitted large numbers of separate serial jobs with different parameter values (e.g. for λ , ρ_0 , ρ_L) to Eddie3, and then allowed the machine to process them in its own time.

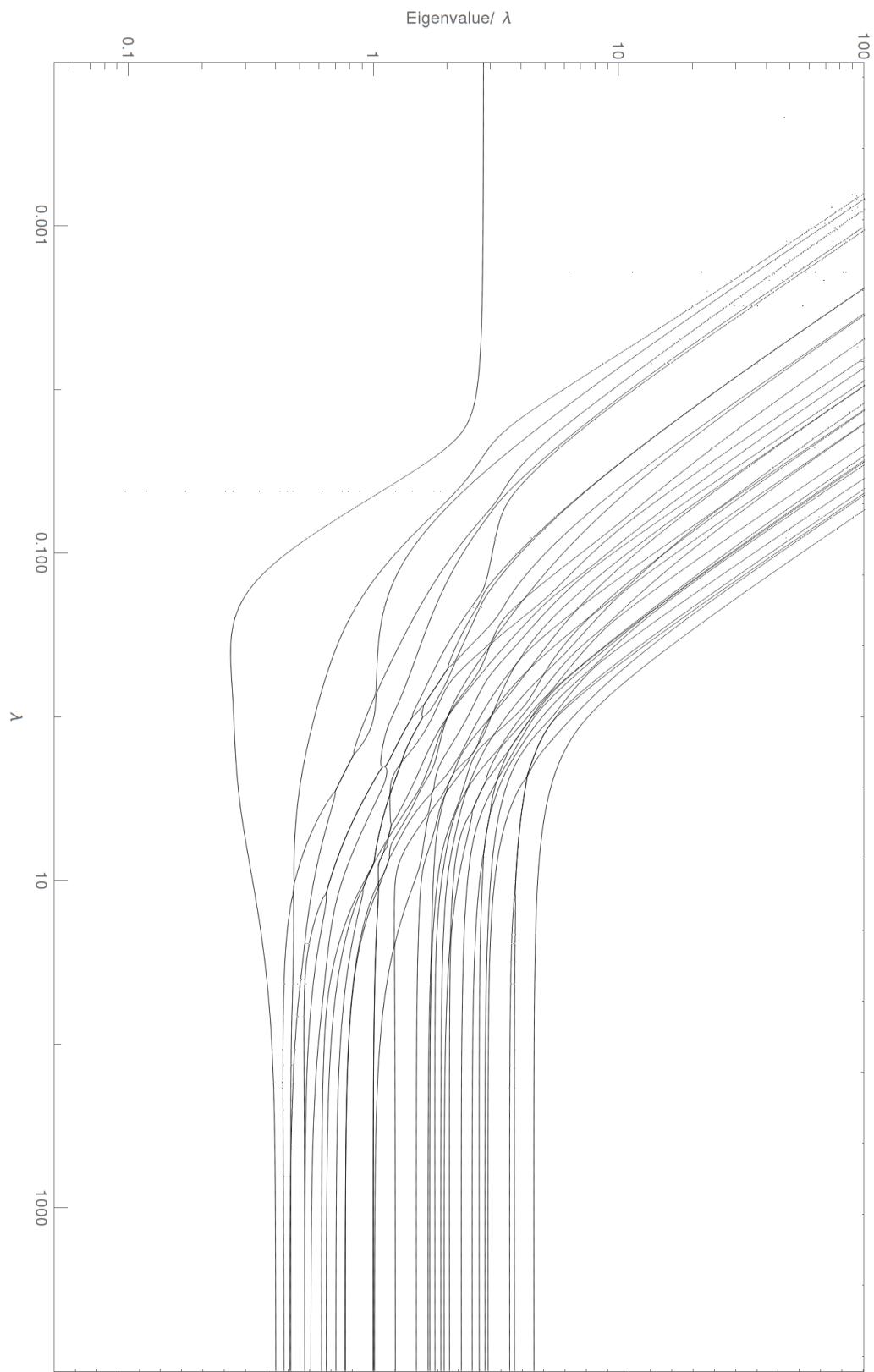
Small Calculations

First, we performed a calculation with $L = 5$, $b = 1000$ in which we computed the negative real part of the 32 eigenvalues with real part closest to zero for a wide range of λ with fixed boundary conditions. The resulting eigenspectrum is displayed in Fig. 3.3. Of course, for such a system we expect that all eigenvalues have negative real part, as we discussed in 3.1, so we have chosen to plot the negative real part divided by λ as a function of λ ; the zero eigenvalue, which we expect to exist and is generally found to exist in numerical terms, has been ignored, as its magnitude is simply an artefact of the numerical calculation and is therefore meaningless except possibly as a check on the numerics. The decision to divide by λ was taken because the eigenvalue closest to 0 (which dominates approach to equilibrium) mostly scales as $\mathcal{O}(\lambda)$, so plotting without division uses graph space poorly.

There are several things to note about this spectrum:

- It would appear that, for all λ , there is a single eigenvalue which is closest to 0 and undergoes no definitive crossing as λ varies. At both extremes of λ , it clearly scales as $\mathcal{O}(\lambda)$, but is unusually low in an intermediate regime of $\lambda \in (0.03, 10)$, reaching a minimum at $\lambda \sim 0.3$, which could be interpreted as an “avoided crossing” with the 0-eigenvalue. This is important, as the eigenvalue with smallest negative real part is the one that controls the approach to equilibrium; thus, we can see that around 0.3 the system should take unusually long to relax from an arbitrary prepared state to equilibrium.
- All eigenvalues tend to scale as $\mathcal{O}(\lambda)$ as $\lambda \rightarrow \infty$.

Figure 3.3 The TRM eigenspectrum, computed for a system with $\rho_0 = 0.6$, $\rho_L = 0.4$, $L = 5$, $b = 1000$.



- Eigenvalues other than the bottom one tend to scale as $\mathcal{O}(1)$ at small λ . Thus, whilst the overall approach to equilibrium occurs with rate $\mathcal{O}(\lambda)$, there are still plenty of (probably more localised) processes occurring over much quicker timescales, whilst the whole system is sluggish to equilibrate.
- The eigenvalues tend to retain their order at the extremes of λ , but they frequently cross, merge and separate again in the intermediate regime. This suggests that something complicated is occurring; an analytic solution to this dynamical model would need to describe these crossings in detail, as they are important as eigenvalue crossing implies the appearance of degenerate eigenspaces in the decay modes which the associated eigenvectors refer to. This makes it look rather unfeasible that such an analytic solution could be constructed in the first place.

Of course, this is only the 32 lowest eigenvalues; there could be different behaviour in the higher ones. Before we perform a larger calculation however, let us turn our attention to the dependence of the eigenspectrum upon b and L , which we have studied by repeating our calculation with different values for those parameters, as shown in Fig 3.4. Again, there's a lot going on in this image, so let's break it down.

- Firstly, let us note that keeping L constant and switching b between 100 and 1000 seems to have very little impact on the eigenspectrum, as we can see by the fact that the points corresponding to the same system sizes are more or less lying on top of each other unless we delve into the low- λ limit of the $L = 5$ system. In the same limit, we see that the discrepancy is vastly reduced in the $L = 10$ system, suggesting that it is less bad for large systems, and is in some sense a small-size effect. Regardless, it would appear to be the case that adjusting b doesn't affect these low-lying eigenvalues very much, so it seems to be playing its role as a regularisation parameter as intended. Thus, we're simply going to use $b = 1000$ from now on, unless explicitly specified otherwise.
- The bottom eigenvalue, which controls relaxation to equilibrium, is generally several orders of magnitude lower for the $L = 10$ system than for the $L = 5$ one; this effect is particularly extreme in the limit of small λ . This makes sense; the larger system should take much longer to equilibrate than the smaller one, as so many more fluctuations need to occur to make it

Figure 3.4 The lower TRM eigenspectrum, computed for a system with $\rho_0 = 0.6$, $\rho_L = 0.4$, with all combinations of $L = \{5, 10\}$, $b = \{100, 1000\}$. Missing computations, visible via the vertical gaps in the data, are due to computational issues, rather than being numerically meaningful. Note that in many places green/black and blue/red lines overlap, because their results are so similar; the image is deliberately prepared in a high resolution so that these intricacies may be observed, at least in the digital copy.

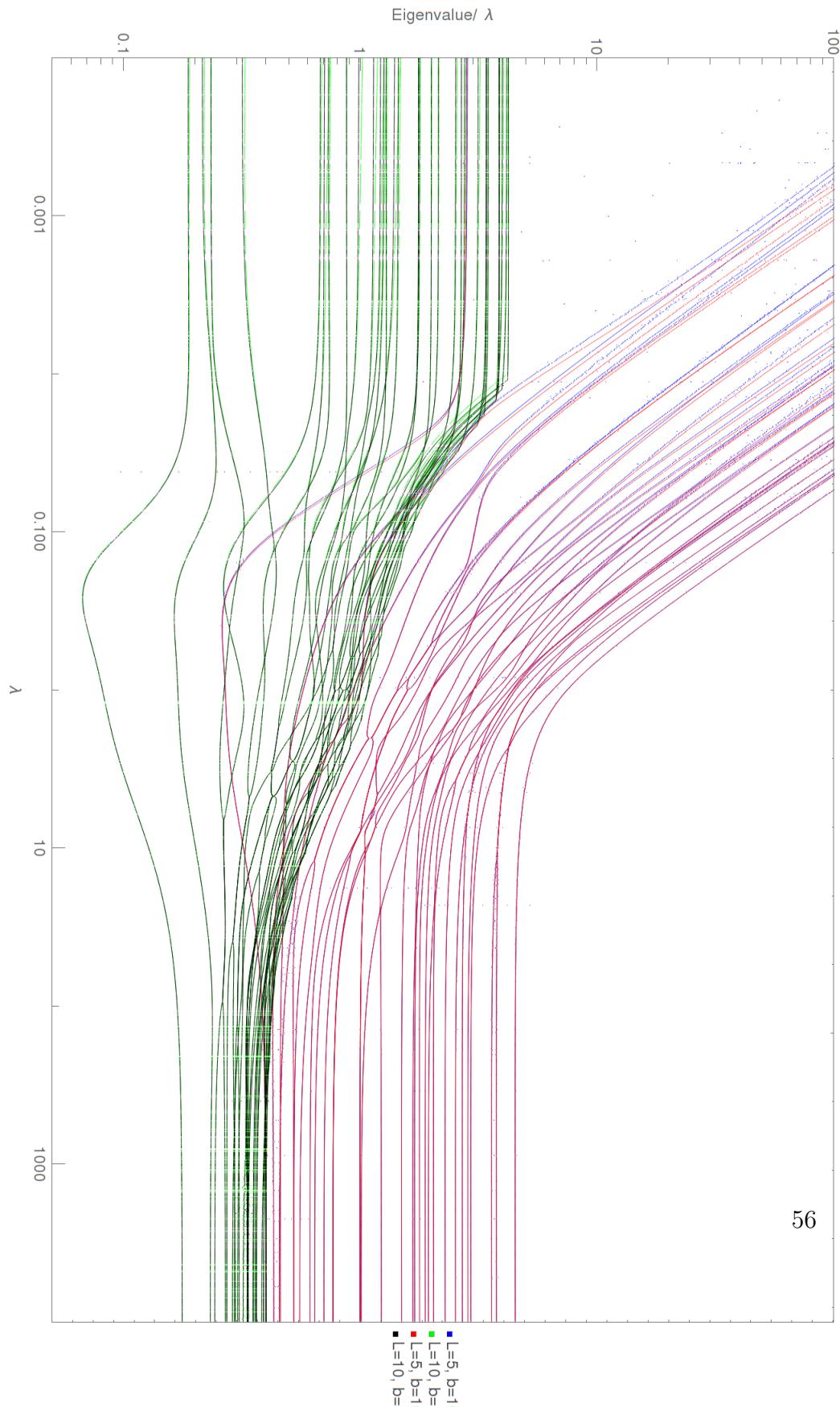


Table 3.1 A table of the number of slow modes occurring at low λ for given L .

| L | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 |
|-----------|---|---|---|----|----|----|----|-----|-----|
| Bandwidth | 1 | 3 | 6 | 11 | 20 | 36 | 64 | 113 | 199 |

change its global state. This is something we should remember for later, as it suggests that equilibration time might scale pretty aggressively with system size, which is bad news for our attempts to simulate the system using Monte-Carlo methods (Chap. 4).

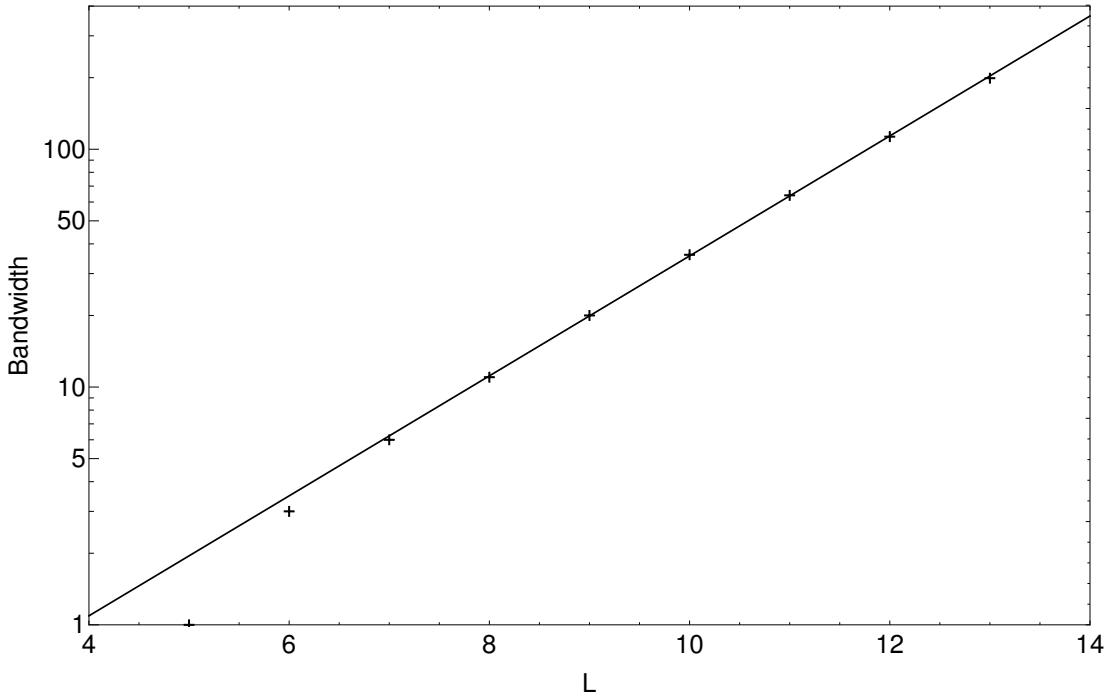
- The big, obvious difference between the spectra of different system size is the behaviour at small λ . For $L = 5$ there is only 1 eigenvalue which scales as $\mathcal{O}(\lambda)$ as $\lambda \rightarrow 0$, whereas for $L = 10$, it would appear that they all do!

The last point begs the following question: how does the number of eigenvalues corresponding to “slow modes” at small- λ scale with the system size? This is important, because it determines how many of the available decay modes would tend to persist for any meaningful amount of time during the relaxation towards equilibrium. We will attempt to address this in the next subsection.

The Scaling of the Number of Slow Modes at Low λ

To find out how the number of nonzero eigenvalues in the $\mathcal{O}(\lambda)$ -scaling band (as observed in Fig. 3.4) scales with system size, we can simply fix λ to be sufficiently low (say, $\lambda = 0.0001$) and repeat computations of the low-lying eigenspectrum for different L . Here, due to the computational limitations we are up against, we only compute up to $L = 13$. The number of slow modes as a function of L is displayed in Tab. 3.1. If we plot the number of slow modes as a function of system size on a plot with a logarithmic y -axis, as we have done in Fig. 3.5, it is rather obvious that the bandwidth scales exponentially with system size. However, the number of slow modes seems to scale as $\mathcal{O}(2^{0.84L})$, whereas the number of possible states, and therefore the total number of eigenvalues of the TRM, scales as 2^L . Thus, one can see that although the number of slow modes grows very aggressively with the system size, it would appear to eventually become outpaced by the growth in the total number of eigenvalues; in this way, we can say that the slow band carries increasingly less of the total eigenvalue density as system size becomes large.

Figure 3.5 A plot of the scaling of the number of slow modes in the eigenspectrum of the TRM of an SPM system of size L . The trendline displayed has equation Bandwidth = $2^{a(L+b)}$, with $a = 0.84$, $b = -3.9$.



The Upper Part of the Spectrum

By varying L and b and analysing the lower (more physically relevant) part of the TRM eigenspectrum, we concluded that b has very little impact and essentially acts as a regularisation parameter, whilst L , the system size, has a large impact, particularly on the approach to equilibrium. However, both b and L contribute to the actual numbers contained within the TRM, so it stands to reason that they must impact the eigenspectrum in some way. Therefore, we have done an additional check, by varying b and L and performing a calculation in which the 64 eigenvalues with largest negative real part were calculated. The negative real parts of these eigenvalues are displayed in Fig. 3.6. We see that now the tables are indeed turned. Altering L does have a small effect on the upper band, as we can see by how the red/black and blue/green points lie on top of each other. On this graph, we see that for the $L = 5$ datasets there are apparently two bands, whilst for $L = 10$ there is only one; however, recall that we have produced this by plotting the negative real parts of the 64 eigenvalues with largest negative real part; thus, as the $L = 10$ systems have $2^5 = 32$ times as many eigenvalues in total,

it makes sense that we may not have finished the very top band in the $L = 10$ case yet whilst we have moved on to the second band in the $L = 5$ case; thus, we conclude that system size has little effect on the top of the eigenspectrum.

The value of b , however, has a huge impact, as it determines the overall magnitude of these larger eigenvalues. The eigenmodes associated with them almost certainly relate to fluctuations at the boundary, and then the ensuing joint fluctuations travelling a little into the bulk. As these are boundary effects, it again makes sense that L has little impact. The value of b directly controls the speed of the fluctuations at the boundary, thus the large shift in magnitude observed should be expected.

Large Calculation

Now that we are more or less satisfied that the value of b doesn't matter so long as it's quite large, let's perform a similar calculation (sweeping through λ with constant boundary conditions), but this time compute a lot of eigenvalues in order to get a general overview of the TRM. Plotted in Fig. 3.7 are the negative real components of the 1024 eigenvalues with smallest negative real part, for the TRM of an SPM system of length $L = 8$ and regularisation parameter $b = 1000$. We say that this is the “full” eigenspectrum not because we have computed all of the eigenvalues (we have not, as there are $2^{12} = 4096$ in total), but because the eigenvalues with very large negative real part seem to converge upon each other, forming the hyperbola-shaped entity towards the top of the graph. We believe that this is because the relevant dynamical modes correspond to extremely rapid changes in system state which are limited to the region immediately around the boundaries; as these should have no impact on large-scale properties such as the current, they are of very limited interest to us.

This large computation continues the themes we have seen in the small ones:

- For large λ there is a thick band of eigenvalues spread over a couple of orders of magnitude which scale as $\mathcal{O}(\lambda)$.
- For intermediate λ the bottom eigenvalue becomes unusually small, whilst the higher ones in the thick band are constantly crossing over each other.
- As we go to small λ a thin band of $\mathcal{O}(\lambda)$ -scaling eigenvalues split off from the main sequence, which scales as $\mathcal{O}(1)$.

Figure 3.6 The upper TRM eigenspectrum, computed for a system with $\rho_0 = 0.6$, $\rho_L = 0.4$, with all combinations of $L = \{5, 10\}$, $b = \{100, 1000\}$.

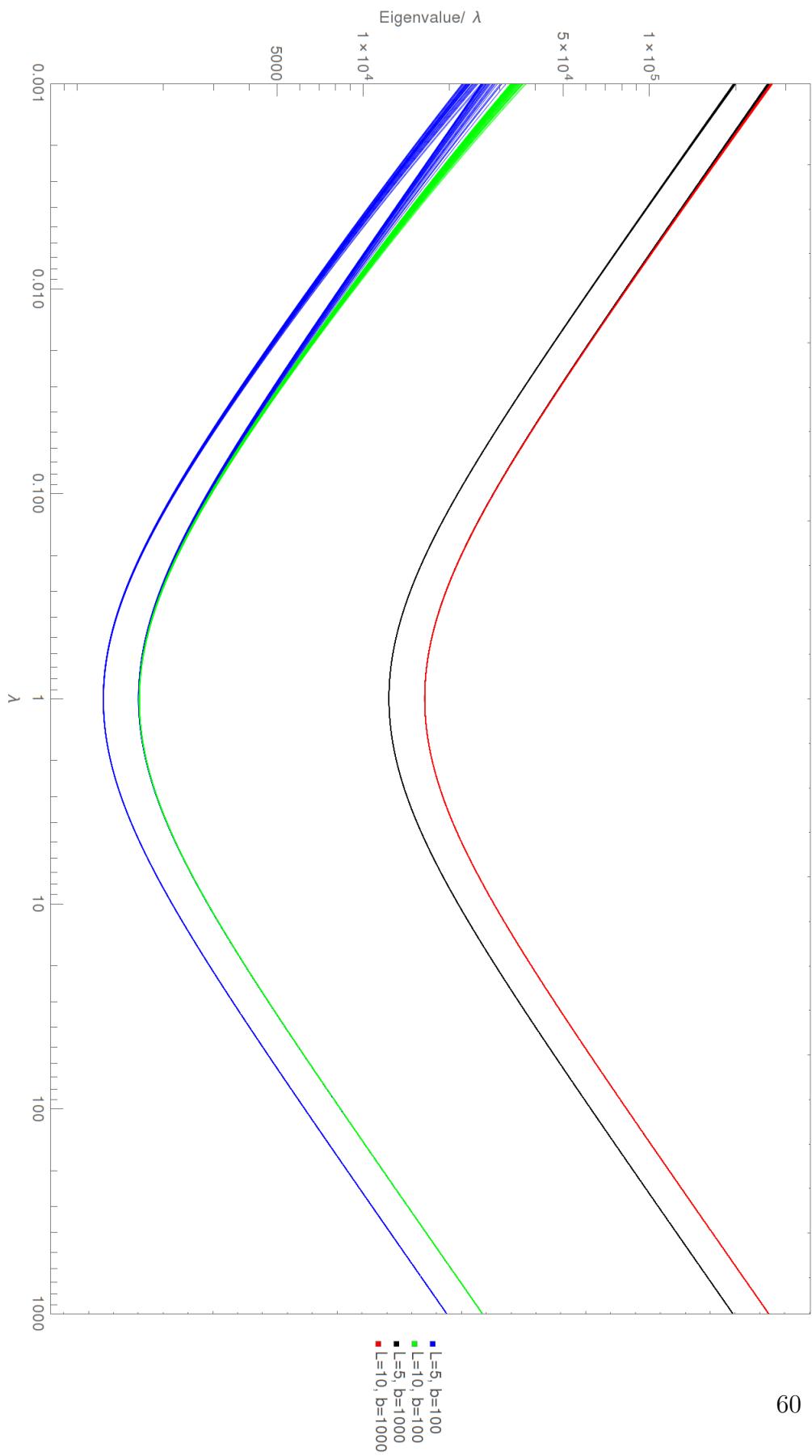
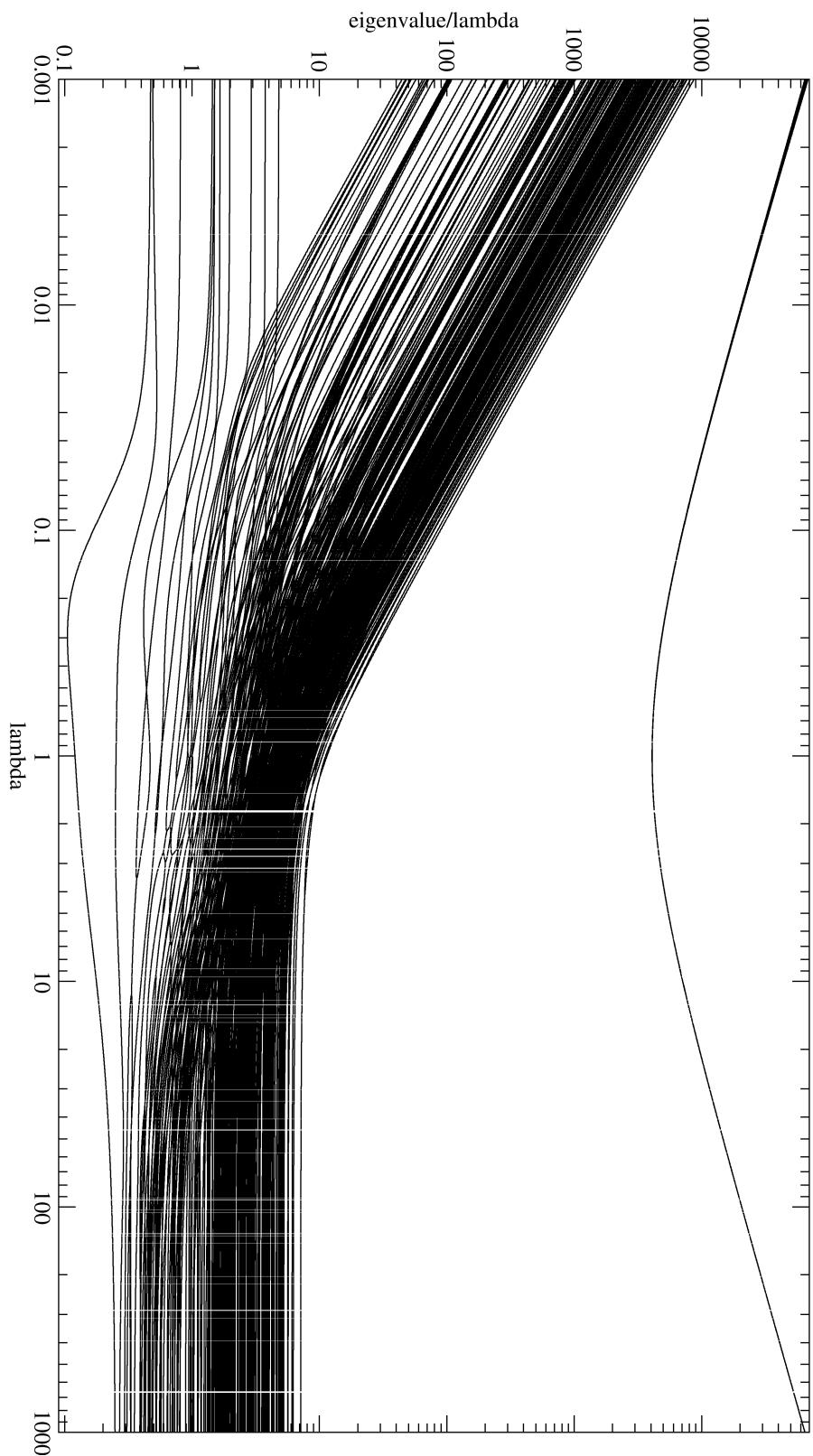


Figure 3.7 The “full” TRM eigenspectrum, computed for a system with $\rho_0 = 0.6$, $\rho_L = 0.4$, with $L = 8$, $b = 1000$. Missing computations, visible via the vertical gaps in the data, are due to computational issues, rather than being numerically meaningful.



- For all λ there is the hyperbola-shaped band of eigenvalues with highly negative real part, which are many orders of magnitude different to the main sequence; thus, their dynamics operate over an extremely short timescale, and they are almost certainly attributable to the flickering motion at the boundaries.

3.3.3 Current and Density in the Steady State

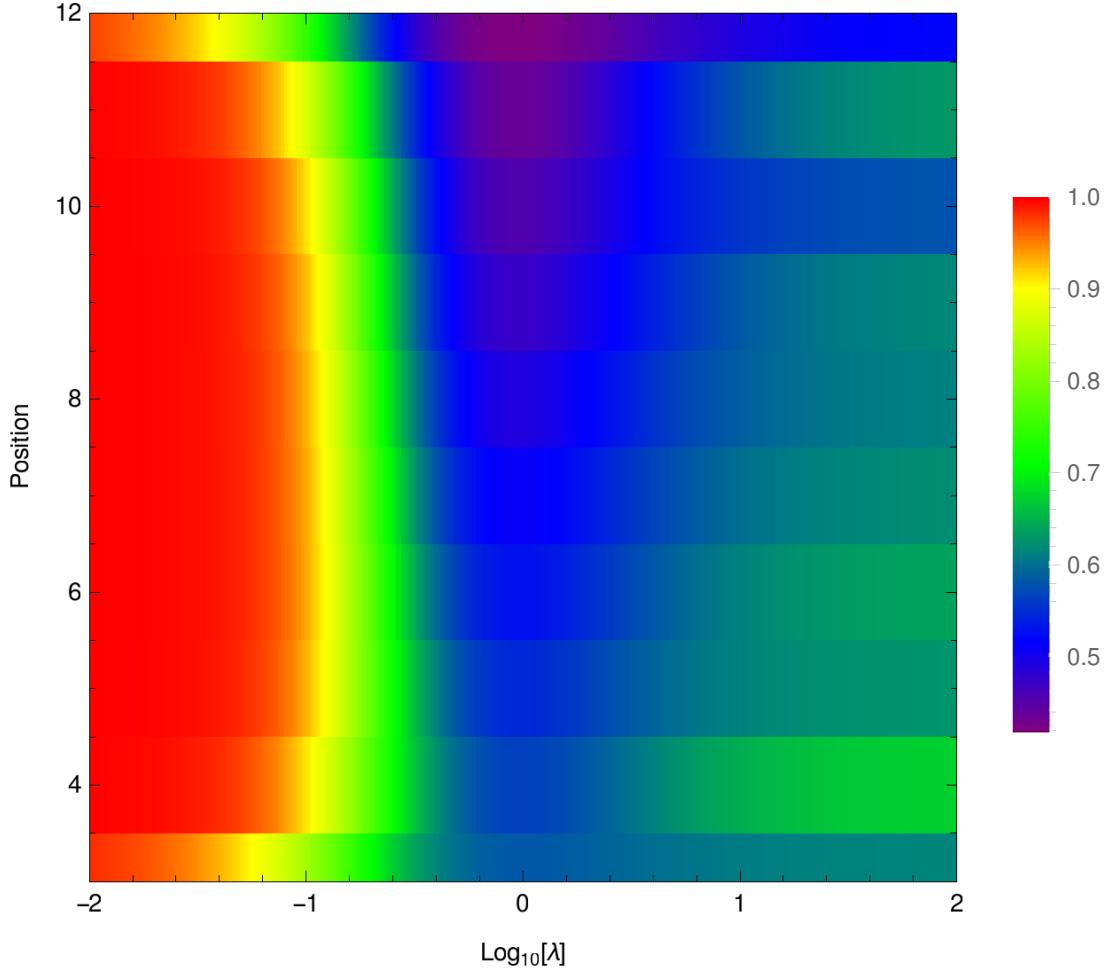
The eigenvectors of the TRM correspond to decay modes, and in general contain both positive and negative components; thus, they do not in general correspond to a particular distribution, only to a time-dependent part of a sum of vectors on their way to equilibrium. The exception to this is the 0-eigenvalue eigenvector, which **does** correspond to a physical state, if properly normalised.

Given a vector corresponding to a normalised probability distribution, we can extract the mean occupation of its i^{th} site by taking the inner product of the distribution vector with a vector which has component 1 on states in which the i^{th} sites is occupied and 0 otherwise. Thus, we can construct an operator $P : \Xi \rightarrow \mathbb{R}^{L+4}$ which can tell us the density profile of the system given its ground state, which we have found using the linear algebraic methods discussed in Sec 3.3.1. We can create a similar operator $J : \Xi \rightarrow \mathbb{R}^{L+3}$ which gives the current; however, if used on a steady state, it simply gives a constant result internal to the system, due to the existing constraint that particle number is conserved. We can of course extract the homogeneous current through the system by using this operator and then simply keeping one of the current components.

TRM-Computed Density Profile

By way of example, we computed the density profile for a system with $L = 10$, $b = 1000$, again sweeping through a wide range of scales of λ as in the previous section. We kept the boundaries constant with our usual $\rho_0 = 0.6$, $\rho_L = 0.4$ configuration. A density plot of the data is shown in Fig. 3.8. Looking at $\lambda = 1$, we see that there is a roughly constant gradient for the density profile between the boundaries, which makes sense as this is the simple exclusion limit, so it's essentially just normal diffusion. As we go to smaller λ , we see that the system in general is quite full. Presumably this is because material has been sucked into the system due to the attraction implied by the small λ ; this indicates that the

Figure 3.8 A coloured plot showing the variation of the density profile with λ in steady state for a system with boundary conditions ($\rho_0 = 0.6, \rho_L = 0.4$) of size $L = 10$. Note that the indices of the internal sites are 3-12 inclusive, as 1-2 and 13-14 are reserved for the boundaries, whose densities are already prescribed and therefore aren't plotted here.



boundaries used are a bit odd in this limit, as the system might not naturally form a homogeneous phase for such a λ and could instead wish to form a mixture of lower and higher density clumps, we will discuss this at greater length in Chap. 4.

Our theory about why this occurs is as follows. At large λ , we see that the density profile defaults to $\rho \sim \frac{2}{3}$ plus oscillations on the boundaries which decay as we go towards the interior. This is suspiciously close to the critical density of the MFT (Chap. 2) which permits a maximal flow; therefore, we suggest that the system self-organises at high λ to favour configurations that permit high current. The reason this density permits a high current is that in such a configuration, any particular particle should on average be in contact with an adjacent particle, with

a free space to move into, so the dynamics of the system should be dominated by λ rather than 1. If the density were higher, some particles don't have spaces to move into, so that will slow things down; if the density were lower, particles aren't receiving the speed benefit of being a neighbour. We believe that this self-organisation occurs in the bulk, with thin boundary layers occurring near to boundaries with densities fixed away from $\frac{2}{3}$. We believe that this thin layer is revealed by the oscillatory layer observed near to the boundaries; right next to the boundary, in alternating sites, particles will tend to linger in sites where they are alone, rapidly leaving when they are prompted by the arrival of a new particle in the space next to them. This effect then smooths out as one goes deeper into the bulk.

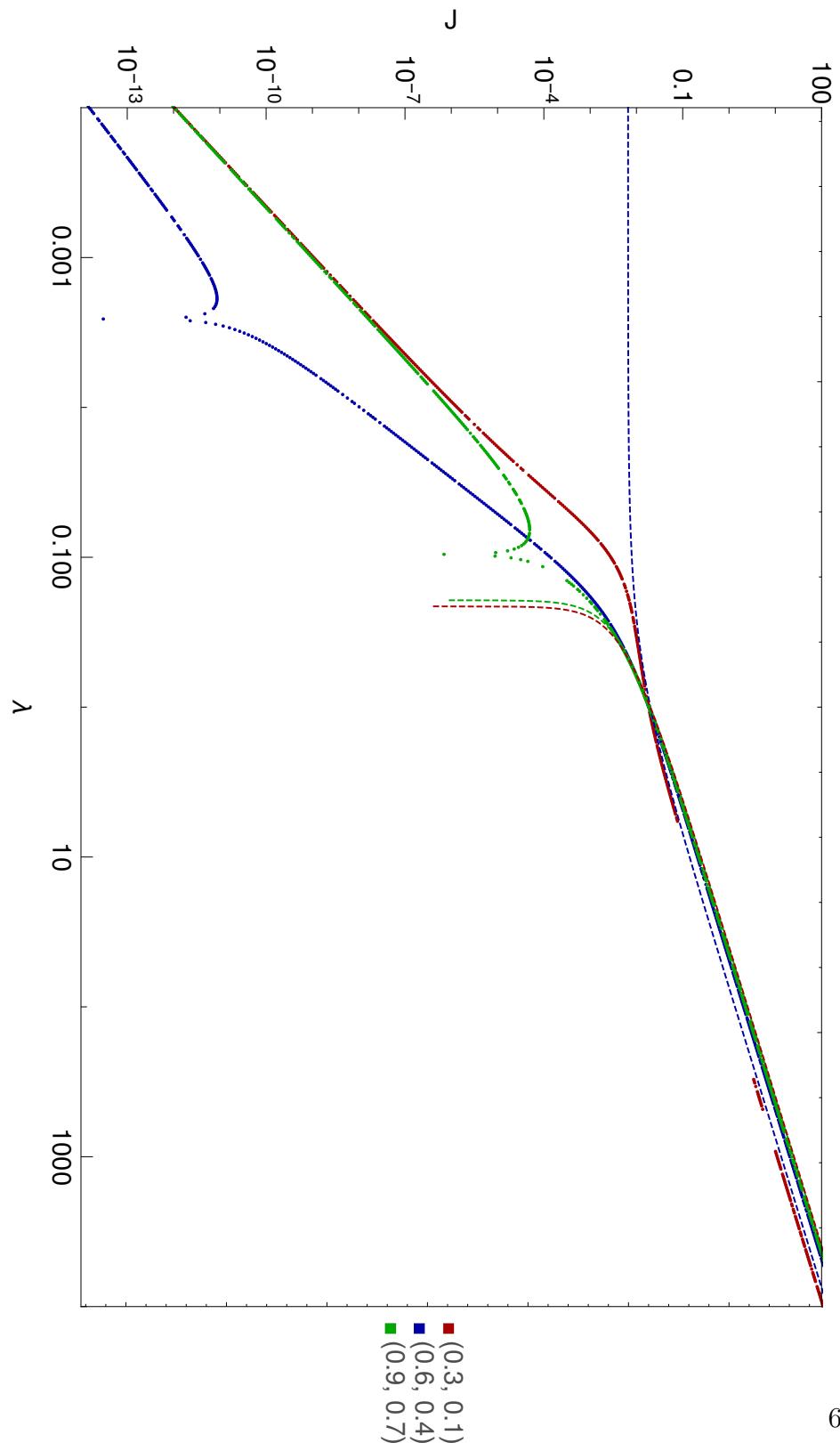
As for why a density which permitted fast flow would be favoured, consider what happens when a region of low flow comes into contact with one of high flow: the high-flow region will invade the other region with particles or vacancies, raising the overall flow rate (which previously would have been limited by the slowest-flowing region) and thus flushing out or bringing in new particles. In this way, the system as a whole will tend to favour the fast-flow densities, as they are more stable than the alternatives.

TRM-Computed Current

Whilst we computed the density profile, we also measured the steady state current. We also performed the same computation for different boundary conditions, and the results are displayed in Fig.3.9, along with the corresponding MFT prediction. Note that some of these calculations failed hence the gaps in the spectra. There's a lot going on here, so let's break it down.

- At large λ , we see that the scaling ($\mathcal{O}(\lambda)$) is the same for both the MFT and the TRM numerics. The actual values predicted by the MFT and the TRM calculations differ slightly, but not by a great deal until we have $\lambda < 1$. It is to be expected that there is a discrepancy, as the MFT is in the continuum-limit whereas the TRM system is only of size $L = 10$. Furthermore, there should be a discrepancy anyway as the TRM system is not mean-field; it just doesn't seem to be so relevant in this regime.
- At $\lambda = 1$, MFT and TRM calculations coincide, because the boundary conditions all differ by the same amount and $\lambda = 1$ is the trivial case of

Figure 3.9 A logarithmic plot showing the variation of current with λ in steady state for a system with boundary conditions (ρ_0, ρ_L) as indicated with system size $L = 10$. The current measurement is oriented so that positive current corresponds to flow from high density to low (the normal case). The relevant MFT prediction is shown via a dashed line in each case.



simple exclusion.

- As we explore very small values of λ , we find that the MFT and the TRM have very little resemblance. For boundary conditions $(0.3, 0.1)$ and $(0.9, 0.7)$ the MFT predicts that the flow should start running backwards or at the very least be pinned at zero, whilst for $(0.6, 0.4)$ it suggests that the current should tend to a constant as $\lambda \rightarrow 0$. Instead of the flow crashing to zero, the TRM results show a more gradual reduction of the current. Measurement by comparison with a trendline shows that the current varies as $\mathcal{O}(\lambda^3)$ as λ becomes small.
- There is an intermediate regime in which the dependence of the current upon λ transitions between $\mathcal{O}(\lambda^3)$ and $\mathcal{O}(\lambda)$. This is the situation for $\lambda \in (0.03, 0.5)$.

So, it seems that whilst we don't see a transition to zero flow as predicted by the MFT, we do see a big change in the way the flow depends upon λ as we pass between the large- λ to small- λ regimes.

TRM-Computed Diffusion Coefficient

It is also possible for us to compute an approximation to the effective diffusion coefficient. Recall that the diffusion coefficient D should satisfy

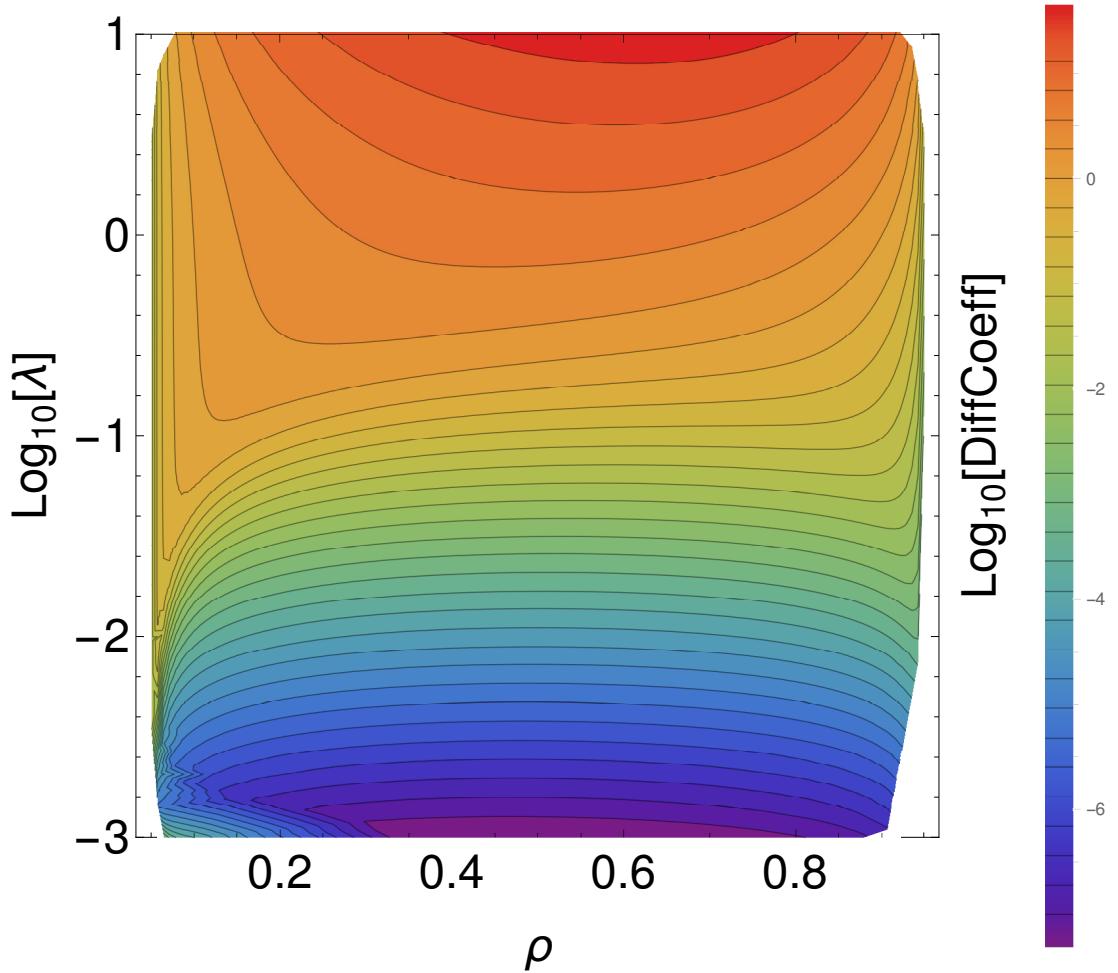
$$\mathbf{J} = -D\nabla\rho; \quad (3.20)$$

thus, we should be able to compute the diffusion coefficient using the TRM by computing the current J with boundary conditions $(\rho_0, \rho_L) = (\rho + \frac{1}{2}\delta\rho, \rho - \frac{1}{2}\delta\rho)$, and then evaluating the quantity

$$D(\rho, \lambda) = \frac{L}{\delta\rho} J. \quad (3.21)$$

We performed such a calculation, in which we picked $\delta\rho$ to be 0.001 and $L = 10$. We tabulated D for a range of λ and ρ , and a contour plot of the data is displayed in Fig. 3.10. In general, for given $\lambda < 1$ we see that the variation with ρ is usually pretty mild, with the more extreme diffusion coefficients tending to occur for intermediate ρ . The exception to this occurs primarily at small- λ , low density, where the diffusion coefficient is unusually high. This is presumably because the

Figure 3.10 A coloured plot showing the variation of the diffusion coefficient with λ and ρ in steady state for a system of size $L = 10$.



density of particles is so low that the small value of λ has little impact on the diffusion coefficient. There is also some odd behaviour, in particular for extremely low λ and low ρ . We probably shouldn't trust those the results. As $\lambda \rightarrow 0$, the space of vectors with eigenvalue zero switches from being 1-dimensional to being very high-dimensional, because any state containing adjacent particles becomes a steady state (and in fact an absorbing state). Thus, we should expect that at some point as we go to smaller and smaller λ our calculations should start behaving badly because the lower nonzero eigenvalues become numerically indistinguishable from the actual zeros, and that is probably what is happening here. For $\lambda > 1$, we see that maximal flow for a given λ occurs for intermediate values of ρ , consistent with the normal SEP result for $\lambda = 1$. We can see that as λ becomes large, the maximal flow density drifts towards $\rho = \frac{2}{3}$; this agrees with our notion, backed

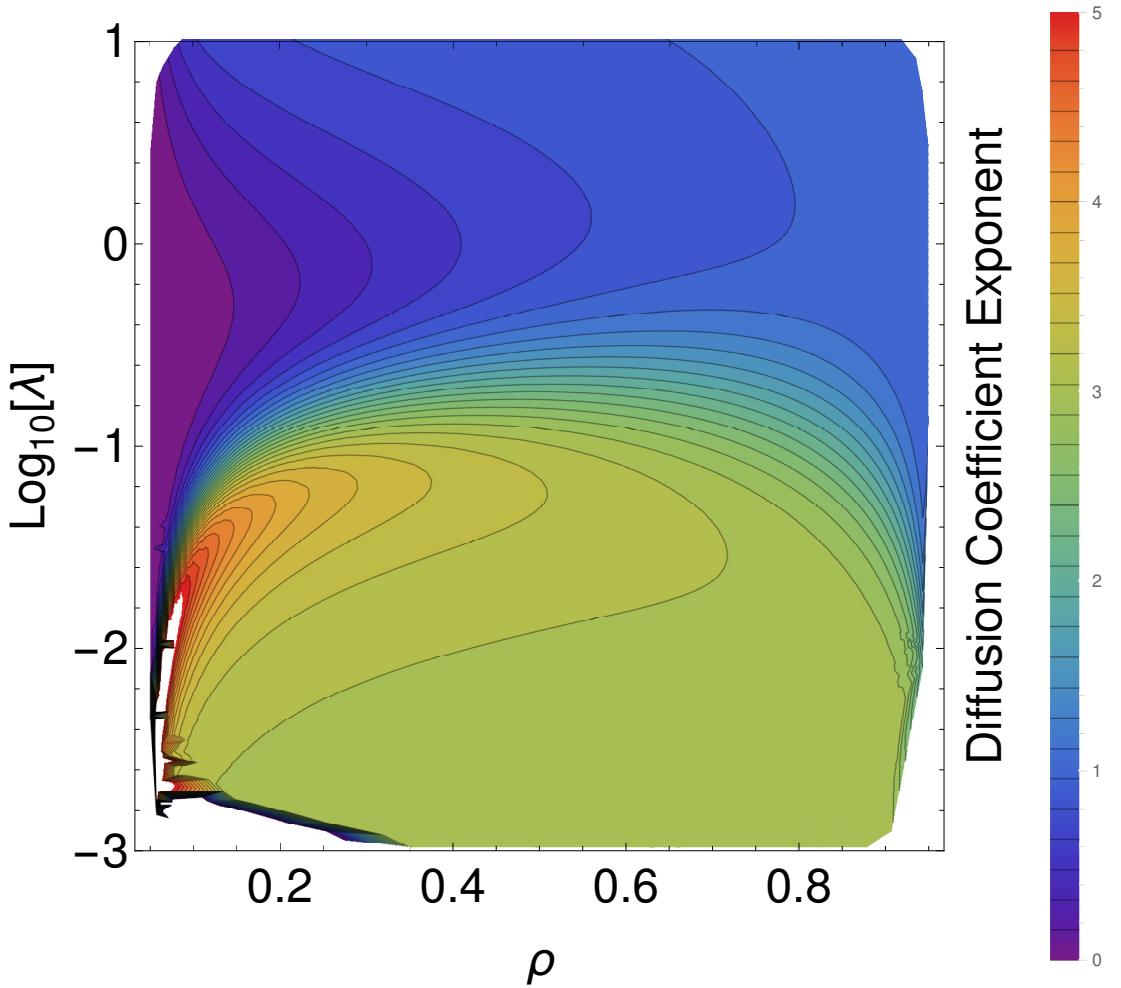
by the MFT, that maximal flow at large- λ should occur for $\rho = \frac{2}{3}$.

The trend in terms of λ is that the gradient of the diffusion coefficient as it varies with λ and ρ is generally low for large λ and high for small λ . In light of our results in Fig. 3.9, this makes sense as the power law seems to change as we switch between low and high λ . This suggests that an order parameter of the form

$$\chi(\rho, \lambda) = \left(\frac{\partial \ln D}{\partial \ln \lambda} \right)_\rho \quad (3.22)$$

should reveal the power-law structure. We can compute this from our existing data in Fig. 3.10, and it is displayed in Fig. 3.11. As you can see, this order

Figure 3.11 A coloured plot showing the variation of the order parameter χ with λ and ρ in steady state for a system of size $L = 10$.



parameter does seem to nicely partition (ρ, λ) space into two components, divided

by a region of extremely rapid change. The order parameter χ could be regarded as some kind of susceptibility. An issue here is that it is hard to see how χ 's apparent transition will vary with system size, as $L = 10$ is a very small system, and transitions only become sharp in the limit of large systems. We could of course use it in a Monte-Carlo calculation in a larger system, but then we have the problem that it's difficult to take meaningful derivatives of noisy data. We will leave it, then, as a curiosity.

3.4 Time-Dependent Properties of Small SPM Systems

Although we have mostly concentrated on calculating steady state properties of the SPM, it is also possible to calculate some dynamical properties.

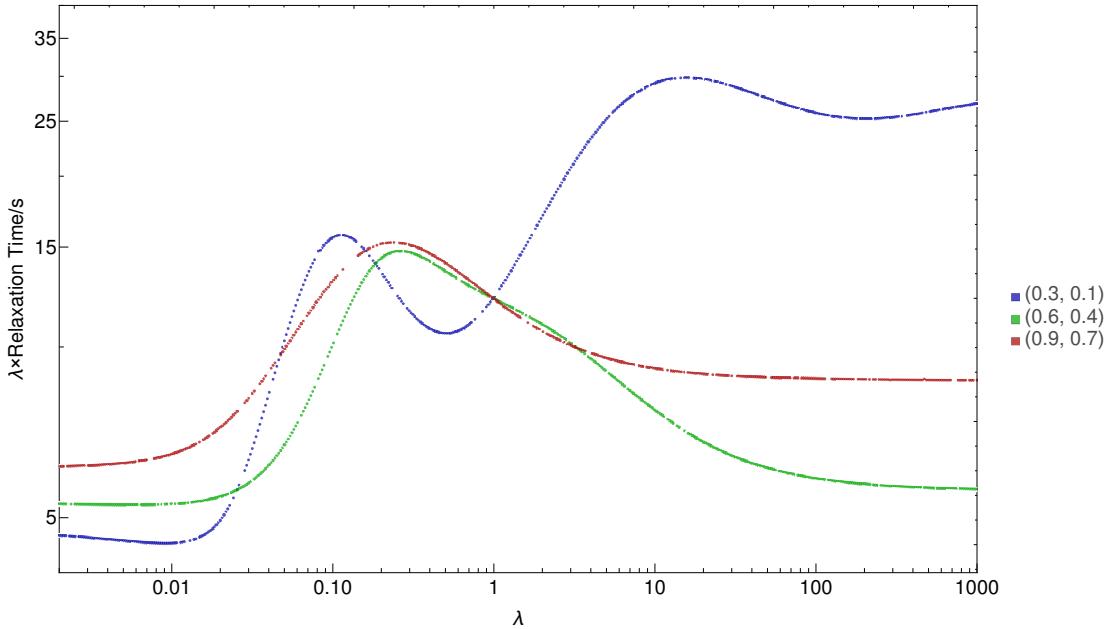
3.4.1 The Relaxation Time for the SPM

As we have seen, the eigenvalue with least negative real part effectively controls the rate at which a generically-prepared system equilibrates, by acting as a lower bound on the rate of relaxation towards equilibrium. Therefore, the reciprocal of this relaxation rate should yield a characteristic time for convergence to equilibrium, the relaxation time. We have computed this relaxation time for our three standard boundary conditions in Fig. 3.12. Notice first that we have plotted the relaxation time multiplied by λ , as the relaxation time generally scales as $\mathcal{O}(\lambda^{-1})$. One can see why we made this choice by looking at the extremes, where the lines are generally quite flat.

Observe that the relaxation times for large λ are generally a little higher than for low λ once the $\mathcal{O}(\lambda^{-1})$ dependence is taken into account. In particular, the system with low densities at the boundary is much slower to relax to equilibrium than the system with high boundary densities, which in turn is slow compared to the system with medium densities. This is presumably because the system is attempting to reach the maximal flow density we observed, $\rho = \frac{2}{3}$, and it finds it more awkward to do that the further away the furthest boundary is from $\frac{2}{3}$.

For very low λ , the difference between boundary configurations isn't so great, and we believe that the differences between the boundaries occur for the same

Figure 3.12 A plot showing the dependence of the relaxation time upon λ for a system of size $L = 10$ with boundary densities $(0.3, 0.1)$, $(0.6, 0.4)$ and $(0.9, 0.7)$. In terms of units, here we are presuming that the default diffusion rate $\tau_0 = 1^{-s}$.



reason as for the high- λ situation, only now the system is trying to fill instead of hold at $\rho = \frac{2}{3}$. For intermediate λ , things are less clear. At $\lambda = 1$, all systems equilibrate at the same rate; other than that, behaviour varies pretty wildly. Equilibration time is generally on the high side for $\lambda \in (0.03, 1)$, which is where the current looks like it's undergoing some kind of transition, so this increase in relative equilibration time could be interpreted as an accumulation of fluctuations in that regime.

3.4.2 Time-Evolution of States

Recall from Sec. 3.1 that the Master Equation (Eq. 3.1) has a formal solution given by Eq. 3.8, in which we premultiply the initial state by a matrix exponential of the TRM. Throughout this chapter we have been making use of the fact that the TRM is sparse in order to perform our computations. Thus, it would seem that we couldn't investigate the time evolution of systems, as e^A is in general not sparse even if A is sparse, and thus we would instantly run out of memory if we tried to compute anything.

However, the premultiplication of a vector by a sparse matrix can be performed *in place*, and this is implemented by Python’s routine `scipy.sparse.linalg.expm_multiply`. Therefore we have been able to calculate the time-evolution of arbitrary initial distributions. By way of an example we prepared three systems with initial uniform distributions, in which all possible configurations are equally likely, and then monitored how their spatial occupations varied as they relaxed towards equilibrium (which should always occur, as discussed in Sec. 3.1). The results are displayed in Fig. 3.13. In the plots we have included the innermost site in the boundary layer; notice that it switches to the “correct” mean occupation almost instantly, which is exactly what is supposed to happen as it represents a highly responsive reservoir. The system then gradually makes its ways toward equilibrium, starting at the boundaries and working inward, and the characteristic timescale over which this occurs seems to be in line with our results from 3.4.1.

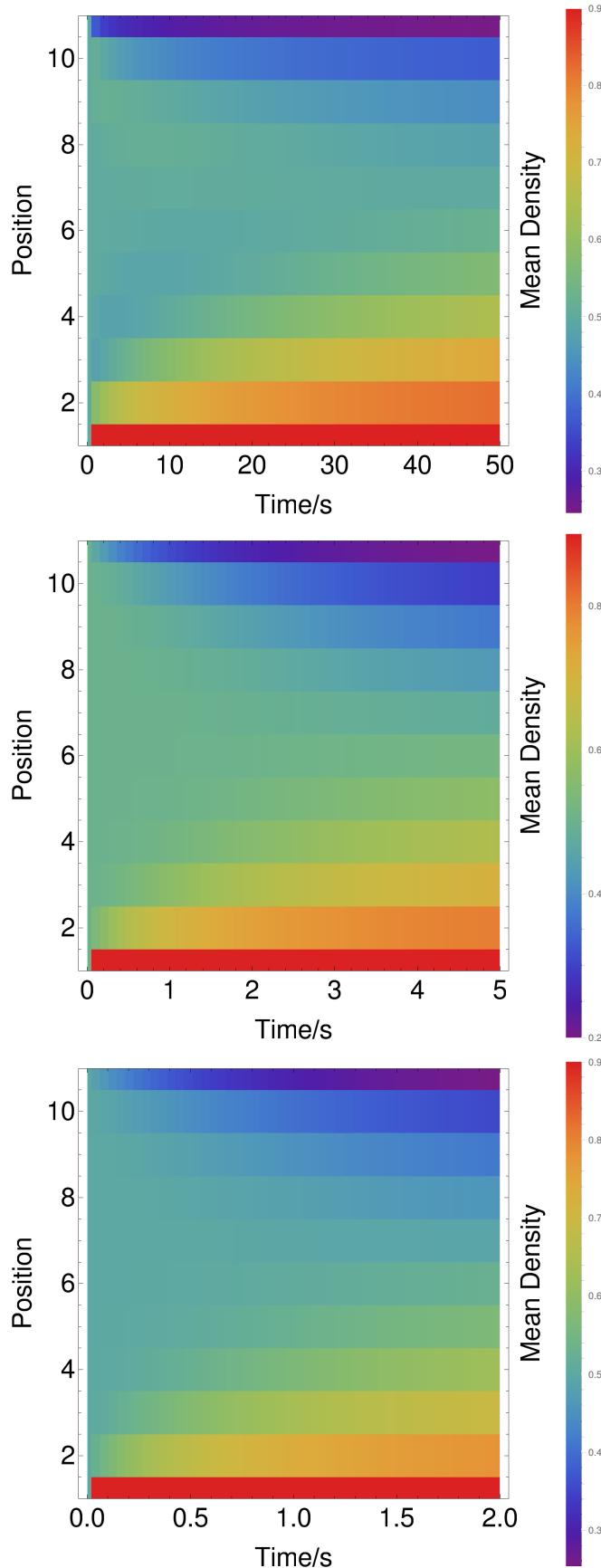
Of course, this is more of a demonstration of what this method could achieve than actually useful results. We are quite sure that with a little effort one could use it to calculate time-dependent spatial correlation functions, for example, but we simply worked out how to perform TRM calculations too late in the project to be able to use it to full effect.

3.5 Conclusions

Numerically approximating the eigenpairs of the transition rate matrix is a convenient alternative to the use of Monte Carlo methods for the computation of the properties of a Markovian statistical mechanics system. When using the TRM, we sacrifice system size for accuracy, as the memory requirement for TRM calculations scales exponentially with system size, whilst similar Monte Carlo calculations scale linearly. However, the TRM method avoids the sampling issues which often emerge when attempting Monte Carlo simulations, and can yield the relevant statistics with relative ease once the matrix computations are completed. Furthermore, we can also investigate time-dependent properties at our convenience, which can be awkward in Monte-Carlo calculations as there we don’t usually work with normal time.

The main reason we performed these calculations, however, is to investigate the current flowing due to the boundary conditions. We found that the MFT prediction that the currents drops to zero or becomes negative below a critical

Figure 3.13 Plots of the time-evolution of the density profiles of systems prepared with uniform distributions. These systems are of size $L = 10$, with boundary densities $(0.9, 0.1)$, $b_0 = 100.0$ and $\lambda = \{0.1, 1.0, 2.0\}$, going from top to bottom, respectively. The total time intervals have been adjusted to suit the equilibration time.



λ (depending upon boundary conditions) does not seem to be correct; however, we have found that the scaling of the current does seem to undergo some kind of transition, from $J \propto \lambda$ for high λ to $J \propto \lambda^3$. This is something which we can investigate further using Monte-Carlo methods on larger-scale systems, and we will do this in the next chapter.

Chapter 4

Monte-Carlo Simulations of the SPM

We now have numerical results for SPM systems using TRM analysis; however, this only allows us to study relatively small systems. In order to study larger ones, we have used Monte-Carlo methods. In this chapter, we will discuss the methods we used, the results they yielded and their meaning, with particular emphasis on what they tell us about the suspected transition between low and high- λ behaviours.

4.1 Numerical Simulations of Continuous-Time Markov Processes

Here we will discuss the theory behind the Monte-Carlo methods used to simulate continuous-time Markov processes. We will assume throughout that we have the computational means to produce pseudorandom floating-point numbers in a way which closely approximates the uniform real distribution over $(0, 1)$.

4.1.1 Purpose of Monte-Carlo Methods

We should first really describe what we mean by a Monte-Carlo method. In essence, Monte-Carlo methods refer to numerical routines in which we attempt to characterise an unknown distribution, generated via known rules, by using pseudorandom numbers in order to produce sample data which is hopefully faithful to the original distribution, at least in terms of the statistics we are trying to calculate. A good example of a commonly-used Monte-Carlo method in Physics is the Metropolis-Hastings algorithm, which in its original form is used to calculate statistics for equilibrium statistical mechanics systems.

In our situation, we wish to be able to mimic a continuous-time discrete-state Markov process. As we saw in Chap. ??, the state space for a TRM system of size L scales as $\mathcal{O}(2^L)$; thus we quickly run out of size if we try to consider exactly probability distributions, which correspond to vectors in \mathbb{R}^{L^2} . We can, however, store individual configurations, which only occupy $\mathcal{O}(L)$ space. Therefore, we need to find a way to produce trajectories through the discrete state space which sample the actual space of system trajectories well enough to allow us to access the statistics we want. Of course, there isn't a unique “best” way to do this. We have considered two contrasting methods, which differ primarily in the way in which they convert the original continuous time into discrete steps which we can use in an algorithm.

4.1.2 Evenly-Spaced Timesteps

If we wished to numerically approximate an ODE system, one might use the Euler forward or Runge-Kutta methods. These both involve discretising time simply by dividing it into evenly-sized pieces, and then converting the ODE into a discrete form by using finite differences to approximate derivatives. We need to be careful to choose a small enough timestep for the approximation to the derivative to remain good, but otherwise it is a very simple and effective approach.

We can do a very similar technique with continuous time Markov processes. In our SPM system, if we ignore the boundaries, there are two rates, 1 and λ , and our system is homogeneous. Let us represent the system with a binary array of length $L+2$, with L sites for the bulk and a site each representing the boundaries. Therefore, in order to simulate the action of the SPM as defined in preference to

appropriate section in introduction, we can use the following recipe:

1. **START.** Advance time by Δt . Pick a site, which we will call Site, (of which there are $L + 2$) at random. If the site chosen is one of the boundaries with density ρ , reset the site to be occupied with probability ρ and unoccupied with probability $1 - \rho$.
2. If Site is occupied, pick one of the two adjacent sites, which we will call Target, at random with equal probability. This will be the site we attempt to move into. If it is not, go back to **START**.
3. *If Site is not on the boundary:* If Target is occupied, go to **START**. Otherwise, consider the other adjacent site, which we will refer to as Rear. If Rear is empty, move the particle in Site into Target randomly with probability $\frac{1}{1+\lambda}$; otherwise, move the particle with probability $\frac{\lambda}{1+\lambda}$. Return to **START**.
If Site is on the boundary: If Target is outside the system, go to **START**. If Target is occupied, go to **START**. Assign an occupation value for Rear randomly, occupied with probability ρ , unoccupied with probability $1 - \rho$, where ρ is the density of the relevant boundary. Now, if Rear is empty, move the particle in Site into Target randomly with probability $\frac{1}{1+\lambda}$; otherwise, move the particle with probability $\frac{\lambda}{1+\lambda}$. Return to **START**.

We define Δt via

$$\Delta t = \frac{\tau_0}{L(1 + \lambda)}. \quad (4.1)$$

In terms of the algorithm's correctness at producing reasonable trajectories, we simply need note that the rates at which particular transitions should occur are in the correct proportions, and that the boundaries result in the correct densities in equilibrium; then, we just need to verify that the rate at which free particles move is the correct one in absolute terms, which it is, and we're done.

For Monte-Carlo methods, we generally rate their performance by the amount of computational power required to explore a given amount of the probability space. In methods in which we are exploring this space by advancing through time (and invoking ergodicity) we desire methods which move us quickly through time whilst maintaining good sampling and performing little computation.

The advantage of this method is that it is very simple; thus, there aren't too many opportunities for error when writing the code, it uses very little memory

(all calculations can be performed in-place), and each iteration should be very fast as there are very few overheads. It should also produce trajectories which are good samples of the original probability distribution we are trying to replicate.

If λ is close to 1, the probability of rejection (i.e. a step which results in no overall change to the system) is $\sim \frac{1}{2}$, and this is the situation in which the algorithm really shines; similarly it also performs well for large or small *lambda* if the system density is very high or very low respectively. For extreme λ in general however, performance drops off considerably, as we are often performing lots of calculations and advancing time very little, and thus not seeing much of the distribution simply because we aren't moving much.

We could have made this code marginally more efficient by making the more likely moves certain, and correspondingly adjusting the timestep size Δt to account for this; however, this only improves efficiency by a factor of around 2 , whilst making the code more complicated, so as we only used this method to verify the results of our main code we didn't bother. It is possible for us to get around this issue by advancing time in a variable fashion, although this comes at the cost of a little more computation per iteration.

4.1.3 The N-Fold Way, or Gillespie Algorithm

A popular way to produce trajectories for a continuous-time Markov process is the N-Fold Way, also known as BKL, or Gillespie Algorithm [1, 10, 12]. It evolves us through time as follows:

1. **START:** Make a list of all states which can be transitioned to in a single move from the current state, and the associated rates at which this occurs.
2. Weight each successor state by the transition rate into it, and then select a successor state by random selection from a uniform distribution over the weighted possible successors. Change the system state to the chosen successor.
3. Now advance time by an increment chosen from an exponential distribution whose decay rate is the sum of all of the rates of the possible transitions to a successor. Go back to **START**.

Now we just need to supply the rates *[from introduction]* that define the

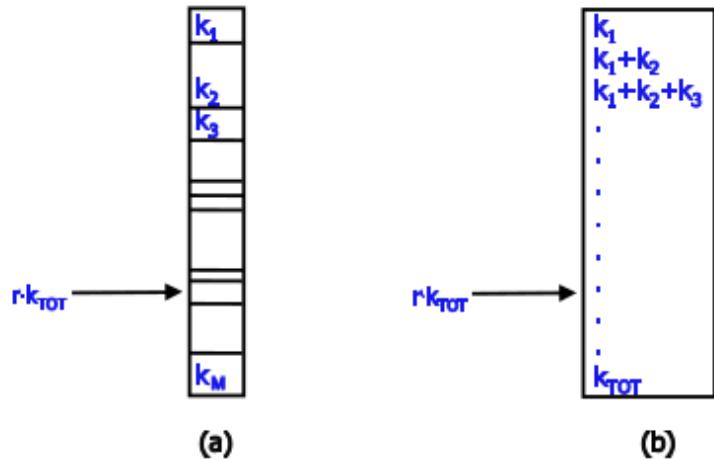
SPM, along with some additional rates describing processes at the boundary. Specifically, we use the method described in 3.2.1 to do this, whereby we have a double layer of “blinking” boundary sites and sites in the internal layer undergo the same transitions as in the bulk. However, unlike in our TRM calculations, we should not set the incoming and outgoing rates to be extremely high, as then these rather trivial processes come to dominate the calculation and cause the timesteps to be on average extremely small, wasting our computing time. Instead, we set them to be proportional to the geometric mean of 1 and λ , and thus in the language of 3.12 this corresponds to setting $B_0 = \sqrt{\lambda}$. This way the boundaries refresh often, but not too often, and should still act as suitable reservoirs.

We will get into the fine details about how the software we use implements KMC in Sec. 4.2.2. Let us instead discuss how we obtain the required probability distributions using the uniform distribution on $(0, 1)$, $U(0, 1)$:

- We can randomly choose the successor state required in step 2 by creating a list of weighted partial sums. If the transition rate from the current state the i^{th} potential successor state is k_i , then let us define $k_{\text{Tot.}} = \sum_i^n$, where n is the number of potential successors. Create the list of partial sums via $s_i = \sum_j^i k_j$, then generate the random number $u = rk_{\text{tot}}$ where r is drawn from $U(0, 1)$. We can then use a binary search to find $i : s_{i-1} \leq u \leq s_i$, and then this i indicates the successor state which has been chosen. This process is illustrated more visually in Fig. 4.1.
- In step 3, we need to generate random numbers in an exponential distribution with decay rate k_{tot} . We can do this by generating r from $U(0, 1)$, and then $w = -\frac{1}{k_{\text{tot}}} \log r$ follows the desired distribution.

I will defer to Voter (see in particular Sec. 5 of [12] for the “proof of correctness” of the method. The primary advantage of this method is that we are certain to advance time every step, so we are not potentially “wasting” steps as when we use even timestepping; this comes at the cost of having to compute which transitions are possible from the current state. For our SPM, a given state has $\mathcal{O}(L)$ possible transitions, thus the time complexity of a single timestep is $\mathcal{O}(L)$; note that our method with evenly-spaced timesteps has constant time complexity, but the size of each timestep scales as $\mathcal{O}(L^{-1})$; thus we’re not actually losing as much as it appears by using variable timesteps. Furthermore, there is the possibility that the

Figure 4.1 An illustration of the suggested method for choosing a successor state in the n -fold way. Reproduced from [12].



process by which we calculate which transitions are possible could be performed in parallel, and so the walltime cost of a single timestep in the n -fold way can end up comparatively cheap. The process of choosing a successor state once the options are found involves performing the equivalent of a search, and therefore takes $\mathcal{O}(\log L)$ time and so should be insignificant.

4.2 Implementation of Monte Carlo Methods

4.2.1 Our Implementation of a Metropolis-Hastings Algorithm with Evenly-Space Timesteps

We have written a Fortran code which implements the algorithm in Sec 4.1.2. This is stored in `location of code`. This programme initialises the system to have a particle density equal to the average of the two desired boundary densities, and then proceeds in a manner extremely faithful to the simple accept/reject algorithm.

4.2.2 KMCLib

The vast majority of our Monte-Carlo calculations have been performed using the n-fold way, described in Sec. 4.1.3. This is implemented for continuous-time Markov processes on crystalline lattices (of which the SPM is an example) in a software package called **KMCLib**, documented at [7] developed by Dr Mikael Leetmaa.

KMCLib is a Python-wrapped C++ package. This means that the frontend, where one specifies the system to be simulated, the data to be recorded, and how the simulation is run, is written in a Python script; then, when this script is run, it executes C++ code in order to represent the system and actually carry out the desired operations. Furthermore, **KMCLib** can perform calculations in parallel if so desired. Whilst there exist examples [3, 9] of kinetic Monte-Carlo codes other than **KMCLib**, we chose to use that one due to our familiarity with all of the languages involved, and preference for a Python frontend.

Of course, setting up different calculations which vary different parameters or measure different things require different scripts. Going through every script we wrote individually would cause this thesis to be around twice as long and three times more dull; therefore we have instead chosen to focus upon a single set of codes designed to perform a particular calculation, which we have annotated and included here `\location`; the intention is that a reader wishing to reproduce any of our results could do so by performing a few simple modifications to the code listed there. A more comprehensive codebase is stored at `\location`, but this is sparsely annotated working code, and so might not be very helpful.

The exemplary code which actually interfaces directly with **KMCLib** is contained within `concFlow.py`. This script takes in several command line inputs. These provide the parameters for a simulation of the SPM, with the desired value of λ , system size and boundary conditions. It then sets up the representation of the system configuration and the means to enumerate possible transitions and their associated rates, as is necessary to implement the n-fold way. The initial configuration is generated by randomly inserting particles into the system until its density is equal to $\frac{1}{2}(\rho_0 + \rho_L)$; we then perform N_{eq} KMC steps in order to equilibrate the system (in case the initial configuration we chose was highly deviant from the norm for the prescribed parameters). The actual measurements are performed by time-averaging values for system quantities (e.g. the number

of particles entering the system at one end) over N_{meas} steps, relaxing the system (in other words, performing steps but taking no measurements) for N_{req} steps, and then repeating this process N_{pass} times. This way, we can generate N_{pass} time-separated observations of, say, the total current through the system, and because we are relaxing the system between measurement runs we should not have to worry too much about the results being unduly correlated with each other, (assuming we set N_{req} high enough). Thus, we supply `concFlow.py` with the following parameters as command line inputs:

1. The particle reservoir concentration at one end of the domain, ρ_0 .
2. The particle reservoir concentration at the other end of the domain, ρ_L .
3. The value of λ to use in the simulation.
4. The system size, L .
5. The interval between measurements performed by the analysis routines, N_{anal} . This should be set to 1 in order to measure the current.
6. The number of equilibration steps, N_{eq} .
7. The number of analysis steps per pass, N_{meas} .
8. The number of reequilibration steps per pass, N_{req} .
9. The total number of passes, N_{pass} , which give separate sets of observations, performed during this calculation.
10. A timescale, δt , which indicates how often to evaluate, and to what accuracy to record times, when measuring the number of particles in the system. This should probably be small compared to the expected KMC timestep size.

In terms of the output of the code, it produces a short file summarising the input parameters, some trajectory dump information (usually redirected to `/dev/null` in order to save hard memory, which is often in short supply), as well as data taken by measurement routines. We nominate, from a suite of possible routines, which measurements we would like it to take during analysis phases. Note that in our calculations, we do not consider any quantity's value on during particular KMC steps; rather, we always average our quantities over some amount of time. This is partly because some of the quantities we are interested in do not really

have any value during a single timestep (e.g. the flux of particles through one of the boundaries), and also because the amount of time spent in particular configurations could potentially vary wildly between configurations. The amount of time spent in a particular configuration in the n-fold way is drawn from an exponential distribution with decay rate k_{tot} , as we saw in Sec. 4.1.3; thus, one could easily imagine a situation in which the transition time varies wildly. For example, say we have a system with very low λ . If this system was quite full, there would be few transitions possible, and those possible transitions would likely occur with low rates, therefore the kmc timesteps would tend to be very long. However, later during the same simulation we could find ourselves in a situation where the system is less full, and so more transitions can occur, and generally with much higher rates, leading to much shorter timesteps. Thus, we shouldn't really treat particular quantities derived from these configurations with an equal footing, as the amounts of time the system spends in each are so very different.

The precise nature of our time-averaging depends a little on the measurement in question. The types of measurements we usually perform are the following, where T is the total time elapsed during our N_{anal} step measurement run:

- **Current** We count the total number of particles which enter or leave a given boundary over the course of the measurement run. Let the number of particles entering and exiting at the 0 boundary be u_0 and w_0 respectively, and likewise for the L boundary with u_L and w_L . Then

$$J = \frac{u_0 + w_L - u_L - w_0}{2T} \quad (4.2)$$

should be a good estimate of the total current through the system during that time period.

- **Block Size Distribution** In one dimension, we can look at a configuration and count how many contiguous runs (“blocks”) of particles there are of different sizes (e.g. size 1 means a single particle sandwiched between adjacent vacancies). We can find the distribution of block sizes, weight it by the length of the associated kmc timestep, and then add this to a running total. If we do this over our N_{meas} analysis steps and then normalise, we can build a histogram of the block sizes during that time period.
- **Particle Density** Similarly, we can count the total number of particles in the system, weight it by the length of the kmc timestep, and then use

this to build another histogram of the system particle density. By keeping track of particles entering and leaving the system, it would be possible to code this very efficiently to take $\mathcal{O}(1)$ time; however, as our routine to detect block sizes scans through the system and counts as it goes along, we have just opted for a simple $\mathcal{O}(L)$ scan of the whole system for our density measurement as well.

Using these analysis routines, we can generate time-averaged values for particle density histograms, the block size distribution and the current. By calculating N_{pass} separate instances of these observables, we get N_{pass} samples from the relevant distributions, and from there we can probe the statistics of these variables.

4.2.3 Managing KMCLib Calculations in Parallel

Of course, it is one thing to have a code which can run on a laptop to produce the output of a particular simulation over the course of a day. It is quite another undertaking to run thousands of separate calculations in order to map out parameter spaces and compute derived quantities such as the diffusion coefficient.

We have been running our calculations on Edinburgh University’s **Eddie3** computing cluster. This machine does not boast the high level of processor interconnection density of **ARCHER** or the extremely high working memory of **DiRAC**; however, for the purposes of our calculations it turns out that we need neither. The KMC algorithm only stores a single state of the system under simulation at any given time, therefore its space complexity only scales as $\mathcal{O}(L)$. Furthermore, whilst KMCLib can be run in parallel mode in order to take advantage of a multithreaded environment, this isn’t actually an advantage when we wish to run very large numbers of separately-parametrised calculations, as the total amount of CPU time required remains the same, whilst incurring additional overheads associated with parallelism. Therefore, we have used a single-threaded environment for all of the calculations featured in this thesis.

In order to set up a batch of calculations, we use the following procedure, implemented by the codes stored in `kmc/1d/` within `|locationi`:

1. Created a batch of input files, in the subdirectory `jobInputs/`. In our setup, we require that files titled `testInput.i` are generated, with $i \in [1, n]$

where n is the total number of calculations to be performed. These input files are typically generated by a code such as `lambdaFlucCreator.py`, and typically the parameters which determine the overall structure of the system (e.g. L , the system size) will be held constant across calculations, whilst parameters such as the stickiness or the boundary densities will be varied between them. These input files contain a single line of code, which will be appended to `python` and called in the command line.

2. In order to actually perform the calculations, we submit them as `gridengine` batch jobs. We then run the script `kmcSubmit.sh`, which submits the nominated tasks whose input files are in `jobInputs/`, using the scripts `kmcJobArray.sh` and `initKmc.sh` for intermediate steps. `kmcSubmit` is also the place where we specify calculational parameters, such as maximum memory usage, maximum runtime, etc; these will be taken into consideration by `gridengine` when it comes to scheduling these calculations, so it is important that the maxima be relatively tight upper bounds, otherwise the calculation priority will be extremely low, assuming that the cluster in question is being simultaneously for many other calculations.
3. The jobs will then be executed, in their own time. The results will be placed in the location nominated by the input files.
4. Once the run is complete, and the data has been saved, we are then ready to process it into a more useful format, in our case a data file which can be interpreted by Mathematica, the programme we used for most of our analysis and graphing. This is done using a script such as `lambdaPostProc`. Note that such a script needs to be able to handle the fact that data may not be produced for some of the calculations (around 5% in our experience). The most likely source of the problem seems to be an issue with type conversion between Python and C++, which only seems to become a significant issue in larger calculations.

Note that throughout our calculations, we have stored data in a human-readable format, instead of in a more compressed binary format. This is because we believe that the benefit of having a human-readable format, and therefore a much greater ability to look through data and check the output, outweighs the associated memory cost (around a factor of 10 or so), especially given that any memory

reduction attained due to such a format change wouldn't change what was feasible in terms of what we can afford to store.

4.3 1D Calculation Results

4.3.1 Calculational Choices

4.3.2 Flow Patterns

4.3.3 Current

4.3.4 Density

4.3.5 Diffusion Coefficient

4.4 2D Calculation Results

4.4.1 Calculational Choices

4.4.2 Results

4.5 Conclusions

Chapter 5

Conclusions

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

Appendix A

Code Listings

A.1 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])+" \u2202Exp ["+str(pair[0]-maxAdj)+ "b] \u2202 + "
    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])+" \u2202Exp ["+str(pair[0]-maxAdj)+ "b] \u2202 + "
        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) + "}")

```

A.2 n -Dimensional Continuum-Limit MFT

This Mathematica script computes the current which flows between two adjacent sites (offset in the e_1 direction) in the MFT of the n -dimensional SPM; due to symmetry, this tells us what happens in an arbitrary direction. In this case n is set to 3, but it still works if changed to any positive number.

```

n = 3;
i = 1;
zero = 0*UnitVector[n, 1];
e[i_] := UnitVector[n, i];
Hess = Table[
  Piecewise[{{d2p[j, i], j > i}}, d2p[i, j]], {i, 1, n}, {j, 1, n}];
Jacob = Table[dp[i], {i, 1, n}];
p[x_] := p0 + Jacob.x + 1/2 x.(Hess.x);
rightJ = 1/
  t0 (1 - p[1/2 a e[i]]) p[-(1/2) a e[i]] (1 -
  z p[-(3/2) a e[i]]) Product[
  Piecewise[{{(1 - z p[-a e[j] - 1/2 a e[i]]) (1 -
  z p[a e[j] - 1/2 a e[i]])}, {j != i}}, 1], {j, 1, n}];
leftJ = 1/t0 (1 - p[-(1/2) a e[i]]) p[
  1/2 a e[i]] (1 - z p[3/2 a e[i]]) Product[
  Piecewise[{{(1 - z p[-a e[j] + 1/2 a e[i]]) (1 -
  z p[a e[j] + 1/2 a e[i]])}, {j != i}}, 1], {j, 1, n}];
fullJ = rightJ - leftJ + 0[a]^3;
FullSimplify[fullJ]

```

Bibliography

- [1] Bortz, A., M. Kalos, and J. Lebowitz. “A new algorithm for Monte Carlo simulation of Ising spin systems.” *Journal of Computational Physics* 17, 1: (1975) 10 – 18. <http://www.sciencedirect.com/science/article/pii/0021999175900601>.
- [2] 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>.
- [3] Hoffmann, M. J., S. Matera, and K. Reuter. “kmos: A lattice kinetic Monte Carlo framework.” *Computer Physics Communications* 185, 7: (2014) 2138–2150.
- [4] Ivanova, N. M. “Exact solutions of diffusion-convection equations.” *arXiv preprint arXiv:0710.4000* <https://arxiv.org/abs/0710.4000>.
- [5] Laisant, C.-A. “Intégration des fonctions inverses.” *Nouvelles annales de mathématiques : journal des candidats aux écoles polytechnique et normale* 5: (1905) 253–257. <http://eudml.org/doc/101975>.
- [6] Landau, L. D., E. M. Lifshitz, and L. Pitaevskii. *Statistical physics, part I*. pergammon, Oxford, 1980.
- [7] Leetmaa, M., and N. V. Skorodumova. “KMCLib: A general framework for lattice kinetic Monte Carlo (KMC) simulations.” *Computer Physics Communications* 185: (2014) 2340–2349.
- [8] Martynov, G. A. *Classical Statistical Mechanics*, volume 89. Springer Science & Business Media, 2012.
- [9] Plimpton, S., C. Battaile, M. Chandross, L. Holm, A. Thompson, V. Tikare, G. Wagner, E. Webb, X. Zhou, C. G. Cardona, et al. “Crossing the mesoscale no-mans land via parallel kinetic Monte Carlo.” *Sandia Report SAND2009-6226* .
- [10] Prados, A., J. J. Brey, and B. Sánchez-Rey. “A dynamical monte carlo algorithm for master equations with time-dependent transition rates.”

Journal of Statistical Physics 89, 3: (1997) 709–734. <https://doi.org/10.1007/BF02765541>.

- [11] Sherratt, J. A. “On the transition from initial data to travelling waves in the Fisher-KPP equation.” *Dynamics and Stability of Systems* 13, 2: (1998) 167–174. <https://doi.org/10.1080/02681119808806258>.
- [12] Voter, A. *Radiation Effects in Solids edited by KE Sickafus and EA Kotomin*. Springer, NATO Publishing Unit, 2005.