Coarsening Dynamical Systems: Dynamic Scaling, Universality and Mean-Field Theories

by

Craig K. Chapman

A thesis submitted to the
College of Science and Engineering
at the University of Glasgow
for the degree of
Doctor of Philosophy

August 2011

Abstract

We study three distinct coarsening dynamical systems ($\mathcal{CDS}$) and probe the underlying scaling laws and universal scaling functions. We employ a variety of computational methods to discover and analyse these intrinsic statistical objects. We consider mean-field type models, similar in nature to those used in the seminal work of Lifshitz, Slyozov [**68**] and Wagner [**118**] (LSW theory), and statistical information is then derived from these models.

We first consider a simple particle model where each particle possesses a continuous positive parameter, called *mass*, which itself determines the particle's velocity through a prescribed law of motion. The varying speeds of particles, caused by their differing masses, causes collisions to take place, in which the colliding particles then merge into a single particle while conserving mass. We computationally discover the presence of scaling laws of the characteristic scale (mean mass) and universal scaling functions for the distribution of particle mass for a family of power-law motion rules. We show that in the limit as the power-law exponent approaches infinity, this family of models approaches a probabilistic *min-driven* model. This min-driven model is then analysed through a mean-field type model, which yields a prediction of the universal scaling function.

We also consider the *conserved Kuramoto-Sivashinsky* (CKS) equation and provide, in particular, a critique of the effective dynamics derived by Politi and ben-Avraham [**89**]. We consider several different numerical methods for solving the CKS equation, both on fixed and adaptive grids, before settling on an implicit-explicit hybrid scheme. We then show, through a series of detailed numerical simulations of both the CKS equation and the proposed dynamics, that their particular reduction to a length-based $\mathcal{CDS}$ does not capture the effective dynamics of the CKS equation.

Finally, we consider a faceted $\mathcal{CDS}$ derived from a one-dimensional geometric partial differential equation in [**120**]. Unusually, an obvious one-point mean-field theory for this $\mathcal{CDS}$ is not present. As a result, we consider the two-point distribution of facet lengths.

We derive a mean-field evolution equation governing the two-point distribution, which serves as a two-dimensional generalisation of the LSW theory. Through consideration of the two-point theory, we subsequently derive a non-trivial one-point sub-model which we analytically solve. Our predicted one-point distribution bears a significant resemblance to the LSW distribution and stands in reasonable agreement with the underlying faceted $\mathcal{CDS}$.

Contents

1 Coarsening Systems 12

1 Introduction to Concepts and Methods 12

1.1 Coarsening: An Overview 12

1.2 Mean-Field Theories 17

1.3 System Types 19

1.4 Mean-Field Evolution Equations 20

2 Objectives and Approach 22

2.1 Summary 24

2 Aggregating, Ballistic Particle ($\mathcal{ABP}$) Models 25

1 Introduction and Background 25

1.1 Objectives and Approach 28

1.2 Summary 30

2 Introduction to the $\mathcal{ABP}$ Model 31

2.1 Overview 31

2.2 Specific Dynamics and Velocity Law 31

2.3 Canonical Problem (Non-dimensionalisation) 32

2.4 Adjusting for Convection 33

2.5 The Scaling Hypothesis 33

2.6 Numerically Validating the Scaling Hypothesis 36

3 Large Scale Simulations and Associated Distributions for $p = 1$ 38

3.1 Update Rule 38

3.2 Data Acquisition 38

3.3 Errors 40

3.4 Vacuum Interval 44

3.5 Support of the Distribution 44

4 Increasing p ($p \geqslant 1$) and Correlations 45

4.1 Vacuum Interval 45

4.2 Support of the Distribution 45

4.3 'Bump' on Distribution Tail 47

4.4 Correlations 47

5 System Dynamics for Large p and '$p \to \infty$' 49

5.1 A 'Min-Driven' System 49

5.2 Minimal movement of particles $\mathfrak{m}_i \neq \mathfrak{m}^\star$ 53

6 One Sided 'Paste-All' Model 55

6.1 Introduction to the 'Paste-All' Model 55

6.2 Discretely Distributed Mass - One Sided Paste-All Model 57

6.3 Continuously Distributed Mass - One Sided Paste-All Model 60

6.4 Solution to the Delay Differential Equation 66

7 Model Comparisons 70

7.1 Overall Conclusions 71

3 CKS Equation, Numerics and Dynamical Systems 73

1 Introduction and Background 73

1.1 The Conserved Kuramoto-Sivashinsky Equation 74

1.2 Proposed Dynamical System 78

1.3 Objectives and Approach 78

2 Numerical Solution of CKS Equation 79

2.1 Explicit update - Fixed Grid 79

2.2 Adaptive Grid Methods 86

2.3 Implicit-Explicit Update - Fixed Grid 103

3 Benchmarking the Code 107

3.1 Linear stability 107

3.2 Emergence of Two-Scale Solution and Coarsening 111

3.3 Outer Parabolic Structure 114

3.4 Inner Boundary Layer Structure 119

3.5 Size of Time Step and Grid Spacing 126

4 Coarsening Pathway and the Dynamical System 130

4.1 Types of event 130

4.2 Comparison with the Dynamical System 131
5 Overall Conclusions 136

4 Hill-Valley Facet Model 139

1 Introduction and Background 139
1.1 Faceted Systems: A Coarsening Dynamical System 140
1.2 Theoretical Predictions and Mean-Field Models 141
1.3 Objectives and Approach 144
1.4 Summary 145
2 The Hill-Valley Facet Model 147
2.1 Overview 147
2.2 Specific Dynamics and Velocity Law 147
2.3 Finite Simulations 148
2.4 Coarsening Events and Update 149
2.5 Conservation of Total Length 150
2.6 Canonical Problem (Non-dimensionalisation) 151
2.7 The Scaling Hypothesis 151
2.8 Numerically Validating the Scaling Hypothesis 152
2.9 Data Acquisition and Simulations 154
2.10 Data Analysis 158
3 The Fokker-Planck Equation and Mean-Field Models 171
3.1 The One-point Degeneracy 171
3.2 The Fokker-Planck Equation for Valleys - Traditional Method 172
3.3 Implications for the Hill Distribution 178
3.4 The Transport Equation for Valleys - A Different Approach 179
3.5 Simplified Model and Associated One-point Theory 192
4 Overall Conclusions 198

References 200

List of Tables

3.1 Common MMPDEs 94

List of Figures

1.1 Typical spinodal decomposition: a numerical example from [**129**] 15

2.1 Typical ballistic particle model evolution 34

2.2 Numerical validation of scaling hypothesis 37

2.3 Mass distribution at three separate times 41

2.4 Typical comparison of histogram and kernel density methods 41

2.5 Mass distribution for $p = 1$ 43

2.6 Distribution for various values of p 46

2.7 Vacuum interval sizes and 5% cut-off values 46

2.8 Correlation coefficients for various particle separations 48

2.9 Correlation coefficients for the ‘Paste-all’ model 56

2.10 Scaled distribution for discrete mass, one-sided Paste-all model 65

2.11 Scaled distribution for continuous mass, one-sided Paste-all model 65

2.12 Solution to delay differential equation 69

2.13 Solution to delay differential equation with numerical data 69

2.14 Comparison of delay equation solution and distributions for various p . . . 72

3.1 Typical CKS step configuration 75

3.2 Typical CKS simulation 77

3.3 Physical and computational domains and $\Psi(\xi)$ 90

3.4 Example of $\Psi(\xi)$ 91

3.5 Second derivative data for varying number of grid points 102

3.6 Linear instability on domain of size $L = 20$ 109

3.7 $L = 40$ CKS simulation, early time 112

3.7 $L = 40$ CKS simulation, late time 113

3.8 $L = 100$ CKS simulation, early time 115

3.8 $L = 100$ CKS simulation, late time 116
3.9 $L = 100$ CKS simulation inner region paths 117
3.10 Shifted profile and inner region trajectories for $L = 40$ and $L = 100$ 118
3.11 $L = 10$ CKS simulation to check parabola consistency 120
3.12 $L = 60$ CKS simulation ('away from' event) 121
3.13 $L = 60$ CKS simulation ('nearer' event) 122
3.14 Comparison of solution and log(cosh) structure, $L = 10$ 124
3.15 Comparison of solution and fit curve in inner regions, $L = 60$. 125
3.16 Trajectories for $L = 30$, $h = 1/25$ and $\Delta t = h^2/10$. 127
3.17 Trajectories for increased Δt and increased h. 128
3.18 Trajectories for $h = 1/25$ and $\Delta t = h^2/5$. 129
3.19 CKS 'parabola merge' simulation 132
3.20 Inner trajectories for merging parabolas. 134
3.21 Fading inner location compared with dynamical system 135
3.22 Comparison of PDE inner trajectories and dynamical system 137

4.1 Numerical validation of scaling hypothesis 153
4.2 Annihilation procedure 156
4.3 Numerically generated *up* and *down* distributions 159
4.4 Overlaid *up* and *down* distributions 160
4.5 *The* one-point distribution 160
4.6 Uniform initial condition distributions 162
4.7 Rejection sampling initial condition distributions 164
4.8 Product distribution compared to rejection sampling initial hill distribution 164
4.9 *Hill* and *valley* distributions in cartesian and polar coordinates 166
4.10 Extrapolated functional form at origin, $\Phi(\theta)$ 168
4.11 Correlation coefficients for neighbouring lengths 170
4.12 Diagram of separate dynamical system and PDE approaches 180
4.13 Area of integration $\Omega_{\epsilon,\delta}$ 184
4.14 Typical characteristic curves for hyperbolic equations 190
4.15 Comparison of one-point theory and data 199

Acknowledgements

I would like to thank my supervisor Stephen Watson for his support and encouragement over the last four years, without which this thesis would not have been possible. Thanks also go to Steven Roper whose help and advice on numerous occasions has been invaluable; particularly regarding programming and gnuplot! Further, I am grateful to the University of Glasgow for the financial support provided throughout my studies.

My thanks also go to my Mum, Dad, Russell, Donna and all of my extended family for their endless belief that I would actually make it this far. Thanks to Suz for her understanding and support, especially during the tedium of writing-up. Thank you to all of my friends.

Many thanks to everyone in the department who I have had the pleasure of meeting and chatting with over the years. Finally, I would like to acknowledge all of the occupants of 522 who have, without question, made the last four years so memorable!

Declaration

This thesis is submitted in accordance with the regulations for the degree of Doctor of Philosophy at the University of Glasgow.

Chapter 1 and the introductory sections of Chapters 2, 3 and 4 contain background and preliminary information. The rest of the material contained in this thesis is the author's original work, except where explicitly indicated otherwise.

Chapter 1

Coarsening Systems

1 Introduction to Concepts and Methods

1.1 Coarsening: An Overview

Coarsening is the emergence of larger scale structures from smaller ones combined with an overall decrease in the number of objects. Coarsening phenomena are exhibited in a wide range of contexts including crystalline rocks [**70**, **74**], nano-suspensions and nano-emulsions (the 'ouzo' effect) [**85**, **104**, **115**], ice cream [**25**], quantum dots [**37**, **94**, **112**], astrophysics [**81**, **100**] and a variety of other materials science contexts [**91**, **117**]. With such a wide range of areas which exhibit coarsening in one way or another it is perhaps understandable that such processes have become an important topic of discussion. Understanding these processes allows for physical processes to be controlled [**69**] to counter certain negative aspects, such as coarsening of medicine in a nano-suspension, or promote the positive aspects, such as controlling quantum dot size. This thesis will primarily deal with so-called *coarsening dynamical systems* [**48**, **68**, **118**, **122**] and so we begin by briefly introducing some of the common concepts and ideas that will be encountered throughout, together with an overview of some classical coarsening problems.

A well studied coarsening process is *Ostwald Ripening*, which dates back to the work of Wilhelm Ostwald in 1896 [**82**]. The study of Ostwald ripening received increased attention some sixty or more years later [**68**, **118**] as it was shown to have important implications in a wide variety of areas, including many of those listed earlier. To demonstrate the ideas, concepts and theory that underlies much of the research into coarsening systems, an abbreviated version of the history of Ostwald ripening is informative. In particular, since it is known to be representative of the coarsening process in many physical systems; e.g. ice cream, the ‘ouzo’ effect and quantum dots, as mentioned earlier. Ostwald ripening involves the increase in average size of regions of one phase (component) of a two-phase mixture suspended in the other with the loss of smaller regions as they are depleted by larger ones. More specifically, Ostwald ripening occurs where the quantity of one of the components is small compared to the other. The quantity of one component per unit of total volume is known as the phase fraction, k ($0 \leqslant k \leqslant 1$), and so, in particular, Ostwald ripening in a binary mixture generally concerns the case where the phase fraction of the minority component is very small compared to the other component, i.e. $k \ll 1/2$; note that $k = 1/2$ in a binary mixture would simply mean that we have an equal quantity of both components. As time progresses it is energetically favourable for some of these regions of the minority phase to grow, at the expense of other regions which shrink and disappear, and eventually the mixture will consist of fewer larger regions of the second phase. The mechanism by which regions grow is known to be by material, in the form of individual atoms, detaching from relatively high-curvature regions of their respective components and then reattaching, upon diffusing through the other phase, to its relatively low-curvature components. More specifically, the energy in the system is associated to the total perimeter of the regions with an energetic tendency to reduce this total perimeter in the system [**117**]. Considering the much simplified 2D case of a set of circles, one can immediately see that the larger the circle the smaller the curvature and hence larger circles tend to be more favourable. Note also that in a full physical system regions may grow to such an extent that they touch one or more other regions. In such a circumstance the system will coalesce these regions into one single region, reducing the overall perimeter, and thus furthering the coarsening process.

To illustrate these ideas in an everyday context we consider ice crystals in ice cream, which are known to display Ostwald ripening behaviour [**25**]. When the ice cream leaves the factory the ice crystals present are small and distributed evenly throughout the system. Water molecules diffuse through the ice cream from ice crystal to ice crystal and, in exactly the manner of Ostwald ripening, it is thermodynamically favourable for the ice to form relatively large crystals on average as opposed to leaving many relatively small ones. The Ostwald ripening process is ultimately the reason why ice cream becomes ‘crunchy’ when left in the freezer too long.

Another well studied class of problems that exhibit a type of coarsening process are thermodynamically unstable mixtures that undergo *spinodal decomposition*. Spinodal decomposition is in effect the separation into distinct phases (regions) of a mixture of two or more components; see, for example, [**7**, **17**, **18**, **29**, **65**, **92**, **129**]. Driven purely by the diffusion of the material in the system, spinodal decomposition occurs throughout the material and is not localised to any specific points. In much the same way as Ostwald ripening, the system forms distinct regions composed of one of the components of the mixture. Over time these regions, which generally will start small and numerous, combine and form larger intertwined regions of material [**19**, **65**, **66**]. Cahn and Hilliard [**19**] provided the first insight into these types of problems, ultimately proposing what is now known as the *Cahn-Hilliard* equation as a candidate for understanding and further probing the chemical processes seen experimentally. The Cahn-Hilliard equation models the evolution of a mixture from a homogeneous initial state right through to the final coarsening regime. One can clearly see the coarsening of such systems; a large number of small regions combine into fewer larger regions, e.g. [**3**, **129**]. A typical numerical simulation from [**129**] is shown in Figure 1.1 displaying spinodal decomposition where the phase fraction is $k = 1/2$. During spinodal decomposition of a mixture with phase fraction around $k = 1/2$ we see the emergence of a labyrinthine structure, whereas in the small phase fraction regime, $k \ll 1$, we see localised regions of the minority phase shrinking or growing in the matrix of the majority phase.

There are two key features which are frequently reported in the literature, the concept of a *growth law*, or *scaling law*, and *dynamic scaling*. In coarsening systems the growth in time t of an associated characteristic length scale (a length which is representative of the scale of the system such as the mean size) is often deduced or derived. Simple arguments based on the fundamental properties of the individual system often predict power law type behaviour [**62**]. For example, in the case of Ostwald ripening under the assumption that regions never meet and coalesce, more specifically in the limit of small phase fraction, $k \to 0^+$, it is known that the mean radius of clusters, $\bar{R}(t)$, obeys the scaling law $\bar{R}(t) = \mathcal{K}t^{1/3}$, where $\mathcal{K}$ is some constant [**68**, **118**]. Characterising and understanding scaling laws and in particular finding bounds on the constant $\mathcal{K}$, or similar constants in other systems, is a central topic of research into coarsening systems, see for example [**14**, **27**, **60**–**62**, **83**].

Another key property of many coarsening systems is a *dynamic scaling* effect, alternatively called *statistical self-similarity* [**75**], which essentially means that statistics of the

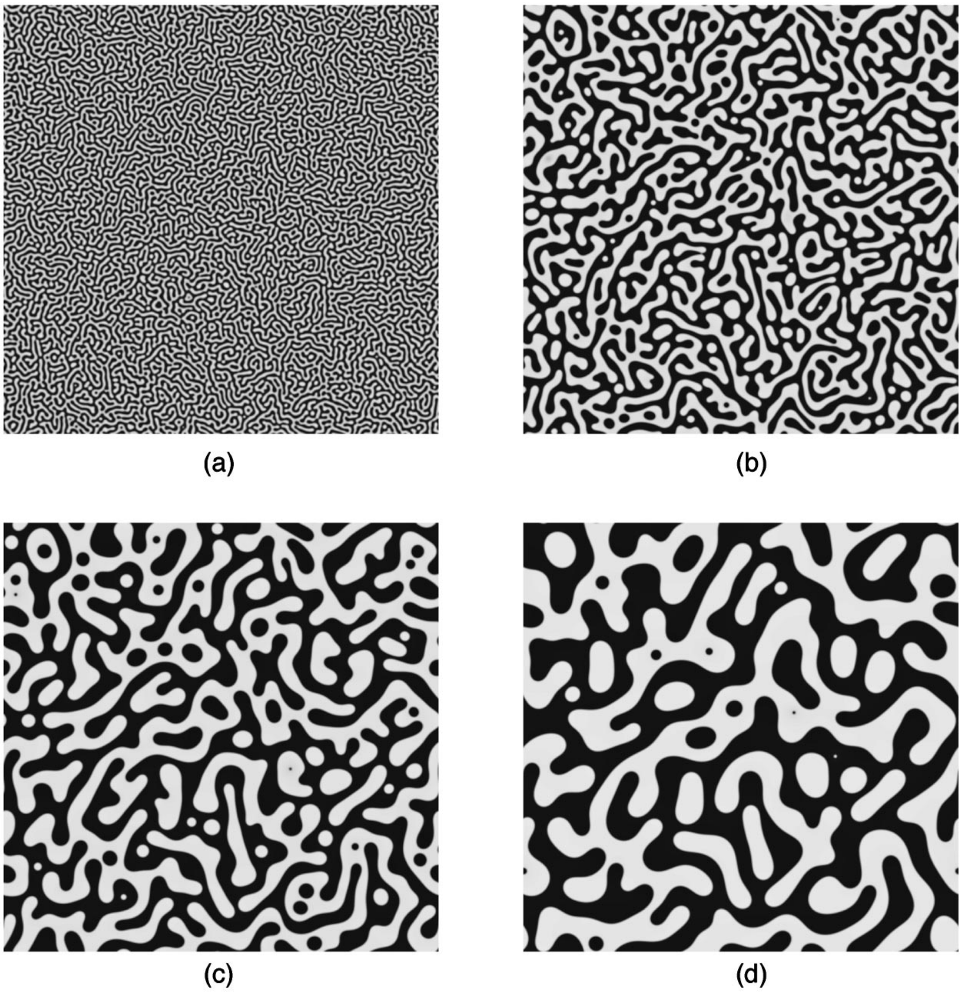


Figure 1.1 – A typical example of a numerical simulation of a binary mixture undergoing spinodal decomposition. Images captured at various time points: (a) t=100, (b) t=2000, (c) t=10000, (d) t=30000, [**129**].

system remain the same when suitably scaled by the characteristic length scale. The precise circumstances under which a system will display dynamic scaling are not well understood, its frequent appearance being a mystery which invokes many interesting questions on coarsening systems and other systems alike [**6**, **14**, **26**, **35**, **41**, **110**, **114**]. As an example of this behaviour, if we take a system where each object has an associated mass, $\mathfrak{m}_i(t)$, we may suppose that at time t the one-point distribution of mass is given by $\rho(m,t)$. If we then take the characteristic length scale of the system, $\langle\mathfrak{m}\rangle(t)$ say, we may find that the one-point distribution of mass satisfies the following relationship,

$$\rho(m,t) = \frac{1}{\langle\mathfrak{m}\rangle(t)}\mathcal{P}\left(\frac{m}{\langle\mathfrak{m}\rangle(t)}\right),$$

where $\mathcal{P}$ is a *universal one-point scaling function* for the system. Note that the function $\mathcal{P}$ is constant in time and so this function is an invariant of the evolution. The consideration and formulation of the function $\mathcal{P}$ in appropriate contexts forms a cornerstone of much of the research into these types of systems, since knowledge of relatively little statistical information would allow us to understand much of the overall evolution of the set of objects [**11**, **21**, **22**, **26**, **30**, **45**, **48**, **68**, **72**, **80**, **86**, **110**, **118**]. In fact, this property can be roughly discerned in Figure 1.1, taken from [**129**], by considering the later two images, wherein the system has settled into the scaling regime, following the transient evolution away from the homogeneous initial data. If one imagines 'zooming in' on the earlier time stage it is not hard to imagine that qualitatively they would both look the same, that is, under suitable spatial re-scalings, the size and shape of regions as well as the overall pattern would appear similar. This neatly illustrates the definition of dynamic scaling.

The key questions which arise in coarsening models are frequently related to understanding these scaling laws and universal scaling functions and so let us now consider the types of approach commonly used in studying these coarsening systems. One can take a purely analytical approach to try and understand the evolution. This type of analysis often discovers various bounds on the characteristic length scale and growth laws within the system, an excellent overview of which can be found in [**14**], and similarly in [**62**]. This type of approach can be very informative, although since they focus only on the characteristic length scale of the system they tell us little about the statistics of the ensemble, e.g. [**60**, **61**, **83**].

Conversely, so-called *coarsening dynamical systems* ($\mathcal{CDS}$) [**122**] allow us to understand both the associated scaling laws *and* statistical information about the system. Many coarsening systems are derived from multi-scale partial differential equations [**48**, **120**, **122**] which due to their multi-scale nature are often complex to simulate and understand. By considering the effective dynamics of these systems it is possible to reduce them to a derived coarsening system which maintains the essential coarsening structure of the overall system; this is what we term a $\mathcal{CDS}$. This derived dynamical system contains both derived dynamics and derived coarsening laws which can then be analysed independently from the original problem in order to understand the underlying scaling laws and the statistical structure of the scaling functions. By adopting this method it is hoped that we can discover statistical information which is meaningful in the context of the full governing system. It is this approach which will primarily be adopted in this thesis. Note, however, that a $\mathcal{CDS}$ need not have a physical context to make it worthy of study. Indeed, the system encountered in Chapter 2 is developed free from physical context, purely as means to further understand coarsening systems in general.

1.2 Mean-Field Theories

In the seminal theory of Lifshitz, Slyozov [**68**] and Wagner [**118**] (LSW theory) a *mean-field* theory is developed to approximate the statistics of the radial-distribution of droplets which arise in the very low minority phase Ostwald ripening regime, $k \to 0^+$. The theory assumes that each radial cluster of the minority phase evolves in relation to a *mean-field*, which reflects the mean cluster size, and isn't fundamentally influenced by its immediate neighbours. Ultimately, the growth rate of any individual cluster is effectively determined by a relation between its size relative to the mean.

Many coarsening systems, including Ostwald ripening and spinodal decomposition, are shown experimentally and computationally to exhibit ensemble type behaviour, that is, objects such as a universal scaling function exist [**11**, **12**, **14**, **21**, **22**, **34**, **36**, **58**, **68**, **72**, **81**, **86**, **117**, **118**, **130**]. We would like to better understand and explain this type of behaviour. In particular, we would like to be able to take a given $\mathcal{CDS}$ and extract information on the associated scaling laws and universal scaling functions. *Mean-field theories* are one approach developed to achieve this. In a typical $\mathcal{CDS}$ the entire ensemble of objects will evolve together. More specifically, each object is not free to evolve entirely on its own. Objects might, for example, evolve in a different manner depending on the 'size' of its neighbouring objects, e.g. [**30**, **80**]. Mean-field theories are then derived models which introduce the concept of a *mean-field* which captures information about the entire ensemble, such as the characteristic length scale, but retains no information on the specifics of any of the individual objects. This system-wide mean-field is then used to simplify object evolution by comparing an object's individual information to the information contained within the mean-field, before evolving accordingly. In LSW theory [**68**, **118**], for example, the mean-field contains information on the mean radii of all the clusters and evolution proceeds by those smaller than this size shrinking and those larger growing. By considering a mean-field model of our $\mathcal{CDS}$ we hope to be able to use the simplified structure to predict and understand statistical information such as the scaling laws and universal scaling functions.

So, the general principle of a mean-field theory is to take an individual object or possibly objects and assume that instead of being surrounded by the rest of the system it only 'feels' the mean-field effect, which approximates the rest of the system. As an example consider a water droplet as part of a 'breath figure' [**30**], seen for example on cold window panes, but instead of having each droplet surrounded by others of all shapes and sizes, as in the physical system, we can consider what effect the surroundings have on each particle on average, the mean-field effect. In this case the mean-field simply contains information on the mean droplet size and as a result we have reduced the full system evolution to essentially a discussion about individual droplet size relative to the current mean.

One of the key merits of mean-field theories is their ability to predict self-similarity for a given system. Ostwald ripening, for example, displays this scaling behaviour [**68**, **117**, **118**]. The LSW theory famously predicted the form of *the* scaling distribution of cluster radii for Ostwald ripening, together with its associated scaling law. As already mentioned, the origins of self-similarity, or dynamic scaling, remain mysterious and so it is interesting to note that mean-field theories, such as the LSW theory, can provide some insight into this interesting property. Self-similarity itself should allow for the full complex system to be better understood by consideration of relatively few statistics [**62**, **80**, **121**]. Use of a mean-field type theory to simplify a complex system often allows us to probe the specific nature of the *single* scaling distribution, predicted by self-similarity, as well as the associated scaling law. Further to this, the mean-field theory often allows us the opportunity to predict this single distribution directly.

Contained within many mean-field theories are assumptions on the interactions and relationships within the system. Often it is assumed that neighbouring objects are uncorrelated, sometimes referred to as a 'random order approximation' [**21**, **76**]. In some systems, an object's evolution might depend on the immediate surroundings and so statistically we require information on the joint probability of finding various pairs, triples etc. of objects that lie next to each other. To perhaps clarify this idea further, consider the probability distribution of ordered pairs $\rho(x, y)$, the probability of finding two neighbouring objects with respective attributes x and y. A simple consequence of probability theory then says that if x and y are independent then the joint distribution is the product of the single one-point distributions, that is $\rho(x, y) = \rho(x)\rho(y)$. Note that this *wouldn't* be the case if they *weren't* independent but the approximation is then to assume that they *are*, in fact, independent and so we can then make use of this kind of factorisation. Such factorisations are frequently employed when developing mean-field theories [**11**, **21**, **22**, **30**, **72**, **80**, **86**] and often prove fruitful in situations where objects are not truly independent but the correlation between them is weak.

Ultimately mean-field models serve as a tool to gain information on the statistics of a $\mathcal{CDS}$ by a reduction to a simpler closed model on n-point statistics (for some n). Developing these types of theory for a given coarsening system offers the possibility to predict characteristics, such as dynamic scaling, and statistical information, such as *the* scaling distribution and scaling law.

1.3 System Types

1.3.1 Coarsening Dynamical Systems

Coarsening dynamical systems ($\mathcal{CDS}$) entitles systems which evolve via some prescribed deterministic dynamics together with some rule to govern coarsening events e.g. annihilation or coalescence. Here we have a set of objects which evolves via a prescribed dynamics, which generally makes direct reference to neighbouring objects and/or the system as a whole. The Ostwald ripening models within the LSW theory [**68**, **118**] are a paradigm example. This thesis is primarily concerned with the study of $\mathcal{CDS}$. A frequent source of $\mathcal{CDS}$ are multiple-scale partial differential equations, from which effective dynamics are derived and it is these dynamics which form the $\mathcal{CDS}$ [**19**, **46**, **48**, **120**, **122**]. A paradigm example of this approach, and of key significance for our future studies, is the $\mathcal{CDS}$ derived from the convective Cahn-Hilliard equation via a novel matched-asymptotic analysis, appearing in [**122**].

In any general coarsening system we want to understand and theoretically predict the scaling laws and universal scaling functions. Studying $\mathcal{CDS}$ is useful because their relatively simple closed dynamics is generally more amenable to analysis than considering the original multiple-scale partial differential equation alone. More specifically, we can often more easily simulate and gather data from the $\mathcal{CDS}$ than the original partial differential equation. Since the $\mathcal{CDS}$ is derived from the original problem we expect the statistics of both to closely match and so considering the $\mathcal{CDS}$ allows us to better understand the statistics of the original problem. Developing predictions of the universal scaling functions and scaling laws using the $\mathcal{CDS}$ can then be translated back into an understanding of the original problem, thus informing us in ways which may not be directly accessible by considering the original problem alone.

1.3.2 Probabilistic Coarsening Systems

Probabilistic coarsening systems essentially have the same overall setup as previously mentioned; a set of objects and generally some neighbour relations or perhaps an ordering within the system [**1**, **123**]. The coarsening in these systems is different from the dynamic type of system and comes from use of a prescribed probabilistic rule which updates the system in some particular way, e.g. [**22**, **86**]. A neat example of this kind of system is given by so-called *min-driven* systems where events occur sequentially and at each stage we find the smallest object(s) (assume, for example, that each object has mass) and then combine this with one or more other objects in the system, chosen by use of the probabilistic rule [**21**, **30**, **43**, **72**]. We then have several possible options such as; do we combine to a neighbouring object or any object in the system, is the object combined as a whole or split into smaller parts, etc. [**86**]? This leads to a wealth of possible systems, each of which displays coarsening as expected since, by definition, we continually remove small objects and create larger ones. Interestingly, in some circumstances it is useful to identify a coarsening dynamical system with a probabilistic counterpart.

1.4 Mean-Field Evolution Equations

A common outcome of research within coarsening systems is the derivation of an equation governing the evolution of the associated probability distribution which can hopefully be solved to predict this distribution [**11**, **30**, **48**, **80**]. A prediction of the dynamical system scaling distribution together with the associated scaling law form the key phenomena to understand. As already mentioned, this is one of the main outcomes of the LSW theory; under dynamic scaling a prediction of the scaling distribution for Ostwald ripening. Consideration of these evolution equations often shows that they support a scaling ansatz solution which in turn allows for a time dependent evolution to be reduced to a time independent equation for the single universal scaling function, e.g. [**11**]. In a general system we want to understand and theoretically predict the distribution of some aspect of the system, e.g. mass or length, but interactions or correlations within the system may make this a particularly difficult task. LSW theory was the first example which used a mean-field to predict dynamic scaling and find the scaling laws and one-point scaling distribution, in this case, of cluster radii. Since then, similar methods have proved fruitful in a range of coarsening systems, e.g. [**1**, **11**, **14**, **21**, **22**, **30**, **48**, **72**]. For a given coarsening system, one writes down an evolution equation based on the specific dynamics of the system. Where the dynamics of the system are complex but with limited correlations between objects, one can make use of an appropriate mean-field model to attempt to make some headway.

Smoluchowski's coagulation equation [**116**] and the *Fokker-Planck equation* [**50**] are two equations which are frequently used as the basis for mean-field evolution equation discovery, with the former used primarily in probabilistic systems where there is little or no dynamics and the latter in deterministic dynamical systems. These two equations essentially capture the same information but the model equations are modified to take account of the specific nature of the system involved. We refrain from exploring these in detail here and instead prefer to simply discuss their use and general outcome. See the introductory material presented in Chapter 2 for more information on Smoluchowski's equation and Chapter 4 for the Fokker-Planck equation.

In both cases these equations provide a general framework for writing down an evolution equation for the probability distribution of a given system. By considering how the objects in the system change in time, either by a probabilistic rule or driven by some velocity, we hope to be able to characterise all the ways in which objects can disappear (sinks) and all the ways in which objects can be created (sources) during the system evolution so that as time evolves we know how many objects of a particular size remain. In both the probabilistic and the dynamic case the equations effectively contain terms which directly capture the rate of these events and whether we gain objects of a particular size, or indeed lose them, and/or characterise the deterministic flow governed by the dynamics. Often we can make direct use of these equations if we know the relative probabilities, rates of events or velocity laws as appropriate, e.g. [**11**, **80**]. Characterising these sinks and sources is equivalent to understanding the boundary conditions on the domain on which the distributions are defined.

Interestingly, despite there being many systems which evolve in different manners and by entirely different means, often one finds that qualitatively the distribution of objects predicted by some mean-field theories are the same, for example the one-point distributions found in [**11**] versus [**68**, **118**] (LSW theory). This leads us to believe that there are so-called *universality classes* of coarsening systems, a name used to indicate that despite the separate problems being fundamentally different, their underlying statistical structure is very similar. This is evidently useful since it allows us to tie together systems which otherwise might be considered unique and develop a theory which is, in fact, common to both. Later we will discuss how the distributions from Chapter 3, Chapter 4 and the prediction made in the LSW theory can be considered to be, in some sense, within the same universality class.

2 Objectives and Approach

We begin, in Chapter 2, by considering a simple model which displays coarsening behaviour, an aggregating, ballistic particle ($\mathcal{ABP}$) model. In this model we have a set of point objects, or *particles*, each with some initial scalar attribute which we call *mass*, $\mathfrak{m}_i$. We locate the particles on a periodic line and subject them to a prescribed *velocity law* which is based on a particle's individual mass namely, $\mathcal{V}(\mathfrak{m}_i) = 1/\mathfrak{m}_i^p$, where p is a fixed positive integer. Each particle's differing speed causes them to collide and merge with one another, conserving mass.

This $\mathcal{ABP}$ model exhibits dynamic scaling with a scaling law of the form $\langle\mathfrak{m}\rangle(t) \sim t^{1/(p+1)}$, where p is from the velocity law and $\langle\mathfrak{m}\rangle(t)$ is the mean mass. We derive this result using a simple scaling argument and find further agreement by conducting an extensive numerical study. In this case, we are able to show two features of the dynamic scaling; that the system reaches a unique scaled distribution and that the system displays the stated power-law exponent. Confirmation and discussion of the resulting distributions and law is made with use of further numerical simulations. Following this, we probe perhaps the most interesting aspect of this type of model by varying the velocity exponent p. Modification of this exponent causes significant change in both the distribution and the scaling law. Both of these aspects are probed further by use of a series of numerical simulations.

We continue by showing that in the special limit $p \to +\infty$ the system behaves like a *min-driven* system in that the smallest mass combines 'instantly' at each step. Therefore, this system, which for finite p values was a coarsening dynamical system, can be considered to be of probabilistic type. We then apply a mean-field theory to this system which predicts the structure of the scaling mass distribution. This predicted distribution is then compared to various finite p value distributions providing strong evidence that this is indeed the limiting distribution. These simulations and analysis confirm and extend the outcomes of [**53**].

In Chapter 3 we study the coarsening dynamics of the *conserved Kuramoto-Sivashinsky*

(CKS) equation, focusing in particular on a $\mathcal{CDS}$ claimed to represent the leading order behaviour [**89**]. We begin, for completeness, with a detailed numerical study of the CKS equation using a variety of numerical schema, including adaptive grid methods, and discuss the output therein. The multiple-scale nature of the CKS equation requires careful consideration to resolve the boundary layers. Ultimately we settle upon use of an implicit-explicit hybrid numerical schema. We then compare CKS simulation data from our chosen code to data gathered by directly simulating the proposed dynamical system, ultimately finding that the suggested dynamics do not contain a sufficient amount of information to properly capture the full system evolution. Evidence of this is presented by making a direct comparison of numerical data.

In the final chapter, Chapter 4, we consider Watson's coarsening dynamical system derived from a 1D geometric partial differential equation in [**120**]. It involves a sequence of facet lengths l_i which evolve via the dynamics,

$$\mathcal{V}(l_i) = (-1)^i \left(\frac{1}{l_{i+1}} - \frac{1}{l_{i-1}} \right);$$

see also [**122**] where the same dynamics is derived from the convective Cahn-Hilliard equation, and [**119**] for the link to an underlying geometric PDE. It further involves a novel ternionic coarsening event deduced from the underlying partial differential equation [**119**, **120**, **122**].

We present extensive numerical simulations which not only confirm the dynamic scaling of one-point statistics already observed in [**120**], but go further to identify dynamic scaling of a variety of n-point statistics where $n = 2$ or $n = 3$. Reflecting the dichotomy in the dynamics between even and odd lengths, we demonstrate the existence of two distinct two-point distributions. Namely, one for even-odd ordered pairs (l_{2i}, l_{2i+1}), and one for odd-even ordered pairs (l_{2i-1}, l_{2i}).

A novel feature of Watson's $\mathcal{CDS}$ is that it does not admit any obvious one-point mean-field treatment. As a result of this breakdown we therefore seek to develop a higher-order mean-field theory focusing in particular on formulating a two-point theory. Unlike in the one-point case we succeed in deriving a Fokker-Planck equation for the two-point distribution of odd-even pairs. We exploit an interesting symmetry property in this two-point model to extract an effective one-point theory. We are able to explicitly calculate the associated scaling state's one-point distribution, which compares reasonably, though not exactly, with our numerical data on the original $\mathcal{CDS}$.

2.1 Summary

In summary we present analysis of three distinct coarsening dynamical systems. In all three of the following chapters we present a wide range of numerical data to both support and guide further analytical efforts. In Chapter 2 we present a simple ballistic particle model, and numerically demonstrate that in a special limit the system behaves like a min-driven probabilistic system. Chapter 3 provides a critique of dynamics already derived from a specific PDE context, and in particular we numerically demonstrate a significant disparity which invalidates the claimed dynamics. In the final Chapter 4 we develop a novel two-point mean-field theory for Watson's material-science inspired $\mathcal{CDS}$, and succeed in the explicit prediction of a one-point distribution which compares reasonably with direct simulation of the $\mathcal{CDS}$.

Chapter 2

Aggregating, Ballistic Particle ($\mathcal{ABP}$) Models

1 Introduction and Background

Aggregation models, sometimes called coagulation models, of the type we consider here have been studied since at least the beginning of the twentieth century and have maintained the same essential structure since then, e.g [**116**]. The basic set up involves a set of objects (particles, clusters, lengths etc.) which generally evolve in some manner together with a rule which governs the process of object aggregation, that is the joining of two or more objects. Unlike other processes such as Ostwald ripening where an individual object might slowly decrease in size over time, due to loss of individual atoms for example, here we are concerned with the case where an object is redistributed as a whole onto one object or perhaps in parts onto several objects.

As a simple example, consider a set of sticks of various lengths, pick one by some means, random or otherwise, and glue it to the end of one of the other sticks, again chosen by some means. Thus we have removed a stick from the set and made another stick longer. In this example each object has no intrinsic evolution, the only possible events are these joining events, and the aggregation procedure takes an entire stick without splitting. This simple example demonstrates the essential aspects of an aggregation model; a set of objects (sticks) and a rule for aggregation (gluing). Many such examples can be constructed or derived in various contexts such as physical chemistry, astronomy and biological systems. More specifically aggregation models have been used in aerosols, clouds and fog [**38**, **96**, **98**], schools of fish [**77**], deep ocean particles [**59**, **71**] and even the development of structure in the universe [**2**, **100**]. In each case the objects and rules vary but all follow the same simple pattern of aggregation. Frequently there is no intrinsic evolution of the objects in these models, only these discrete events (although they may move around between events), and so in the wider context of coarsening systems we consider them to be of the probabilistic coarsening type. In fact these types of system could be viewed as the prototypical probabilistic coarsening system. Note that, as might be expected, they do satisfy the essential features of a coarsening system; a reduction in the number of objects combined with an overall increase in the 'size' of the remaining objects.

The first significant contribution to the theory came from the seminal work of Smoluchowski [**116**] who showed interest in aggregation models. The lasting outcome of his research was a means to predict statistical information, such as the particle size distribution, and effectively described what is now known as the *Smoluchowski coagulation equation*, a general framework for predicting statistical properties of a system. The Smoluchowski coagulation equation allows for the calculation of a predicted evolution of a number density distribution by taking the rate at which events happen, or are predicted to happen, together with information on the current distribution and then calculating the rate of change by considering appropriate gains and losses of objects, for example; are particles of a particular size created and at what rate does this happen and equally are they removed or involved in an aggregation, and at what rate? There are two equations which fall under the Smoluchowski name, one which involves discrete valued objects (think objects with integer mass) and a second which has continuous valued objects (think real valued stick lengths). Fundamentally they have the same aim but slightly modified statement to cover the different distributions of the individual objects. Each equation governs the number density $n(x,t)$, where the number of objects with parameter in $[x, x+dx]$ is then $n(x,t)dx$ and where x represents the 'size' of the object and t time. In the case of discretely valued objects the equation then reads,

$$\frac{\partial}{\partial t}n(x,t) = \frac{1}{2}\sum_{y=1}^{x-1} K(y, x-y)n(y,t)n(x-y,t) - n(x,t)\sum_{y=1}^{\infty} K(x,y)n(y,t),$$

where $n(x,t)$ is then defined continuously in time and at discrete spatial values. In the continuously valued case the equation has a slightly different form,

$$\frac{\partial}{\partial t}n(x,t) = \frac{1}{2}\int_0^x K(y, x-y)n(y,t)n(x-y,t)\;dy - n(x,t)\int_0^{\infty} K(x,y)n(y,t)\;dy,$$

where in this circumstance $n(x,t)$ is defined continuously both temporally and spatially. Note that in each case time is truly continuous. In both cases K is known as the coagulation kernel and can take the same form regardless of which equation is used. The coagulation kernel in effect describes the rate and likelihood at which particular events happen. The first term on the right hand side of these equations governs the creation of an object with size x from one of size $x - y$ and one of size y and is dependent on the rate this happens, $K(y, x-y)$, and the number of those components present, $n(y,t)$ and $n(x-y,t)$. The half present in these terms is just to ensure we don't double count each aggregation event, this is the source term. The second term in each equation, which is the sink term, describes the loss of objects of size x by collision with smaller objects, that is an object of size x is no longer of that size if it is involved in an aggregation event. Again this term is simply dependent on the number of such objects already present and the associated aggregation rates. Varying the formulation of these kernels allows for various types of process to be explored, some typical examples include,

$$K(x,y) = 1, \quad K(x,y) = x + y \quad \text{and} \quad K(x,y) = xy,$$

where the first constant kernel is a process which has a fixed rate regardless of object size and for the remaining two the chance of events depends on the size of the objects. Many others such kernels exist and have been studied, for further detail and background see, for example, [**1**, **123**]. Following this general pattern of characterising sinks and sources allows many such systems to be analysed. Dynamic scaling is often exhibited in aggregation models and this can aid the process of understanding the size distribution and ultimately gaining a better view of the system as a whole.

Mean-field type arguments like those discussed in Chapter 1 are often employed in these type of coarsening systems. Primarily this is achieved in the typical manner by assuming that the objects surrounding the chosen aggregating object are, in some sense, mean or uncorrelated. Another way of looking at this is to reconsider the stick example from earlier and suppose that the sticks are in some random order. Then, once we select a stick to join to another, instead of picking another at random we pick a neighbouring stick. Assuming there is no correlation to the neighbouring sticks seems consistent with them being placed randomly, that is, since they are in random order there isn't any connection between a stick and its neighbour. If, however, instead of being placed randomly the sticks were, in fact, placed in a particular order we might expect the neighbouring sticks to have some relationship to the chosen stick e.g. the sticks could be in 'large-small-large' order. The simplest mean-field idea is then to assume that these correlations do not exist, sometimes called a 'random order approximation' [**21**, **76**]. Often in this context this type of argument is not necessarily explicitly outlined but hidden in a general statement about choosing the second stick, which the first is joined to, at random. Of course, if we are simulating a physical system this might not be completely accurate as the objects we are considering might have some ordering or correlations between them. Examining these correlations in the system by gathering simulation data allows us to consider their impact on the system as a whole. A system which exhibits little or no correlations between objects would be a good candidate for the simplest mean-field approach.

Several attempts have been made to characterise and predict scaled distributions for a wide range of similar systems with varying levels of success [**12**, **21**, **22**, **34**, **36**, **58**, **72**, **81**, **130**] and we hoped to be able to achieve this here. In these similar systems, however, much like the example of sticks mentioned earlier, there is no dynamics between events and so the whole procedure is essentially probabilistic in nature. Further to that these entirely probabilistic type models often have some very simple procedure for selecting which object is to be aggregated next, in many cases this is the minimum sized object. These *min-driven* systems [**21**, **22**, **30**, **43**, **72**] take the minimum sized object and combine it to another before repeating. In the context of the earlier stick example this means the first method of selection is simply to find the shortest stick. Where no such method of selecting the next object to collide exists there is increased difficulty in understanding the probabilities associated to coarsening events.

1.1 Objectives and Approach

This chapter is concerned with a class of aggregating, ballistic particle models and in particular considers the probability distributions and features therein. Our model consists of a set of $\mathcal{N}$ particles located on a periodic line, each with individual mass $\mathfrak{m}_i$, $i \in \{1, 2, \ldots, \mathcal{N}\}$. Each particle is then subject to the *velocity law*, $\mathcal{V}(\mathfrak{m}_i) = 1/\mathfrak{m}_i^p$, for some fixed, common, positive value of p. As different particles have different mass there will be a variety of velocities and ultimately this will lead to the collision of two or more particles. When particles collide we see the justification of the 'aggregating' part of the model name as the particles simply merge into one, conserving mass. We make no consideration of a particle's shape or physical size but simply assume the mass is contained at a single point location on the line. Thus two (or more) particles which find themselves at the same location simply add mass together and become one particle. The system then continues from here with the current particle mass always dictating the velocity of the remaining particles. With the system and velocity law now known we can investigate the dynamic scaling properties of the system. A simple heuristic argument leads to the scaling $\langle\mathfrak{m}\rangle(t) \sim t^{1/(p+1)}$ for the mean mass, $\langle\mathfrak{m}\rangle(t)$, in the system. We then take these predicted values for various p and check their consistency against numerical simulations.

Now that we have our base system set we proceed by first fixing $p = 1$ and probing the scaled distribution of mass in the system. A simple check of the distribution at various times confirms the dynamic scaling property and allows us to talk about *the* distribution associated with each value of p. By this we mean the unique scaled distribution which is considered to be the distribution associated to the full system ensemble if instead of being on a finite, periodic line it were on the infinite line. We detail the method of gathering and averaging statistics across a large number of independent finite simulations which allows us to probe the makeup of the associated infinite system. We then consider several of the features displayed in this distribution. Of particular interest is the emergence of a *vacuum interval* near the origin. In a small region close to the origin we see that there are no particles present of these small sizes. In practise this means that below a certain size, relative to the mean, there are no particles of this size. This appears to be as a direct result of the particular velocity law applied since small mass moves faster than larger mass and so the smallest mass in the system is most likely to quickly hit a neighbouring particle thus removing it from the system. As the mean grows and in general particles have grown in size we still see the same behaviour since if there are any small masses remaining they quickly catch their neighbours and are removed.

Following on from this we consider the distributions as $p \to \infty$ and compare the set of distributions in particular paying attention to the features exhibited. The size of the vacuum interval is considered across the range of p along with the apparent finite support of the distribution and the emergence of a bump on the tail of the distribution. Consideration of these features leads us to believe that for large p values the system approaches some limiting system and so we aim to explore this possibility. By showing that the behaviour of the system can essentially be controlled for large enough p we do indeed see that in this circumstance the system behaves as though it were a so-called *min-driven* model. We characterise this model and using a Smoluchowski type approach we are able to predict the associated distribution for this min-driven model. Subsequent to our analysis it was found that this model was derived in a much different context, that of coarsening droplets on a thin film [**30**]. We can then draw comparisons to the finite p cases from which we can see strong evidence to support the claim that this is the limiting distribution as expected.

1.2 Summary

We begin in Section 2 by specifying a class of ballistic particle models which vary via a parameter p in the associated velocity law. We detail the rules for aggregation and the reduction of an infinite set of problems to one scaled canonical problem for each value of p. In the context of these ballistic models we propose a scaling rate for both the mean length and mean mass, based on a simple heuristic argument, and consider numerical validation of these scaling rates. These scaling rates are shown to depend on the parameter p in the velocity specification and we demonstrate good agreement between these predicted values and those from numerical simulations.

Following on from this, in Sections 3 and 4, we carry out extensive numerical simulations of the system for various p values and in particular we focus on $p = 1$ to demonstrate the features of the mass distribution which we will consider across a range of values of p. Correlations in the system help us understand the statistical makeup of the set of particles and so we numerically investigate if any such correlations do exist.

As we shall demonstrate the distribution of mass appears to approach some universal distribution and we conjecture that this is found by taking the limit as $p \to \infty$ in the velocity law. In Section 5 we then deduce that for large p the system behaves like a so-called *min-driven* system in that the particles of smallest mass are strictly those which collide next whilst the other particles in the system remain essentially fixed. This leads into Section 6 where we consider a model of coarsening known as the *paste-all* model which we analyse and from which we deduce a mean-field type prediction of the mass distribution. Simulation data from the paste-all model together with various data from the ballistic particle model are then compared to this predicted distribution in Section 7 showing a increasingly good agreement as p is increased, as expected.

2 Introduction to the $\mathcal{ABP}$ Model

2.1 Overview

The model we study consists of a set of particles on a line such that each particle has some individual mass $\mathfrak{m}_i$ and location x_i, with i an indexing of the particles; $i \in \mathbb{Z}$. The particles are then ballistic in nature in that each particle moves with a velocity prescribed by some function, $\mathcal{F}(\mathfrak{m}_i, p)$, and maintains this velocity until a collision takes place. When particles collide their masses are combined and one new particle is formed, conserving mass, thus at each collision there is a net loss of one particle and so the system coarsens. Initially the system is seeded with a random distribution of particle masses, via use of a uniform distribution, and with uniformly distributed location.

2.2 Specific Dynamics and Velocity Law

Each particle moves in a prescribed manner based on its mass alone, therefore the velocity, $\mathcal{V}(\mathfrak{m}_i)$, of a particle of mass $\mathfrak{m}_i$ must be specified. The choice of law used for the system is the following,

$$\mathcal{V}(\mathfrak{m}_i) := \mathcal{F}(\mathfrak{m}_i, p) = \frac{dx_i}{dt} = \frac{1}{\mathfrak{m}_i^p}, \tag{2.2.1}$$

where p is some fixed number common to all particles and we shall generally consider $p \in \mathbb{N}$. We call p the *velocity exponent* and, as already mentioned, discussion is restricted to the case where $p > 0$. We see, therefore, that particles of 'large' mass move slower than 'small' ones and, by considering the standard orientation of $\mathbb{R}$, that all particles drift to the right. Collisions occur when faster particles catch and meet their right hand neighbour(s)[1]. When particles collide they form a new particle with mass the sum of the colliding particles, hence there is no net loss of mass from the system and so mass is conserved. It is clear that this model is a coarsening dynamical system, derived from simple ballistic dynamics and aggregation of particles as the coarsening rule.

In principle this model can now be simulated, however, the fact that the system is still infinite in nature, since the particles live on an infinite line, presents a problem. To get round this we restrict attention to some finite region and then impose that this region has periodic boundary conditions with the hope that with enough particles this will mimic the 'full' system. If we take a region of length $\mathcal{L}$ which is highly populated

[1] It is possible but rare for more than two particles to all meet at the same time, depending on the spatial configuration and the associated masses

with a finite number of particles, $\mathcal{N}_\star$, then we expect simulations on this region to mimic the corresponding infinite system so long as there exists a *separation of scales*, by this we require that the average distance between particles, $\langle\mathfrak{l}\rangle(t)$, is much less than the length of the domain, in other words $\langle\mathfrak{l}\rangle(t) \ll \mathcal{L}$. Since in the finite system $\langle\mathfrak{l}\rangle(t) = \mathcal{L}/\mathcal{N}_\star$, we must equivalently have $\mathcal{L}/\mathcal{N}_\star \ll \mathcal{L}$, and hence $1 \ll \mathcal{N}_\star$. Thus if we have a large number of particles on the finite domain we expect the *separation of scale* condition to hold. Note that if the total mass in the finite system is given by $\mathcal{M}$, then we can define the following two quantities,

$$\langle\mathfrak{m}\rangle(t) = \frac{\mathcal{M}}{\mathcal{N}(t)} \qquad \text{and} \qquad \langle\mathfrak{l}\rangle(t) = \frac{\mathcal{L}}{\mathcal{N}(t)},$$

where $\mathcal{N}(t)$ is the current number of particles in the system. One further point we note, as far as simulations are concerned, is that the length of the domain $\mathcal{L}$ and the total mass of the system $\mathcal{M}$ can be any (positive) number, however, by suitable scaling of the equations it is possible to show that the system can be reduced to the following canonical problem.

2.3 Canonical Problem (Non-dimensionalisation)

In this section we discuss the non-dimensionalisation of the problem. To do this we consider the following generic scalings of mass, position and time,

$$\mathfrak{m}_i = \mathfrak{M}_i\mathcal{M}, \qquad x_i = X_i\mathcal{L}, \qquad t = T\hat{t}. \tag{2.3.1}$$

Inserting these scalings into the velocity law (2.2.1) we have,

$$\frac{\mathcal{M}^p\mathcal{L}}{\hat{t}}\frac{dX_i}{dT} = \frac{1}{\mathfrak{M}_i^p}.$$

Then by taking the free parameter $\hat{t}$ and setting it equal to $\mathcal{M}^p\mathcal{L}$ we are returned to the original velocity law, in non-dimensional form, namely

$$\frac{dX_i}{dT} = \frac{1}{\mathfrak{M}_i^p}.$$

We also note that these scalings reduce the domain to unit length and the total mass of the system to 1 and hence we have our canonical problem.

Canonical Problem:

Any problem of this form, with length of domain $\mathcal{L}$ and total mass of the $\mathcal{N}_\star$ particles $\mathcal{M}$, can be reduced to the canonical problem of $\mathcal{N}_\star$ particles with total mass $\mathcal{M}_\star = 1$ and length of domain $\mathcal{L}_\star = 1$ by a suitable scaling of location, mass and time, as in (2.3.1).

The form of the velocity law remains unchanged.

Therefore, we can reduce a whole class of problems to simply simulating one for each velocity exponent, i.e. simulating $\mathcal{N}_\star$ particles on a periodic, unit-length domain ($\mathcal{L}_\star = 1$) with unit total mass ($\mathcal{M}_\star = 1$),

$$\frac{dx_i}{dt} = \frac{1}{\mathfrak{m}_i^p}, \qquad x_i \in [0,1), \qquad \mathcal{M}_\star = \sum_{i=1}^{\mathcal{N}_\star} \mathfrak{m}_i = 1.$$

2.4 Adjusting for Convection

We have already mentioned that the particles all drift to the right but we now wish to adjust for this general convection as follows. Between collisions we can calculate the average velocity of the particles, namely,

$$\langle \mathcal{V} \rangle (t) := \frac{1}{\mathcal{N}(t)} \sum_{i=1}^{\mathcal{N}(t)} \mathcal{V}(\mathfrak{m}_i), \tag{2.4.1}$$

where $\mathcal{N}(t)$ is the current number of particles in the system. We then make use of this average to counter the the fact the system tends to always drift to the right. We do this by redefining each particle's velocity by subtracting the average velocity from each, hence,

$$\frac{dx_i}{dt} = \mathcal{V}(\mathfrak{m}_i) - \langle \mathcal{V} \rangle (t) := \frac{1}{\mathfrak{m}_i^p} - \langle \mathcal{V} \rangle (t). \tag{2.4.2}$$

As a result, particles whose velocity is larger than average will move to the right, smaller than average to the left and average will remain fixed. It is easy to show that the time to a collision between two (or more) particles is dependent on the difference in their speeds and hence the dynamics of the system remains unchanged, since the constant (between collisions) $\langle \mathcal{V} \rangle (t)$ cancels. This slight modification allows a more straightforward method of visualisation since we no longer have a situation where particles always drift to the right. A typical illustration of the evolution of the system can be found in Figure 2.1.

2.5 The Scaling Hypothesis

Here we present an analytical argument for the scale invariance of the model. Loosely speaking if a system exhibits scaling symmetry then, statistically, the scaled system at a later stage should be indistinguishable from the unscaled earlier stage. To see if this holds for this system we consider the following scalings on length, mass and time,

$$\mathfrak{l} \to \lambda \mathfrak{l} \qquad \mathfrak{m} \to \lambda^r \mathfrak{m} \qquad t \to \lambda^q t.$$

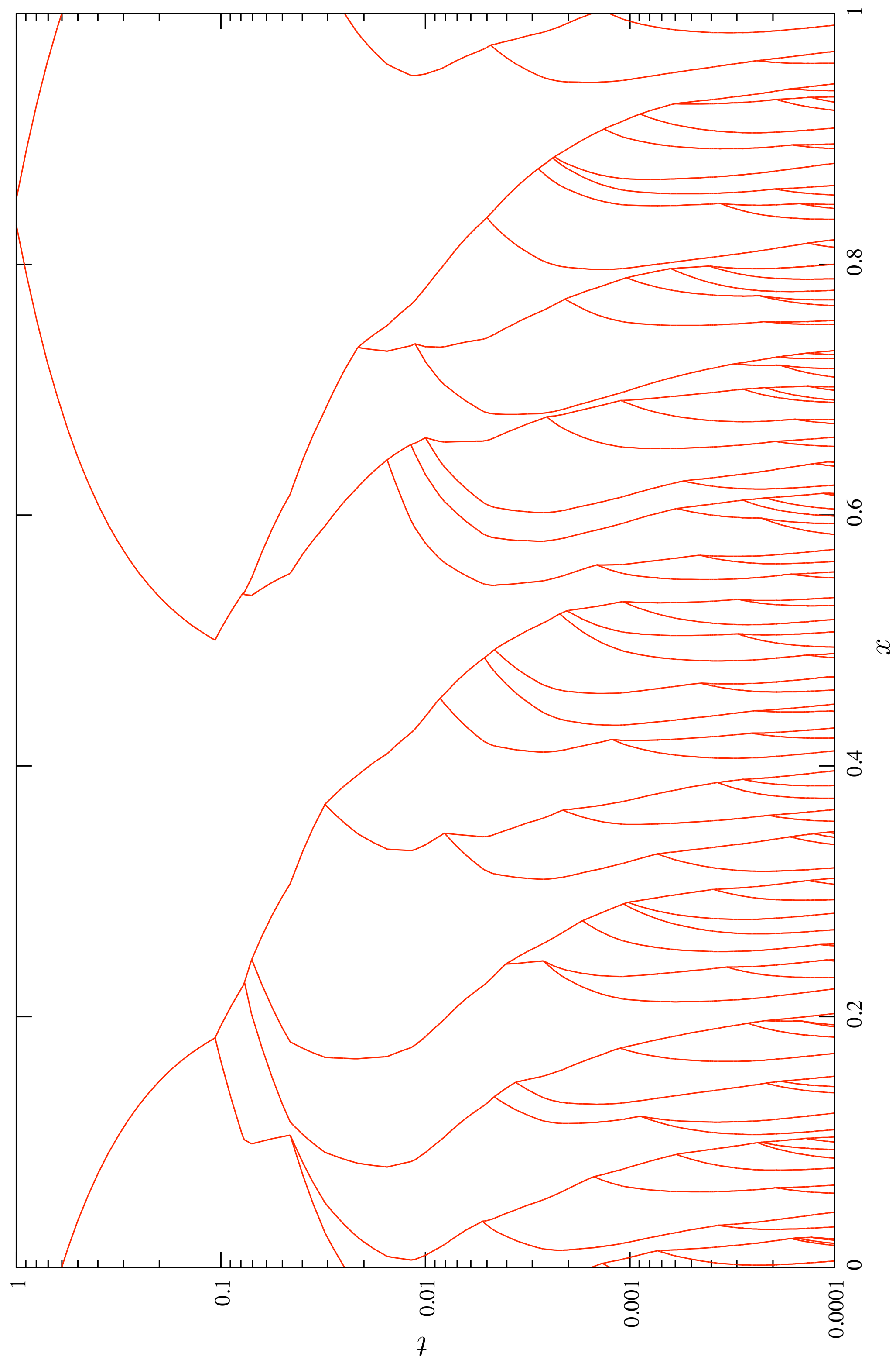


Figure 2.1 – Typical evolution diagram showing the particle positions (red lines), aggregation events (intersection of lines) and the adjustment for convection (particles can move left and right) with the time axis, t, on a log scale.

We want to consider the exponents r and q and try to establish fixed values for these. Since the system coarsens, the average length of the system will increase in time. We want the later time to be statistically indistinguishable from the earlier time so we need to scale the length of the system so that the average length remains constant. This, however, will violate the fact that the mass per unit length is *constant* (one of the constraints of the canonical system) and so we are forced to scale mass by the same factor, hence $r = 1$.[2] Secondly, the scaled system needs to evolve on an adjusted time scale to again remain statistically indistinguishable from the unscaled version. If we consider the above scalings in the velocity law we have the following,

$$\frac{\lambda^{p+1}}{\lambda^q}\frac{dx_i}{dt} = \frac{1}{\mathfrak{m}_i^p},$$

and so to keep this invariant under the scaling we are forced to choose $q = p + 1$. Hence we have the following scalings of length, mass and time,

$$\mathfrak{l} \to \lambda\mathfrak{l} \qquad \mathfrak{m} \to \lambda\mathfrak{m} \qquad t \to \lambda^{p+1}t. \tag{2.5.1}$$

Now, let us suppose that the mean mass, $\langle\mathfrak{m}\rangle(t)$, is described by some function of time, $f(t)$, so that we have,

$$\langle\mathfrak{m}\rangle(t) = f(t).$$

This description must also hold true after scaling the variables, as in (2.5.1), so we must have,

$$\langle\lambda\mathfrak{m}\rangle(t) = f(\lambda^{p+1}t),$$

which simplifies as following, since λ is just some multiplier inside a mean,

$$\lambda\langle\mathfrak{m}\rangle(t) = f(\lambda^{p+1}t).$$

This multiplier λ will vary with time as the system evolves so we make a particular choice of λ to simplify the right hand side. By choosing $\lambda = t^{-\frac{1}{p+1}}$ we see that the right hand side reduces to some unknown constant,

$$t^{-\frac{1}{p+1}}\langle\mathfrak{m}\rangle(t) = f(t^{-1}t) = f(1).$$

From this we can easily read off the following dependence,

$$\langle\mathfrak{m}\rangle(t) \sim t^{\frac{1}{p+1}}. \tag{2.5.2}$$

[2] Consider, for example, that $\lambda = 1/2$. Then the length of the scaled system would be 1/2 but the total mass on that length is still 1, hence we are forced to scale mass by 1/2 to be consistent.

The same argument holds for the average length, $\langle\mathfrak{l}\rangle(t)$, and so we have the same dependence on t. This leads us to the following predicted scaling results,

$$\langle\mathfrak{m}\rangle(t) \sim t^{\frac{1}{p+1}} \quad \text{and} \quad \langle\mathfrak{l}\rangle(t) \sim t^{\frac{1}{p+1}}. \tag{2.5.3}$$

2.6 Numerically Validating the Scaling Hypothesis

We present numerical data to validate the scaling hypotheses (2.5.3). The system was simulated once per velocity exponent and the data for *number of particles*, $\mathcal{N}(t)$, and current *time*, t, gathered. We simulate the problem is in its canonical form and so the average mass and length are both equal to $1/\mathcal{N}(t)$. In Figure 2.2 we see strong evidence that the scaling hypothesis does indeed hold by plotting $1/\mathcal{N}(t) \equiv \langle\mathfrak{m}\rangle(t) \equiv \langle\mathfrak{l}\rangle(t)$ against t on a log-log plot

Figure 2.2 also highlights two other points of note. First, the starting number of particles in each subsequent simulation was ten thousand as this was seen to be sufficient for the system to relax into the scaling regime (the straight areas which closely match the slope of the dotted lines). Inputting a higher number of particles simply 'extends' the whole diagram but provides no additional information. We see that two thousand particles could have been used but nevertheless still found it computationally efficient to use the larger ten thousand particle initial condition. Second, we see that the scaling regime has set in at different times depending on the velocity exponent, however, the fraction of particles remaining in each case appears to be constant. We make use of this fact and consider the system to be scaling when a tenth of the initial particles remain, based on starting with ten thousand, and for it to persist until around a fortieth remain, shown as the shaded area in Figure 2.2.

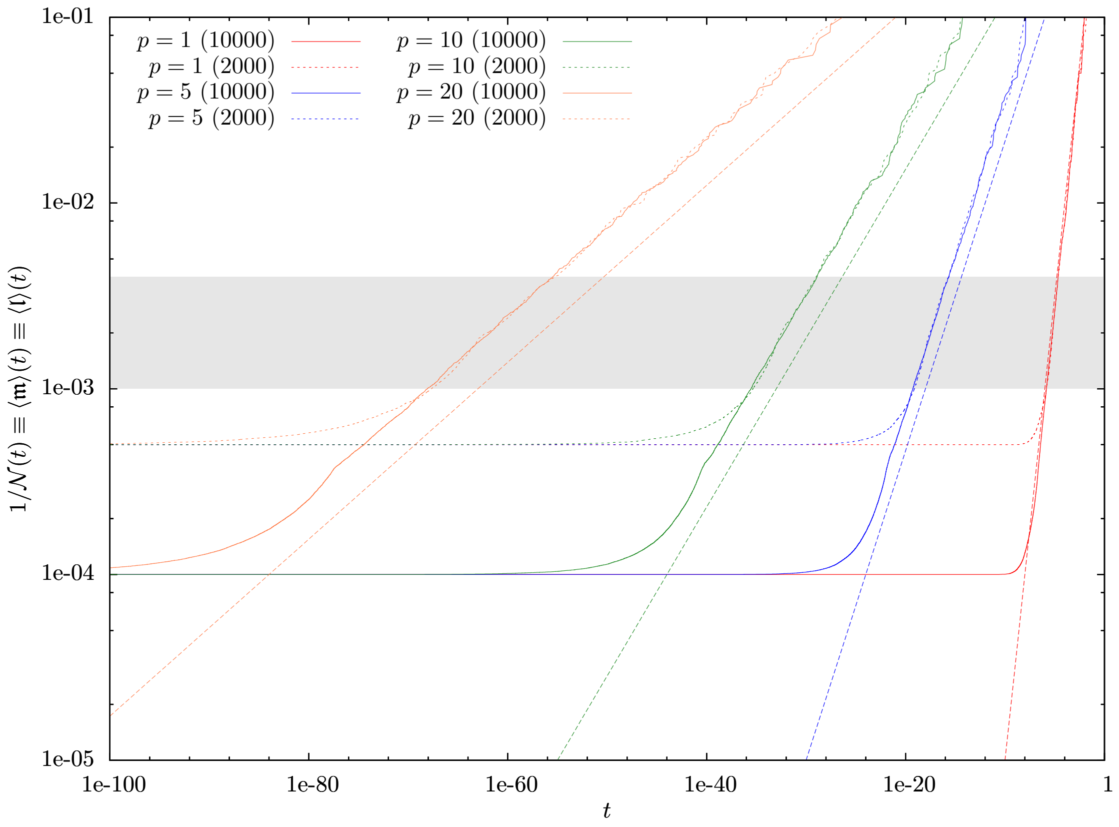


Figure 2.2 – $1/\mathcal{N}(t) \equiv \langle\mathfrak{m}\rangle(t) \equiv \langle\mathfrak{l}\rangle(t)$ plotted against time, t, for various p values on a log-log scale showing strong evidence to support the scaling hypothesis. Long, straight dotted lines indicate the predicted scaling results from (2.5.3), $\langle\mathfrak{l}\rangle(t) \equiv \langle\mathfrak{m}\rangle(t) = t^{\frac{1}{p+1}}$, and the grey shaded area indicates the region where we believe that consistent scaling is taking place, regardless of velocity exponent. Numbers in parenthesis indicate the initial number of particles

3 Large Scale Simulations and Associated Distributions for $p = 1$

In this section we summarise the findings for simulations with $p = 1$. The initial configuration in each case was ten thousand particles each with a random mass and a uniform distribution of locations. The initial configuration of locations was found to have no impact on the long term behaviour so there is no penalty in choosing one fixed configuration. As mentioned earlier there is no need to start with more particles as ten thousand was found to be sufficient to allow the system to relax into the scaling state. The problem was then scaled into the canonical form, i.e. total mass and length normalised.

The same essential procedure is used for each value of p simulated so we give detail of the update rule and data gathering methods here only.

3.1 Update Rule

The simplicity of the system means that we can easily calculate the time to the next collision of particles. After each event we simple calculate this collision time, Δt, update the whole system by this time Δt, aggregate the appropriate particles and repeat. There is no need to take small time steps or use a more accurate integration step since between collisions each particle has fixed velocity.

3.2 Data Acquisition

Data acquisition was carried out in two distinct ways - a standard *histogram* method and the *kernel density* (KD) method. For both it was found that a more suitable '*time*' coordinate for collection of data was the fraction of particles lost. So, as already mentioned earlier, we choose to collect data within the region $\mathcal{N}_\star/10$ to $\mathcal{N}_\star/40$, where again $\mathcal{N}_\star$ is the initial number of particles, as this is where we believe consistent scaling to be taking place.

3.2.1 Histogram Method

Each mass of the system at the recording time was compared with the average mass at that time. So for each mass $\mathfrak{m}_i$ we calculate $\overline{m}_i$ where,

$$\overline{m}_i := \frac{\mathfrak{m}_i}{\langle\mathfrak{m}\rangle(t)}, \tag{3.2.1}$$

and we then place 'one' into the appropriate bin of the histogram; typically we use two hundred histogram bins. At this point we have collected $\mathcal{N}(t)$ items into the histogram bins, where $\mathcal{N}(t)$ is the number of particles at the recording time. We then scale the height of each bar by $\mathcal{N}(t) \times b_w$, where b_w is the width of one bin, so that the histogram does reflect a true probability distribution, i.e. the area covered by the histogram bars is 1. Since we expect, and can show, that the system scales (see Fig. 2.3) we choose to sample the system at ten different times within the scaling region per run and first average these to build one distribution. These individual distributions are then averaged over thousands of runs to create a system mean.

We believe, and have seen evidence, that the average mass scales with time, so we expect this scaling to be quite informative. We see in Figure 2.3 that the distribution function plotted using the histogram method at three separate times do indeed coincide via this scaling as they are almost indistinguishable. Figure 2.3 was found by performing thousands of runs and recording at three 'times' within the scaling region. The values at the same times across all the runs were then averaged to find the corresponding curves. This figure motivates our decision to sample at ten times in the scaling region as mentioned above. Note that the curves plotted are drawn as the line passing through the midpoints at the top of each bar in the histogram. This curve plotting method will often be used when the histogram method is implemented.

3.2.2 Kernel Density (KD) Method

The second method used for data acquisition was the Kernel Density method [**13**, **101**]. For this method we gather a large list of masses relative to the current average at the recording time as in (3.2.1) but instead of binning this into a histogram with a finite number of bins we simply make a note of all these numbers. What we end up with is a massive collection of points on the $\overline{m}$ axis. The method then proceeds by placing a 'small' *kernel* function centred around each of the points on the $\overline{m}$ axis and summing these functions across all the points. Essentially if we have N points we place N kernel functions, each with integral $1/N$ so that after summing we have an estimate of the probability distribution function. Therefore if we have a set of points $\{\overline{m}_i\}$, we calculate the estimated probability distribution function, ϱ, as follows,

$$\varrho(\overline{m}; \overline{m}_i, h) = \frac{1}{Nh} \sum_{i=1}^{N} K\left(\frac{\overline{m} - \overline{m}_i}{h}\right), \tag{3.2.2}$$

where K is the chosen *kernel function*, N is the total number of points used and h is the *bandwidth*, which needs to be calculated or estimated. For our purposes we chose to use the following kernel function,

$$K(X) = \frac{1}{\sqrt{2\pi}} e^{-\frac{X^2}{2}}, \tag{3.2.3}$$

a standard Gaussian curve. We can easily read off that each kernel has integral h and so the integral over $\overline{m}$ of the estimated probability distribution function, ϱ, is exactly 1.

Unfortunately methods of finding the optimal bandwidth, h, require knowledge of the density function we are trying to estimate. To break this loop we made use of the histogram method as a basis for what we *predict* the distribution to look like. This allowed us to close in on a suitable first guess at the bandwidth. Further estimates were then found by building the distribution with a choice of input bandwidth and then calculating the optimal output bandwidth based on that distribution until a good match of input and output values was found. This value was then used to build the single estimated distribution function.

The KD method has two main advantages over the histogram method. First, the method, with this kernel, returns a smooth probability function by construction. The simple closed form of the estimator then allows, for example, derivatives of the distribution to be easily calculated. Second, the method removes any reliance on selection of bin centres or how many bins to use, which would affect the output of the histogram method.[3]

A comparison of the methods can be seen in Figure 2.4. Unless stated otherwise, any distributions shown will either be the result of thousands of runs with 10 pools of data per run and a histogram created (H) or thousands of runs with one pool of data per run and a KD estimate created (KD). Which method is being used will be indicated where appropriate. Note that the underlying simulations are identical and this choice of method only reflects two ways of visualising the data.

3.3 Errors

With such a wealth of data available we are in an excellent position to present some analysis of the errors involved. Error calculation depends on which method is being employed.

[3] For example, if we have very few bins we wouldn't expect to be able to build up a reasonable distribution.

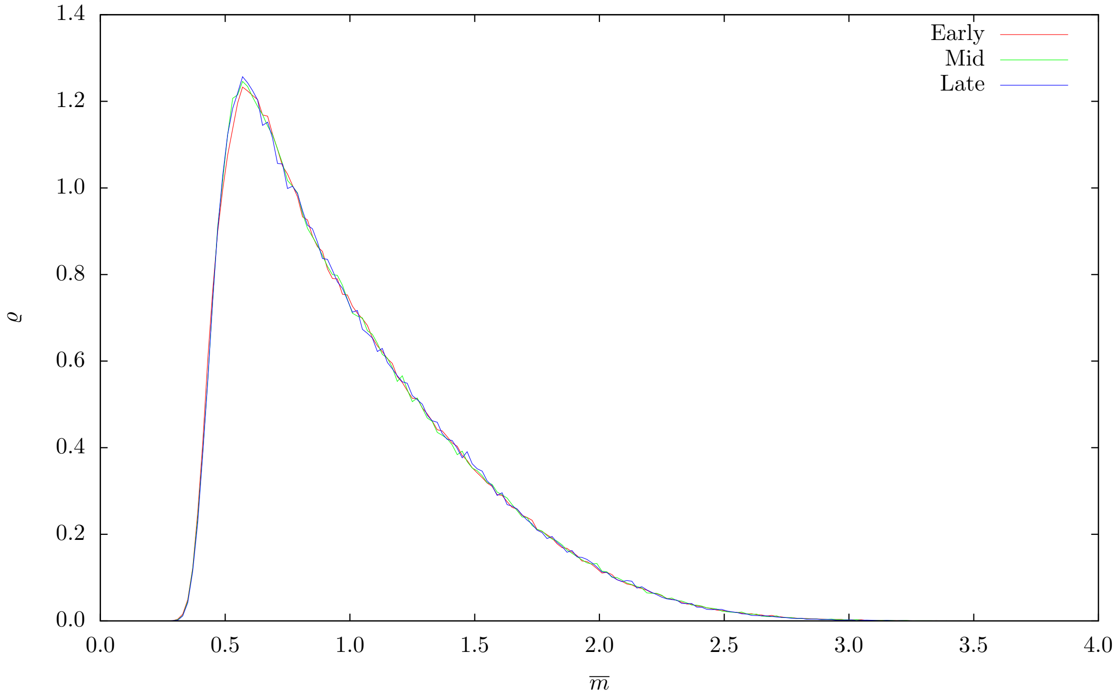


Figure 2.3 – (H) Mass distribution for $p = 1$ at three distinct times within the scaling region, indicated in Figure 2.2, clearly depicting the dynamic scaling nature of the system.

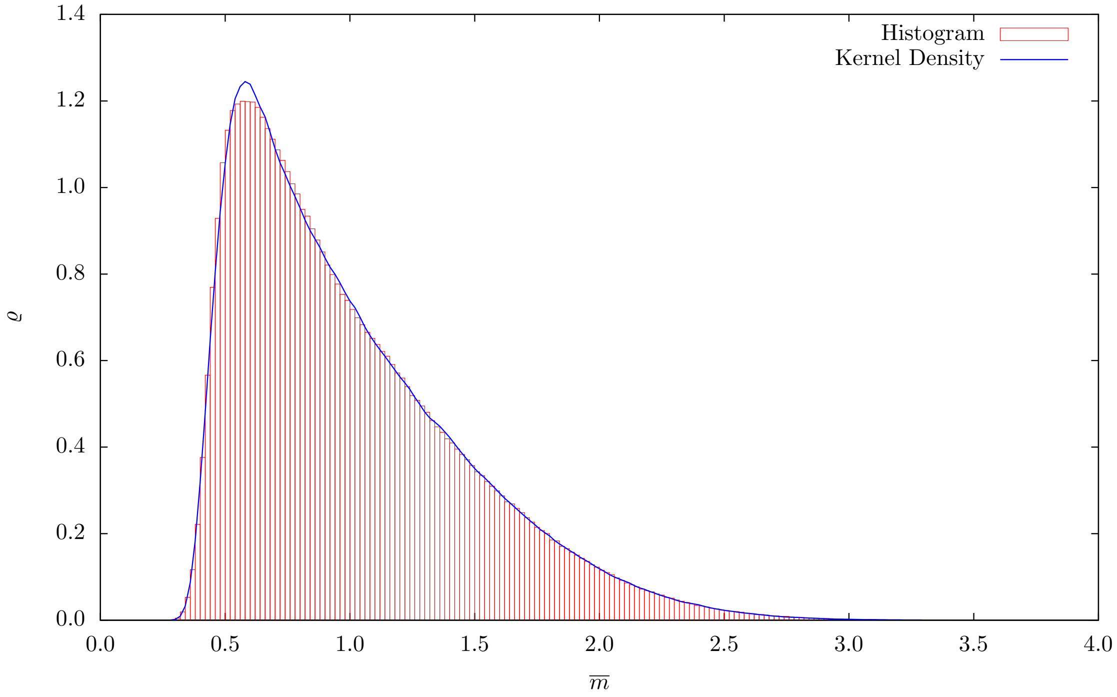


Figure 2.4 – Histogram method vs. Kernel Density method for $p = 1$ showing a very good match. Note that the KD method has a larger peak but qualitatively they are the same.

3.3.1 Histogram Method

As data is gathered the program calculates the *variance*, Var_N, and *standard error*, SE_N, in the data,

$$Var_N = \frac{1}{N}\sum_{i=1}^{N}(X_i - \overline{X}_N)^2 \qquad SE_N = \sqrt{\frac{Var_n}{N}}, \tag{3.3.1}$$

where X_i indicates each value used in deriving the mean, $\overline{X}_N$. From this we can indicate confidence intervals in our mean values, 95% confidence intervals arise by taking $1.96 \times SE_N$. Typical confidence intervals are shown in Figure 2.5 where we see strong evidence that our mean value is accurate.

3.3.2 Kernel Density (KD) Method

To calculate the standard error in the KD method we use the following formula,

$$SE_N(x) = \frac{1}{\sqrt{N}}\sqrt{\frac{1}{N-1}\sum_{i=1}^{N}\left[\frac{1}{h}K\left(\frac{\overline{m}-\overline{m}_i}{h}\right) - \varrho(\overline{m};\overline{m}_i,h)\right]^2}, \tag{3.3.2}$$

$$= \frac{1}{\sqrt{N-1}}\sqrt{\frac{1}{h}\left(\frac{1}{Nh}\sum_{i=1}^{N}\left[K\left(\frac{\overline{m}-\overline{m}_i}{h}\right)\right]^2\right) - \varrho(\overline{m};\overline{m}_i,h)^2}. \tag{3.3.3}$$

Computationally we calculate the following two quantities,

$$\varrho(\overline{m};\overline{m}_i,h) = \frac{1}{Nh}\sum_{i=1}^{N}K\left(\frac{\overline{m}-\overline{m}_i}{h}\right),$$

which is simply the estimated value at $\overline{m}$, as before, and,

$$\psi(\overline{m};\overline{m}_i,h) = \frac{1}{Nh}\sum_{i=1}^{N}\left[K\left(\frac{\overline{m}-\overline{m}_i}{h}\right)\right]^2.$$

The standard error is then,

$$\frac{1}{\sqrt{N-1}}\sqrt{\frac{\psi(\overline{m};\overline{m}_i,h)}{h} - \varrho(\overline{m};\overline{m}_i,h)^2}. \tag{3.3.4}$$

We can then find 95% confidence intervals in our expected values by plotting $1.96 \times SE_N(x)$ above and below the estimated distribution $\varrho(\overline{m};\overline{m}_i,h)$. Where appropriate we will plot the following three curves: the estimated distribution $\varrho(\overline{m};\overline{m}_i,h)$ and the error envelope curves $\varrho(\overline{m};\overline{m}_i,h) + (1.96 \times SE_N(x))$ and $\varrho(\overline{m};\overline{m}_i,h) - (1.96 \times SE_N(x))$.

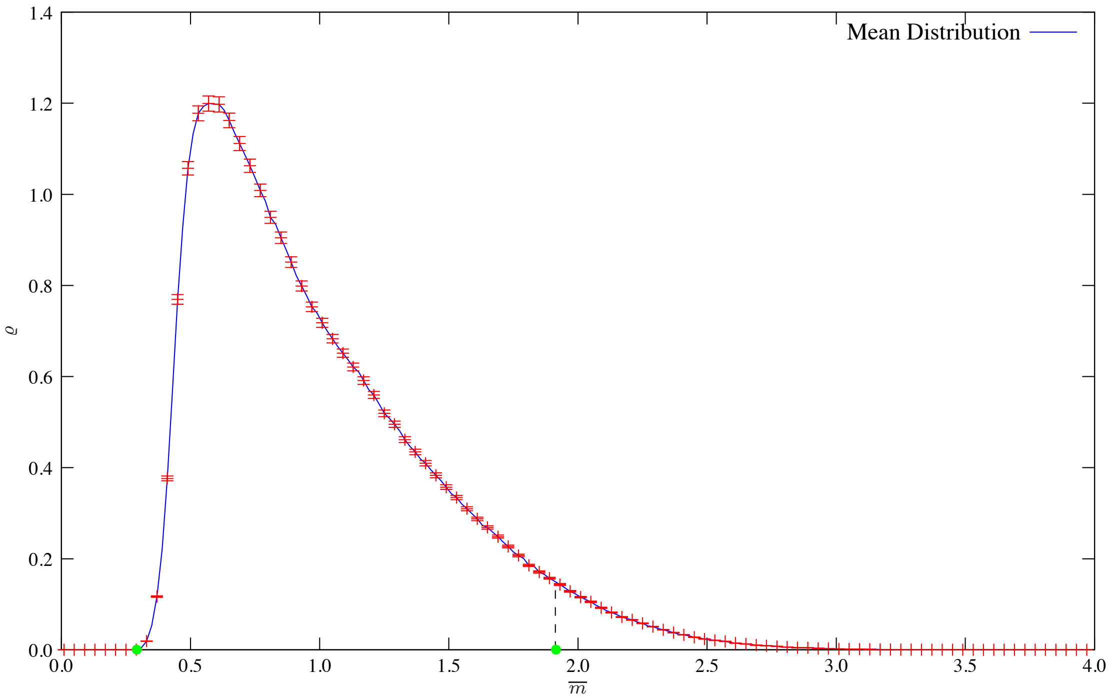


Figure 2.5 – (H) Mass distribution in scaling regime for $p = 1$ with 95% confidence intervals. The green dots indicate features to be discussed later; on the left the end of the *vacuum interval* and on the right the *5% cut-off value*.

3.4 Vacuum Interval

The *vacuum interval* is the region on the $\overline{m}$ axis between 0 and the first point where we have recorded some mass, shown in Figure 2.5. Physically this represents the fact that the smallest masses move the fastest and so are most likely to be involved in collisions early on in the simulations. By the time we reach the scaling state the very small masses have been involved in collisions leading to the emergence of the vacuum interval. The scaling of the distribution, and in particular the vacuum interval, does lead us to one interesting conclusion. As the system evolves and collisions take place the average mass increases, however, the ratio of the smallest mass in the system to the average mass remains constant. In other words the length of the vacuum interval remains constant. This is interesting because it confirms the assertion that the smallest particles are the most likely to collide and therefore be removed from the system. If this weren't true we'd expect to see some particles of a small mass remain at later times. We also note that, unlike some other systems, there is no mechanism by which a small particle can be created. We expect the smallest masses to be more likely to be involved in collisions and there is no way to create new small particles, these two facts go some way to understanding the emergence of the vacuum interval.

One interesting piece of information we will study later is the size of the vacuum interval as we change the velocity exponent p. As we shall see it does not remain at a fixed length and, in fact, increases in size as p is increased.

3.5 Support of the Distribution

At a glance the distribution appears to have finite support, reaching a zero value somewhere in the region of 3. It is, however, difficult to accurately obtain the exact value. Instead we probe the value which splits the distribution in two, with 95% to the left and 5% to the right, since numerically this is easier, and more accurate[4], to obtain. We will call this point the *5% cut-off*.

As with the vacuum interval this value does not remain fixed as we increase the velocity exponent p. As we increase p we find that the 5% cut-off decreases in value.

[4] The estimated distribution function becomes noisy in the tail making it hard to estimate a true value for finite support, if indeed it does have finite support. This noise may, in fact, be indicative of semi-infinite support.

4 Increasing p ($p \geqslant 1$) and Correlations

In this section we present some data and analysis for the distributions associated with higher p values. As we increase the velocity exponent p several changes in the distribution become apparent. Two have already been mentioned earlier, namely the increase in the size of the vacuum interval and the decrease of the 5% cut-off. A third consequence of increasing p is the emergence of a 'bump' on the tail of the distribution, this feature is not at all apparent in the case $p = 1$ above.

Data acquisition was carried out using the same two methods from the $p = 1$ case, namely the *histogram* (H) and *kernel density* (KD) methods. Figures will indicate, where appropriate, which method is being used.

4.1 Vacuum Interval

In Figure 2.6 we see that the vacuum interval persists as p is increased but that its extent is not fixed, in fact its size increases as p increases. We can see, however, that as p is increased the size doesn't increase indefinitely but appears to converge on some value in the region of 0.55. Evidence for this can be seen in Figure 2.7 where we have plotted the size of the vacuum interval against $1/p$. Asymptotically we find a value of 0.557397 as $p \to \infty$. The peak of the distribution doesn't appear to alter its position significantly as p is increased and as a result the left end of the distribution appears to get closer to a sharp jump in value. Compare this to the case when $p = 1$ where the distribution rises steeply but not in the sharp manner seen for higher p values.

4.2 Support of the Distribution

The tail of the distribution also varies as p increases with more of the distribution shifted to the left resulting, along with the change in the vacuum interval, in the increase in the size of the peak. This effect can be clearly seen by the decrease in the value of the distribution in the tail section. The result is that the 5% cut-off drifts to the left as p increases, however, as is the case for the vacuum interval, the point appears to converge in on some value rather than varying indefinitely, see Figure 2.7.

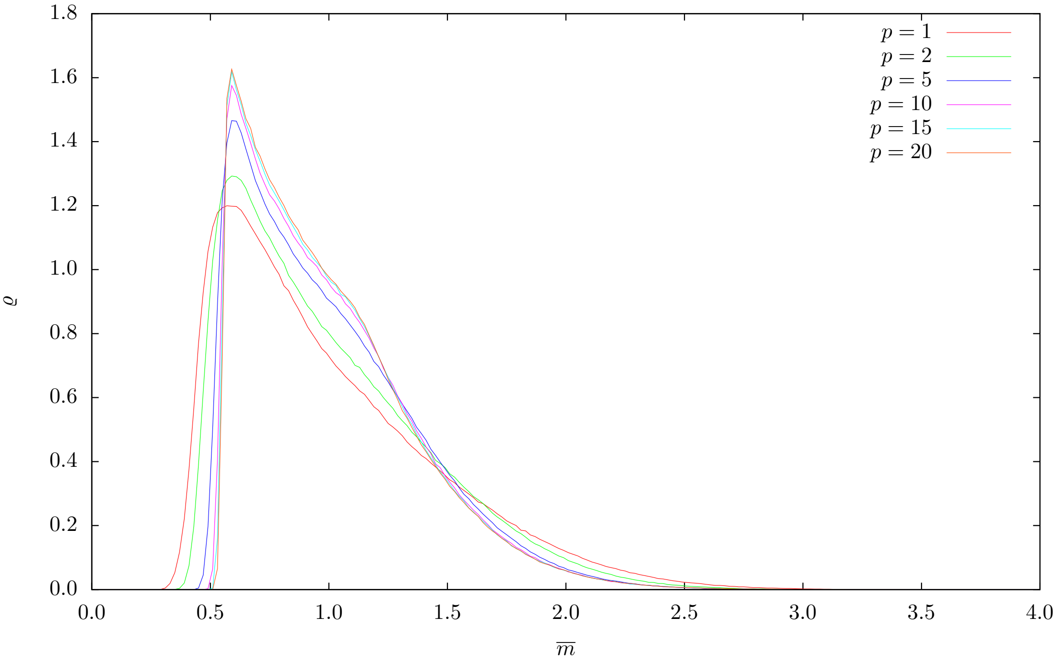


Figure 2.6 – (H) The distribution function for several values of p. We note three important features that are apparent as p increases: The increase in size of the vacuum interval, the reduction in the value of the 5% cut-off and the emergence of a ‘bump’ on the tail of the distribution.

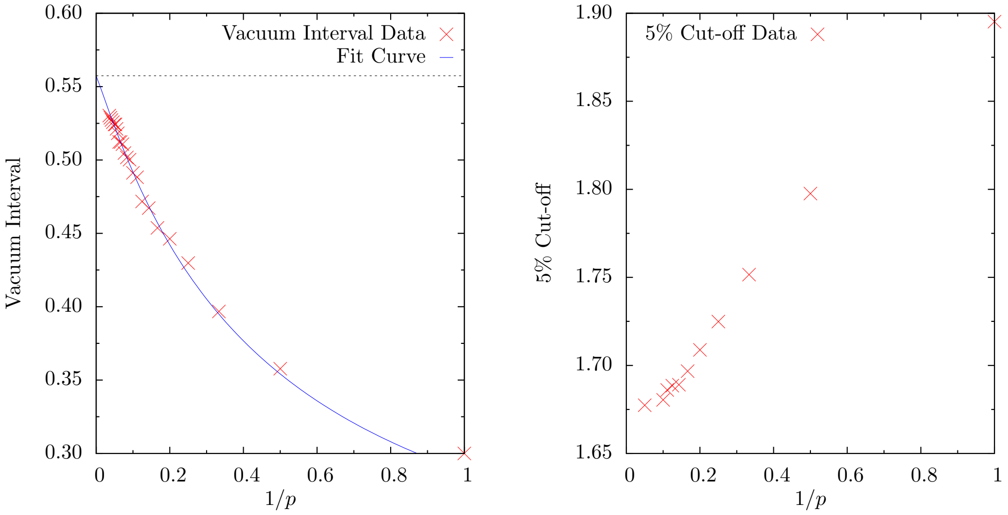


Figure 2.7 – Vacuum interval and 5% cut-off values for various p values together with the associated fit curve of the type $f(x) = -1/(x+a)^b + c$, for the vacuum interval. Data in the distribution tail became noisy for p values higher than 10 and so we found inconsistent results when trying to fit a curve to the data. The data seems to suggest an asymptotic value in the region of 1.65 for the cut-off value. The black dotted line indicates the asymptotic value at infinity of 0.557397 for the vacuum interval.

4.3 'Bump' on Distribution Tail

Perhaps most surprising is the emergence of a 'bump' on the tail of the distribution. This feature is not at all apparent for low p values but becomes quite visible as p is increased. Beyond $p = 10$ we see a clear bump at around twice the value of the minimum/peak of the distribution. This is quite an interesting and unusual feature of the distribution.

One possible explanation is that there is no mechanism in the model by which a particle of mass less than twice the minimum (relative to the average) can be created. Whilst not physically realistic, since they would have the same velocity, two particles of minimum size could combine to form one particle of twice that value. Therefore the 'bump' appears to be related to the difference in dynamics for different sections of the distribution. Above twice the minimum value there is an ability within the system to create a particle of that size. Below that value there is no mechanism for creation so it seems reasonable that this part of the distribution is subject to a different processes during evolution.

4.4 Correlations

We consider the correlations within the system for various values of p. By this we mean that we look to see if there is any general relationship between the masses, for example, is a 'large' mass generally followed by a much smaller one? To do this we make use of the standard *Pearson product-moment correlation coefficient*, $r_{X,Y}$, defined as

$$r_{X,Y} = \frac{\mathbb{E}(XY) - \mathbb{E}(X)\mathbb{E}(Y)}{\sqrt{\mathbb{E}(X^2) - \mathbb{E}(X)^2}\sqrt{\mathbb{E}(Y^2) - \mathbb{E}(Y)^2}},$$

which allows us to probe any linear relationships between random variables X and Y; e.g. $r_{X,Y} = 1$ indicates a perfect linear relationship. The general form simplifies greatly here since we will always be considering the correlation of variables which have the same individual expected value, hence,

$$r_{X,Y} = \frac{\mathbb{E}(XY) - \mathbb{E}(X)^2}{\mathbb{E}(X^2) - \mathbb{E}(X)^2}.$$

In our case we can replace X and Y by appropriate masses and this is then simply written,

$$r_{\mathfrak{m}_i,\mathfrak{m}_{i+k}}(k) = \frac{\mathbb{E}(\mathfrak{m}_i\mathfrak{m}_{i+k}) - \mathbb{E}(\mathfrak{m}_i)^2}{\mathbb{E}(\mathfrak{m}_i^2) - \mathbb{E}(\mathfrak{m}_i)^2},$$

where, i, as before, is an indexing of the particles, and k is an integer indicating which mass away from the current particle we are checking the correlation for, e.g. $k = 1$ are the right neighbours to the $\mathfrak{m}_i$. All of the expected values can then be replaced by appropriate averages since there is no weighting of a particular outcome and so the calculation of these values is then straightforward. We then take an average of all these values for each k to gauge the overall correlations for the full, infinite system. Note that we need only consider right hand neighbours to each particle since a simple reindexing of terms shows that the left correlations are identical. Figure 2.8 shows the results of these calculations. Here we see that there appears to be little or no correlation for even one particle away. This suggests that there is no 'ordering' to the system in the sense mentioned earlier. If we have a particularly large particle, for example, there is nothing to suggest that its neighbour is likely to be of any particular size.

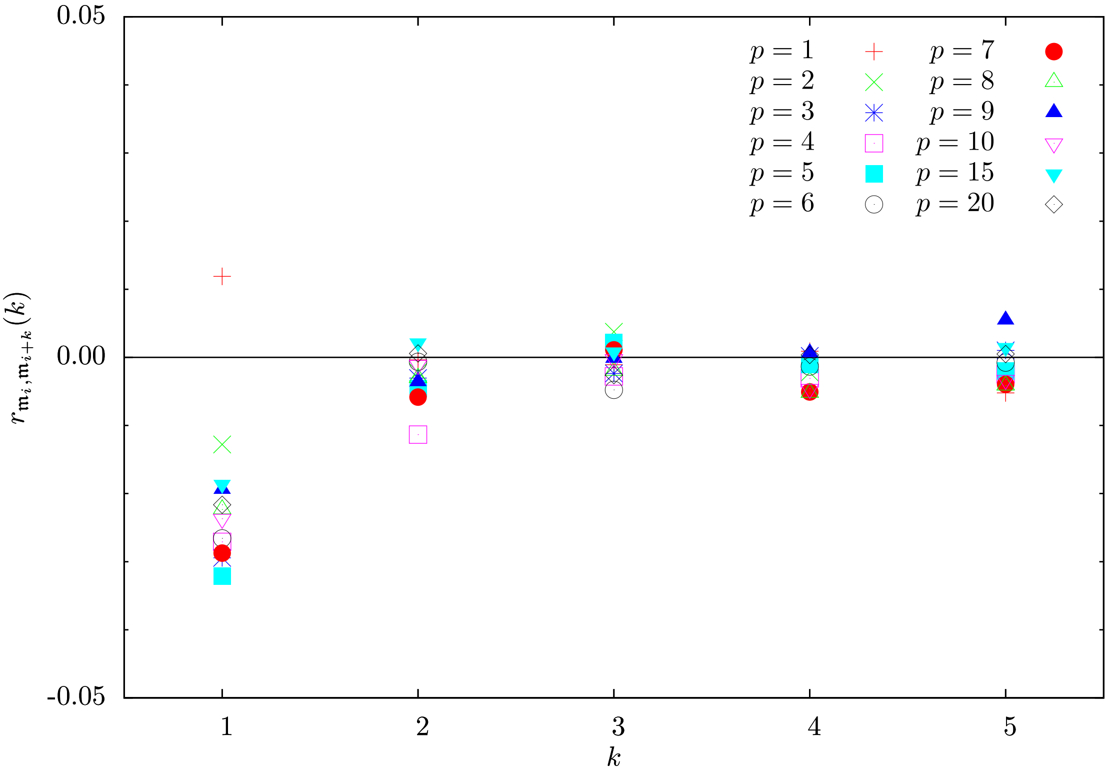


Figure 2.8 – Correlation coefficients for pairs of particles for a range of distances apart for the $\mathcal{ABP}$ model for various values of p. Note that for all values of p the correlation value at 0 is exactly 1, and as such has been omitted from the figure for clarity of the remaining values. Even for one particle away we see that there is very little correlation between the mass of the particles for all values of p.

5 System Dynamics for Large p and '$p \to \infty$'

We have detailed some of the characteristics associated with an increasing p value. We now carefully consider what happens to the system as p is increased. We show first that if p is large enough then the minimum mass particle is the next to collide, regardless of the distance to its neighbour or the difference in size of the neighbouring particles. Second, we show that the other particles only move by a very small amount in the time taken for this collision to take place. Effectively we show that once p is large enough the system behaves in a particular way regardless of the distances or size differences involved.

In Figure 2.6 we see that even for $p \geqslant 10$ the distributions are very similar. This provides some evidence that for large enough p we expect the system to behave in a very similar manner regardless of the exact value of p.

5.1 A 'Min-Driven' System

For all values of p we can easily calculate the time taken for a pair of particles to collide, subject to the fact the the mass to the right is larger than the mass to the left. Suppose we consider the particle $\mathfrak{m}_i$ and its neighbour $\mathfrak{m}_{i+1}$ a distance $\mathfrak{l}_i$ away, then the time for these to collide, t_i, can easily be calculated as follows,

$$\begin{aligned} t_i &= \frac{\mathfrak{l}_i}{\mathcal{V}(\mathfrak{m}_i) - \mathcal{V}(\mathfrak{m}_{i+1})}, \\ &= \frac{\mathfrak{l}_i}{\dfrac{1}{\mathfrak{m}_i^p} - \dfrac{1}{\mathfrak{m}_{i+1}^p}}, \\ &= \frac{\mathfrak{l}_i \mathfrak{m}_i^p \mathfrak{m}_{i+1}^p}{\mathfrak{m}_{i+1}^p - \mathfrak{m}_i^p}. \end{aligned} \tag{5.1.1}$$

From this we can clearly see, since we insisted that $\mathfrak{m}_{i+1} > \mathfrak{m}_i$, that t_i is some positive time depending on the masses, $\mathfrak{m}_i$ and $\mathfrak{m}_{i+1}$, the distance between the masses, $\mathfrak{l}_i$, and the velocity exponent p.

Now consider the mass $\mathfrak{m}_{i+1}$. Since we have insisted it is larger than $\mathfrak{m}_i$ we must have that $\mathfrak{m}_{i+1} = (1+\lambda_i)\mathfrak{m}_i$ for some $\lambda_i > 0$. Inserting this in (5.1.1) we have,

$$\begin{aligned}
t_i &= \frac{\mathfrak{l}_i \mathfrak{m}_i^p ((1+\lambda_i)\mathfrak{m}_i)^p}{((1+\lambda_i)\mathfrak{m}_i)^p - \mathfrak{m}_i^p}, \\
&= \frac{\mathfrak{l}_i \mathfrak{m}_i^{2p} (1+\lambda_i)^p}{(1+\lambda_i)^p \mathfrak{m}_i^p - \mathfrak{m}_i^p}, \\
&= \frac{\mathfrak{l}_i \mathfrak{m}_i^{2p} (1+\lambda_i)^p}{((1+\lambda_i)^p - 1)\mathfrak{m}_i^p}, \\
&= \frac{\mathfrak{l}_i \mathfrak{m}_i^{p} (1+\lambda_i)^p}{(1+\lambda_i)^p - 1} =: \mathfrak{l}_i \mathfrak{m}_i^p \Lambda(p, \lambda_i),
\end{aligned} \tag{5.1.2}$$

where $\Lambda(p,\lambda_i) := (1+\lambda_i)^p/((1+\lambda_i)^p - 1)$.

This expression (5.1.2) is valid for all values of p so we consider $p = 1$ as an illustrative example of the different effects that the terms on the right hand side can have. For $p = 1$ we have,

$$t_i = \frac{\mathfrak{l}_i \mathfrak{m}_i (1+\lambda_i)}{\lambda_i}.$$

Now, it is clear that if either of $\mathfrak{l}_i$ or $\mathfrak{m}_i$ is very small then the time will correspondingly tend to be very small. Alternatively, if λ_i is very small then this will tend to make the time very large. From this we see that all three factors play a role in determining the time t_i and that none can be disregarded as irrelevant.

We now consider the case for much larger p values ($p \gg 1$). First let us consider the term $\Lambda(p, \lambda_i)$ and examine its behaviour as p gets large. We have $(1+\lambda_i) > 1$ and it is also clear that the numerator is always larger than the denominator for all p and so,

$$\Lambda(p, \lambda_i) > 1, \quad \forall p > 0.$$

Formally it is easy to see that the limit as $p \to \infty$ is equal to 1,

$$\lim_{p\to\infty} \Lambda(p,\lambda_i) = \lim_{p\to\infty} \frac{(1+\lambda_i)^p}{(1+\lambda_i)^p - 1} = 1.$$

Since $\Lambda(p, \lambda_i) > 1$ and the limit as $p \to \infty$ is equal to 1 it seems reasonable that for finite p we can make $\Lambda(p, \lambda_i)$ arbitrarily close to 1 so long as we choose p large enough. Suppose that $\varepsilon_1 > 0$ and p is fixed, then it is clear that for some ε_1 we have,

$$\Lambda(p,\lambda_i) = \frac{(1+\lambda_i)^p}{(1+\lambda_i)^p - 1} < 1 + \varepsilon_1,$$

or equivalently,

$$\Lambda(p,\lambda_i) - 1 = \frac{(1+\lambda_i)^p}{(1+\lambda_i)^p - 1} - 1 = \frac{1}{(1+\lambda_i)^p - 1} < \varepsilon_1.$$

In other words, since $\Lambda(p,\lambda_i)$ is finite for finite p, we can always choose some $\varepsilon_1 > 0$ so that the inequality holds. Rearranging the inequality as follows yields a restriction on p that must be satisfied in order for the inequality to hold true for completely arbitrary ε_1.

$$\begin{aligned}
\frac{1}{(1+\lambda_i)^p - 1} &< \varepsilon_1, \\
1 &< \varepsilon_1((1+\lambda_i)^p - 1), \\
\frac{1+\varepsilon_1}{\varepsilon_1} &< (1+\lambda_i)^p, \\
\ln\left(\frac{1+\varepsilon_1}{\varepsilon_1}\right) &< p\ln(1+\lambda_i), \\
\mu_i(\varepsilon_1) := \frac{\ln\left(\frac{1+\varepsilon_1}{\varepsilon_1}\right)}{\ln(1+\lambda_i)} &< p. \qquad (5.1.3)
\end{aligned}$$

What we have shown, provided p is large enough to satisfy (5.1.3), is that the time for two particles to collide can be approximated by $t_i = \mathfrak{l}_i\mathfrak{m}_i^p$, to arbitrary precision. This is achieved by ensuring that the missing factor in the time calculation, namely $\Lambda(p,\lambda_i)$, is arbitrarily close to 1. If we let $\mu(\varepsilon_1)$ be the maximum of all the $\mu_i(\varepsilon_1)$ and ensure that $p > \mu(\varepsilon_1)$ then we can guarantee this arbitrary precision for all the pairwise calculations in the system.

So far we have shown that the time to collision between two particles can be approximated to arbitrary precision by $t_i = \mathfrak{l}_i\mathfrak{m}_i^p$, however, it remains to be shown that the minimum particle does indeed collide next. Even with this simplification, if one of the $\mathfrak{l}_i$ is exceptionally small it could result in a smaller collision time for that pair of particles than the minimum mass collision time.

Let $\mathfrak{m}^\star$ be the minimum mass in the system, assumed, for now, to be unique. We call $t^\star$ the time for the minimum to collide and $\mathfrak{l}^\star$ the distance from the minimum mass to its right neighbour. We want to guarantee that the collision time for this particle is strictly smaller than the collision time for any other. We have assumed that there is a unique minimum mass and so,

$$\mathfrak{m}^\star < \mathfrak{m}_\alpha \leqslant \mathfrak{m}_\beta \leqslant \dots,$$

where the $\mathfrak{m}_\bullet \neq \mathfrak{m}^\star$ are the other masses which are smaller than their right neighbour in the system and hence have valid collision times. As an immediate consequence we have,

$$(\mathfrak{m}^\star)^p < \mathfrak{m}_\alpha^p \leqslant \mathfrak{m}_\beta^p \leqslant \dots$$

and multiplying through by $\mathfrak{l}^\star$ we have,

$$\mathfrak{l}^\star(\mathfrak{m}^\star)^p < \mathfrak{l}^\star\mathfrak{m}_\alpha^p \leqslant \mathfrak{l}^\star\mathfrak{m}_\beta^p \leqslant \dots$$

For each of the other masses in the system we have then one of three possible scenarios (we use α as an example),

i) $\mathfrak{l}^\star = \mathfrak{l}_\alpha$: Then immediately $\mathfrak{l}^\star(\mathfrak{m}^\star)^p < \mathfrak{l}_\alpha \mathfrak{m}_\alpha^p$,

ii) $\mathfrak{l}^\star < \mathfrak{l}_\alpha$: Then $\mathfrak{l}^\star(\mathfrak{m}^\star)^p < \mathfrak{l}^\star \mathfrak{m}_\alpha^p < \mathfrak{l}_\alpha \mathfrak{m}_\alpha^p$, or

iii) $\mathfrak{l}^\star > \mathfrak{l}_\alpha$.

In the third case the result cannot be immediately read off. In fact, for relatively low p values, it is this possibility which allows particles other than the minimum mass to collide next. As already mentioned, if the distance is very small, the size of the mass may be irrelevant in determining the next collision.

We want the following condition to be true,

$$\mathfrak{l}^\star(\mathfrak{m}^\star)^p < \mathfrak{l}_\alpha \mathfrak{m}_\alpha^p,$$

or equivalently

$$\frac{(\mathfrak{m}^\star)^p}{\mathfrak{m}_\alpha^p} < \frac{\mathfrak{l}_\alpha}{\mathfrak{l}^\star}\,(<1)\,.$$

From this inequality we can derive a condition on p which will ensure that the condition is true,

$$\begin{aligned} \frac{(\mathfrak{m}^\star)^p}{\mathfrak{m}_\alpha^p} &< \frac{\mathfrak{l}_\alpha}{\mathfrak{l}^\star}, \\ p\ln\left(\frac{\mathfrak{m}^\star}{\mathfrak{m}_\alpha}\right) &< \ln\left(\frac{\mathfrak{l}_\alpha}{\mathfrak{l}^\star}\right), \\ p &> \frac{\ln\left(\frac{\mathfrak{l}_\alpha}{\mathfrak{l}^\star}\right)}{\ln\left(\frac{\mathfrak{m}^\star}{\mathfrak{m}_\alpha}\right)} =: \nu_\alpha. \end{aligned} \tag{5.1.4}$$

As a result as long as $p > \nu_\alpha$ we can guarantee that $\mathfrak{l}^\star(\mathfrak{m}^\star)^p < \mathfrak{l}_\alpha \mathfrak{m}_\alpha^p$ is true, as required. If we let ν be the maximum of all the $\nu_\bullet$, then as long as $p > \nu$ we can guarantee that $t^\star$ is indeed the minimum time to collision, in other words the minimum mass collides next. Note that a very similar argument can be made to ensure that all the inequalities are strict, which in turn guarantees that collisions are binary and not otherwise.

In summary, provided $p > \max\{\mu(\varepsilon_1), \nu\}$ we can guarantee that the minimum mass particle collides next.

5.2 Minimal movement of particles $\mathfrak{m}_i \neq \mathfrak{m}^\star$

Each of the other particles in the system will move by some distance in the time taken for the minimum to collide. We show that for sufficiently large p that this distance can be arbitrarily small.

The distance any other particle will travel, $\mathfrak{l}_i$, will be equal to $[\mathcal{V}(\mathfrak{m}_i)]t$ and we know the amount of time elapsed, namely $\mathfrak{l}^\star(\mathfrak{m}^\star)^p$, so we have,

$$\mathfrak{l}_i = [\mathcal{V}(\mathfrak{m}_i)]\mathfrak{l}^\star(\mathfrak{m}^\star)^p = \frac{1}{\mathfrak{m}_i^p}\mathfrak{l}^\star(\mathfrak{m}^\star)^p = \mathfrak{l}^\star\left(\frac{\mathfrak{m}^\star}{\mathfrak{m}_i}\right)^p.$$

Note that since $\mathfrak{m}^\star$ is the minimum mass $\mathfrak{m}^\star/\mathfrak{m}_i < 1$. Suppose that $\varepsilon_2 > 0$ and that the following holds,

$$\mathfrak{l}^\star\left(\frac{\mathfrak{m}^\star}{\mathfrak{m}_i}\right)^p < \varepsilon_2.$$

Rearranging this condition leads us to the following,

$$\begin{aligned}
\mathfrak{l}^\star\left(\frac{\mathfrak{m}^\star}{\mathfrak{m}_i}\right)^p &< \varepsilon_2,\\
\left(\frac{\mathfrak{m}^\star}{\mathfrak{m}_i}\right)^p &< \frac{\varepsilon_2}{\mathfrak{l}^\star},\\
p\ln\left(\frac{\mathfrak{m}^\star}{\mathfrak{m}_i}\right) &< \ln\left(\frac{\varepsilon_2}{\mathfrak{l}^\star}\right),\\
p &> \frac{\ln\left(\frac{\varepsilon_2}{\mathfrak{l}^\star}\right)}{\ln\left(\frac{\mathfrak{m}^\star}{\mathfrak{m}_i}\right)} =: \sigma_i(\varepsilon_2).
\end{aligned} \tag{5.2.1}$$

In other words, as long as $p > \sigma_i(\varepsilon_2)$ we can guarantee that the particle $\mathfrak{m}_i$ moves by a distance less than ε_2. The denominator in (5.2.1) is always negative whilst the numerator depends on the size of $\varepsilon_2/\mathfrak{l}^\star$. Consider the situation where ε_2 is very small, then the numerator will be large and negative. We see, however, that this condition also allows us to guarantee that a particle will move by less than any distance we specify so long as p is large enough. Let $\sigma(\varepsilon_2)$ be the maximum of all the $\sigma_i(\varepsilon_2)$, then if we specify some value for ε_2 we can be assured that all the particles larger than the minimum move less than the distance ε_2 provided $p > \sigma(\varepsilon_2)$. Note also that the distance the minimum moves, $\mathfrak{l}$, is,

$$\mathfrak{l} = \frac{1}{(\mathfrak{m}^\star)^p}d^\star(\mathfrak{m}^\star)^p = \mathfrak{l}^\star,$$

for all values of p as expected. In summary, provided $p > \sigma(\varepsilon_2)$ the other particles move by an arbitrarily small amount, namely less than some value ε_2, whilst the minimum mass particle collides.

Simulations of this system for various values of p suggest that the minimum mass is almost always unique, however, there is no theoretical underpinning to this, so we need to make a slight adjustment to our argument for the case when the minimum is not unique.

Suppose that the minimum is not unique and there are, in fact, two masses of minimum size. This argument can be easily extended to more than two masses. Each has mass $\mathfrak{m}^\star$ but it is likely that the distances will be different. If the distances are also the same then the collisions will both have the same estimated time, by the argument above, and so there is no issue in assuming they both hit at the same time. Suppose the distances to collision are $\hat{l}$ and $\mathfrak{l}^\star$ where $\hat{l} < \mathfrak{l}^\star$, then the times to collision will satisfy,

$$\hat{l}(\mathfrak{m}^\star)^p < \mathfrak{l}^\star(\mathfrak{m}^\star)^p.$$

If p is large enough so that both times are strictly smaller than all the remaining masses in the system we can simply evolve the system by the time $\hat{l}(\mathfrak{m}^\star)^p$ which leaves the other minimum mass particle, now unique, a distance $\mathfrak{l}^\star - \hat{l}$ away from its neighbour. Since it is now the unique mass it follows the pattern illustrated earlier and collides next. In this way we can sequentially move all the minimum particles whilst the others all remain essentially fixed over this time scale. As a result we can guarantee that all the minimum particles collide whilst the others barely move at all.

The only other possible special case is when two particles of minimum size are next to each other. In this case we simply move the right most particle in the sequence of two or more until it collides, followed by the others in the same fashion until all the minimum particles are gone and the process repeats. It should be stressed, however, that numerical simulations of the system from various initial conditions rarely lead to either of these outcomes, in other words the minimum mass is, much more often than not, unique.

Combining these two arguments together we reach the following conclusion. If the following condition on p holds, namely $p > \max\{\mu(\varepsilon_1), \nu, \sigma(\varepsilon_2)\}$, then we can guarantee two things about the evolution of the system. First, the minimum mass particle(s) is the next to collide and second, during the time to this collision the other particles in the system move by an arbitrarily small amount.

6 One Sided 'Paste-All' Model

In this section we give some detail of another model, the so-called *'Paste-All'* model [**30**], and in the next section draw comparisons between this model and the aggregating, ballistic particle, $\mathcal{ABP}$, models above.

6.1 Introduction to the 'Paste-All' Model

This model, originally proposed in a different context [**30**], consists of a random distribution of masses on the line. The evolution of the system proceeds in a straightforward manner, first we search for the smallest mass and then this mass is 'pasted' onto one of its neighbours, with equal probability, creating a larger particle. The number of particles in the system thus goes down and the system coarsens. If the minimum mass is not unique we simply perform the 'pasting' procedure on all of the minimum particles and *assume* that two minimum size particles are not pasted together or equivalently that two minimum sized particles are not adjacent.

It can be easily shown that there are only weak correlations within the system (See Figure 2.9), and so each time we paste a particle onto its neighbour we are simply making use of the probability distribution of the masses to find the likelihood of finding a particle of a particular size. Assuming that neighbours are unrelated and have no correlation in their size could be considered a type of *mean-field* assumption, however, since there are no correlations here we expect any predicted evolution of the distribution to be exact. This type of model is evidently of the probabilistic type.

We make one slight alteration to the model to fit better with what happens in the $\mathcal{ABP}$ model.

6.1.1 One Sided 'Paste-All' Model

The *one sided 'Paste-all' model* is identical to the 'Paste-all' model other than the fact we always choose to paste to the mass to the right of the minimum particle.

Interestingly, since there are no correlations in the system, we expect the one sided version to have the same distribution as the two sided version. This is because we only keep track of the distribution of masses, the lack of correlations means there is as much chance of having a particular mass to the right as there is to the left, and so we alter the distribution at each stage in exactly the same manner. In other words, since the mass

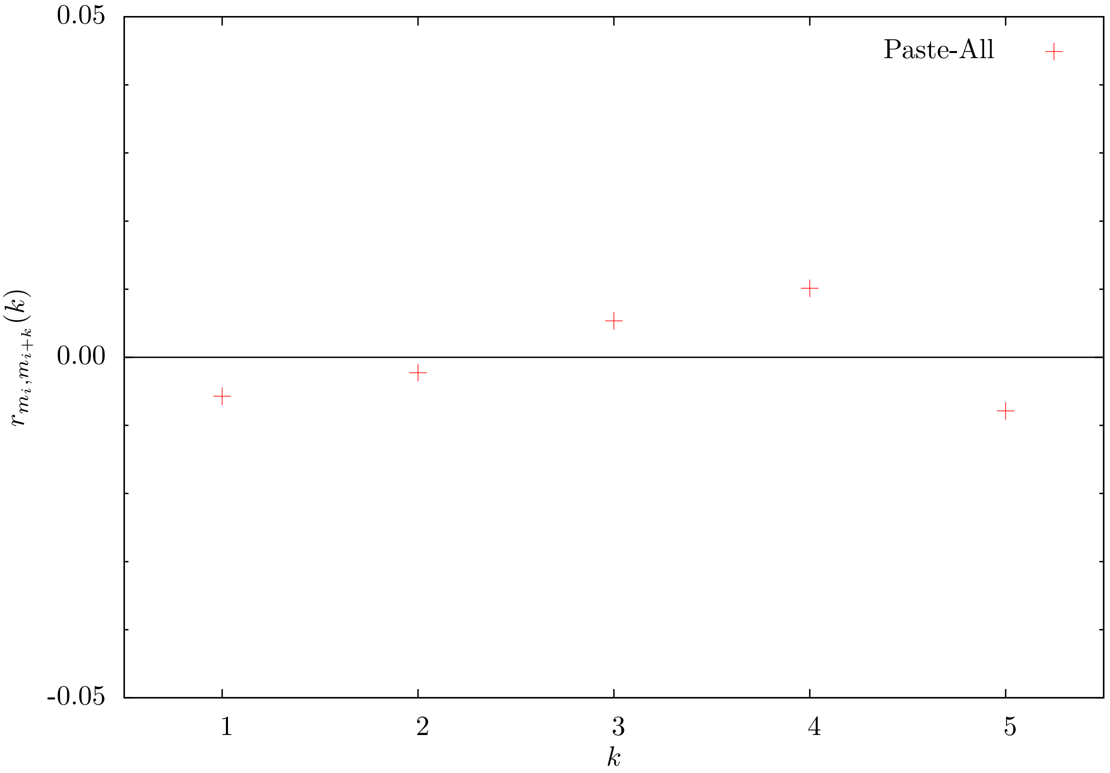


Figure 2.9 – Correlation coefficients for pairs of particles a range of distances, k, apart for the Paste-all model. Note that the correlation value at 0 is exactly 1, and as such has been omitted from the figure for clarity of the remaining values. We see that generally there is very little correlation in particle mass for even as little as one particle distance away.

to the left and right of the minimum are governed by the same distribution it makes no difference if we paste to the left or right. This likely wouldn't be the case if there were correlations and certainly wouldn't be if there were asymmetric correlations.

6.2 Discretely Distributed Mass - One Sided Paste-All Model

This model is a one sided 'Paste-all' model with each mass, m_i, taking some integer value, i.e. $m_i \in \mathbb{N}$, $\forall i$. As in the $\mathcal{ABP}$ model we have a finite number of particles and i is an index of the particles. The initial distribution of masses is fixed but we can easily use any distribution, e.g. uniform, Gaussian. Let $j \in \mathbb{N}$ be the number of iterations passed and we also therefore let 0 iterations passed be the initial condition. We can then define the following quantities,

$$\begin{aligned} n_j(m) \quad & \text{- The number of particles of mass } m \text{ after } j \text{ iterations,} \\ \mathcal{N}_j \quad & \text{- The total number of particles after } j \text{ iterations,} \end{aligned}$$

and hence $\mathcal{N}_0$ is the initial number of particles. We choose to consider the fraction of particles of a particular mass after j iterations rather than use the number directly. We denote this by $\rho_j(m)$ and it is defined as follows,

$$\rho_j(m) = \frac{n_j(m)}{\mathcal{N}_j}.$$

If we denote the minimum mass by $m^\star(j)$ then the system evolves in the manner described earlier. Take all the particles of minimum mass, $m^\star(j)$, and join them at random onto their right neighbours. This is equivalent, because of the lack of correlations, to taking all the minimum sized particles and picking another sized particle at random, subject to the distribution of masses, and summing their masses together. We assume this procedure is not done sequentially but synchronously, i.e. if there are 10 of minimum size we select 10 other particles from the remaining bunch and add the masses at once. Thus, at each iteration there is only one change in the distribution function.

Making use of the particle fractions it is easy to find the probability of a particular particle size being hit, denoted $\tilde{\rho}_j(m)$. This is simply the fraction of particles of a particular size, $\rho_j(m)$, modified to take account of the removal of the minimum mass particles, in other words divided by $1 - \rho_j(m^\star)$,

$$\tilde{\rho}_j(m) = \frac{\rho_j(m)}{1 - \rho_j(m^\star(j))}. \tag{6.2.1}$$

We also note that since there are no particles of mass less than the minimum that the following is immediately true,

$$\rho_j(m) = 0, \quad m < m^\star(j).$$

We now want to build a difference equation for the evolution of the system. This is achieved by simply considering how many particles of a particular size there are then adding in any that are created of that size and subtracting any that are lost from that size, all of which is subject to the probability of such a collision happening. We consider the fraction of a mass $m > m^\star$ after one more iteration, $\rho_{j+1}(m)$ and relate it back to the previous iteration as follows,

$$\rho_{j+1}(m) = \tilde{\rho}_j(m) + \rho_j(m^\star(j))\tilde{\rho}_j(m - m^\star(j)) - \rho_j(m^\star(j))\tilde{\rho}_j(m). \tag{6.2.2}$$

The first term on the right hand side is the adjusted fraction since we have removed the minimum masses. The second term is the creation of a particle of mass m by the collision of the minimum, $m^\star(j)$, and another particle of size $m - m^\star(j)$. The third is the loss of a particle of mass m by being hit by the minimum mass and thus increasing in size.

Due to the discrete nature of the mass distribution we can see that the minimum mass obeys the following,

$$m^\star(j+1) = m^\star(j) + 1.$$

We can also calculate the mean mass, $\langle m\rangle(j)$, of the system easily by making use of the particle fractions $\rho_j(m)$,

$$\langle m\rangle(j) = \sum_{m=1}^{\infty} m\rho_j(m). \tag{6.2.3}$$

Suppose that we now run simulations of this system and plot $\rho_j(m)$ against m; we will see an evolution of the distribution. If, however, we scale the plot by the mean mass, $\langle m\rangle(j)$, and instead plot $\langle m\rangle(j)\rho_j(m)$ against $m/\langle m\rangle(j)$ we see that the distribution tends to some fixed distribution as the iteration number is increased. A numerically generated example of this fixed distribution, labelled ϱ, can be seen in Figure 2.10 which was generated by averaging over five thousand independent simulations.

6.2.1 Self-Similar Solutions and the Delay Differential Equation

We begin by labelling the fixed distribution that emerges by $\hat{\rho}(x)$ where $x = m/\langle m\rangle(j)$ and we can clearly see that

$$\hat{\rho}(x) = 0, \quad x < x^\star, \tag{6.2.4}$$

where $x^\star = m^\star / \langle m\rangle(j)$. After we have performed many iterations we expect to be able to approximate the distribution $\rho_j(m)$ by this fixed distribution as follows,

$$\rho_j(m) = \frac{1}{\langle m\rangle(j)}\hat{\rho}\left(\frac{m}{\langle m\rangle(j)}\right). \tag{6.2.5}$$

We now derive two moment conditions on the fixed distribution $\hat{\rho}(x)$ from the discrete distribution $\rho_j(m)$. Consider that the sum of all the particle fractions is, by definition, equal to 1, we then derive the moment condition as follows,

$$\begin{aligned}
1 &= \sum_{m=1}^{\infty} \rho_j(m), \\
&= \sum_{m=1}^{\infty} \frac{1}{\langle m\rangle(j)}\hat{\rho}\left(\frac{m}{\langle m\rangle(j)}\right), \\
&= \sum_{m=1}^{\infty} h(j)\hat{\rho}\left(\frac{m}{\langle m\rangle(j)}\right), \quad \text{where} \quad h(j) = \frac{1}{\langle m\rangle(j)}.
\end{aligned}$$

From numerical simulations we see that $\langle m\rangle(j) \approx \alpha j$, for some constant α, and $m^\star(j) \approx j$ for large j. It is clear that $\langle m\rangle(j) \geqslant m^\star(j)$ and that as $j \to \infty$, $\langle m\rangle(j) \to \infty$ and $m^\star(j) \to \infty$. It immediately follows that $h(j) \to 0$ as $j \to \infty$ and so we have a Riemann sum for $\hat{\rho}(x)$. A similar procedure using (6.2.3) produces the second moment condition. We therefore have the following two moment conditions on $\hat{\rho}(x)$,

$$\int_0^\infty \hat{\rho}(x)\, dx = 1 \quad \text{and} \quad \int_0^\infty x\hat{\rho}(x)\, dx = 1. \tag{6.2.6}$$

We now reconsider the difference equation (6.2.2), substituting in for $\tilde{\rho}_j(m)$ and collecting like terms,

$$\begin{aligned}
\rho_{j+1}(m) &= \frac{\rho_j(m)}{1-\rho_j(m^\star(j))} + \frac{\rho_j(m^\star(j))\rho_j(m-m^\star(j))}{1-\rho_j(m^\star(j))} - \frac{\rho_j(m^\star(j))\rho_j(m)}{1-\rho_j(m^\star(j))}, \\
&= \frac{\rho_j(m)[1-\rho_j(m^\star(j))]}{1-\rho_j(m^\star(j))} + \frac{\rho_j(m^\star(j))\rho_j(m-m^\star(j))}{1-\rho_j(m^\star(j))}, \\
&= \rho_j(m) + \frac{\rho_j(m^\star(j))\rho_j(m-m^\star(j))}{1-\rho_j(m^\star(j))}. \qquad (6.2.7)
\end{aligned}$$

If we substitute the observations from numerical simulations into the distribution formula we have,

$$\begin{aligned}
\rho_j(m) &= \frac{1}{\langle m\rangle(j)}\hat{\rho}\left(\frac{m}{\langle m\rangle(j)}\right) = \frac{1}{\alpha j}\hat{\rho}\left(\frac{m}{\alpha j}\right), \qquad (6.2.8) \\
\rho_{j+1}(m) &= \frac{1}{\langle m\rangle(j+1)}\hat{\rho}\left(\frac{m}{\langle m\rangle(j+1)}\right) = \frac{1}{\alpha(j+1)}\hat{\rho}\left(\frac{m}{\alpha(j+1)}\right), \\
&= \frac{1}{\alpha j\left(1+\frac{1}{j}\right)}\hat{\rho}\left(\frac{m}{\alpha j\left(1+\frac{1}{j}\right)}\right). \qquad (6.2.9)
\end{aligned}$$

Now considering the difference $\rho_{j+1}(m) - \rho_j(m)$ and making use of three separate Taylor expansions we find the following approximation,

$$\begin{aligned}
\rho_{j+1}(m) - \rho_j(m) &= \frac{1}{\alpha j\left(1+\frac{1}{j}\right)}\hat{\rho}\left(\frac{m}{\alpha j\left(1+\frac{1}{j}\right)}\right) - \frac{1}{\alpha j}\hat{\rho}\left(\frac{m}{\alpha j}\right),\\
&\approx \frac{1}{\alpha j}\left(1-\frac{1}{j}\right)\hat{\rho}\left(\frac{m}{\alpha j}-\frac{m}{\alpha j^2}\right) - \frac{1}{\alpha j}\hat{\rho}\left(\frac{m}{\alpha j}\right),\\
&\approx \frac{1}{\alpha j}\left(1-\frac{1}{j}\right)\left(\hat{\rho}\left(\frac{m}{\alpha j}\right)-\frac{m}{\alpha j^2}\hat{\rho}'\left(\frac{m}{\alpha j}\right)\right) - \frac{1}{\alpha j}\hat{\rho}\left(\frac{m}{\alpha j}\right),\\
&\approx -\frac{1}{\alpha j^2}\hat{\rho}(x) - \frac{x}{\alpha j^2}\hat{\rho}'(x), \quad \text{where} \quad x = \frac{m}{\alpha j}. \qquad (6.2.10)
\end{aligned}$$

With this in mind we now return to (6.2.7) and substitute in for the difference above and the numerical approximations in the remaining terms,

$$\begin{aligned}
-\frac{1}{\alpha j^2}\hat{\rho}(x) - \frac{x}{\alpha j^2}\hat{\rho}'(x) &= \frac{\frac{1}{\alpha^2 j^2}\hat{\rho}(x^\star)\hat{\rho}\left(\frac{m}{\alpha j}-\frac{1}{\alpha}\right)}{1-\frac{1}{\alpha j}\hat{\rho}\left(\frac{1}{\alpha}\right)},\\
-\hat{\rho}(x) - x\hat{\rho}'(x) &= \frac{\frac{1}{\alpha}\hat{\rho}(x^\star)\hat{\rho}\left(\frac{m}{\alpha j}-\frac{1}{\alpha}\right)}{1-\frac{1}{\alpha j}\hat{\rho}\left(\frac{1}{\alpha}\right)},\\
-(x\hat{\rho}(x))' &= \frac{x^\star\hat{\rho}(x^\star)\hat{\rho}(x-x^\star)}{1-\frac{1}{\alpha j}\hat{\rho}(x^\star)}, \quad \text{since} \quad x^\star = \frac{1}{\alpha}. \qquad (6.2.11)
\end{aligned}$$

Since we expect this to only hold for large values of j we can write this to leading order as the delay differential equation,

$$\begin{gathered}
(x\hat{\rho}(x))' = -x^\star\hat{\rho}(x^\star)\hat{\rho}(x-x^\star), \quad x \geqslant x^\star, \qquad (6.2.12)\\
\hat{\rho}(x) = 0, \quad 0 \leqslant x < x^\star,\\
\int_0^\infty \hat{\rho}(x)\,dx = 1 \quad \text{and} \quad \int_0^\infty x\hat{\rho}(x)\,dx = 1.
\end{gathered}$$

6.3 Continuously Distributed Mass - One Sided Paste-All Model

This model is identical to the previous in all but one respect, instead of a discrete integer mass associated to each particle, we now have a continuous distribution of mass. For the purposes of simulation this simply means each particle has some positive mass distributed according to some chosen initial distribution, e.g. uniform, Gaussian. The iteration process is identical, namely, take the particle(s) of minimum size and 'paste' their mass(es) onto a random selection of the remaining particles. Again since there are no correlations in the model this is equivalent to picking a random particle with probability subject to the current probability distribution and joining their masses. The scaling distribution for this model found via numerical simulations, again labelled ϱ, can be seen in Figure 2.11.

First, let $n(m,t)dm$ denote the number of particles with mass in $[m, m+dm)$ at time t. We denote the smallest mass at time t by $m^\star(t)$ and we have that $n(m,t) = 0$ for $m < m^\star(t)$. From this we have that the expected number of events per unit time is,

$$\dot{m}^\star(t) n(m^\star(t), t), \tag{6.3.1}$$

which is essentially capturing that the number of events depends on how fast the minimum mass is increasing and also depends on the number of particles with size that minimum mass.

The total number of particles at time t, denoted $\mathcal{N}(t)$, is given by,

$$\mathcal{N}(t) = \int_0^\infty n(m,t)\ dm. \tag{6.3.2}$$

We want to find a rate equation to describe the evolution of the profile $n(m,t)$. As before, we must take account of any loss of particles of a particular mass and similarly any gain of particles. So, for each mass $m > m^\star(t)$, loss occurs when the smallest mass coalesces with this particular larger mass. This has probability of occurring in time dt,

$$\frac{n(m,t)}{\mathcal{N}(t)} dt, \tag{6.3.3}$$

in other words, the likelihood of a particle of a particular size being hit depends on the fraction of those particles compared to the whole. We can similarly find the associated probability of the gain of a particle of mass m by considering the number of particles of size $m - m^\star(t)$, the probability of this occurring in time dt is given by,

$$\frac{n(m - m^\star(t), t)}{\mathcal{N}(t)} dt. \tag{6.3.4}$$

We make use of (6.3.1), (6.3.3) and (6.3.4) to find the rate equation for $n(m,t)$:

$$\frac{\partial n(m,t)}{\partial t} = \frac{\dot{m}^\star(t) n(m^\star(t), t)}{\mathcal{N}(t)} [n(m - m^\star(t), t) - n(m,t)]. \tag{6.3.5}$$

So far we have made use of the number of particles of a particular mass, we instead prefer to use the density of particles of a particular mass, namely,

$$\rho(m,t) = \frac{n(m,t)}{\mathcal{N}(t)}. \tag{6.3.6}$$

Inserting this into (6.3.5) we have,

$$\begin{aligned}\dot{\mathcal{N}}(t)\rho(m,t) + \mathcal{N}(t)\rho_t(m,t) &= \frac{\dot{m}^\star(t)}{\mathcal{N}(t)}\mathcal{N}(t)\rho(m^\star(t),t)[\mathcal{N}(t)(\rho(m-m^\star(t),t) - \rho(m,t)),],\\ &= \dot{m}^\star(t)\mathcal{N}(t)\rho(m^\star(t),t)\rho(m-m^\star(t),t)\\ &\qquad - \dot{m}^\star(t)\mathcal{N}(t)\rho(m^\star(t),t)\rho(m,t).\end{aligned}$$

And so we have the following rate equation,

$$\rho_t(m,t) = \dot{m}^\star(t)\rho(m^\star(t),t)\rho(m-m^\star(t),t) - \left[\dot{m}^\star(t)\rho(m^\star(t),t)\rho(m,t) + \frac{\dot{\mathcal{N}}(t)\rho(m,t)}{\mathcal{N}(t)}\right]. \tag{6.3.7}$$

We can also transform the other conditions attached to $n(m,t)$ and convert them into conditions on $\rho(m,t)$, first we consider the total number of lengths $\mathcal{N}(t)$,

$$\begin{aligned}\mathcal{N}(t) &= \int_0^\infty n(m,t)\ dm,\\ &= \int_0^\infty \mathcal{N}(t)\rho(m,t)\ dm,\\ &= \mathcal{N}(t)\int_0^\infty \rho(m,t)\ dm,\\ 1 &= \int_0^\infty \rho(m,t)\ dm.\end{aligned} \tag{6.3.8}$$

We would rather not consider the number of particles directly, and certainly not the rate at which it changes, so we consider the term $\dot{\mathcal{N}}(t)$,

$$\begin{aligned}\dot{\mathcal{N}}(t) &= \frac{d}{dt}\int_0^\infty n(m,t)\ dm,\\ &= \frac{d}{dt}\int_{m^\star(t)}^\infty n(m,t)\ dm,\\ &= \int_{m^\star(t)}^\infty \frac{\partial n(m,t)}{\partial t}\ dm - \dot{m}^\star(t)n(m^\star(t),t),\\ &= \frac{\dot{m}^\star(t)n(m^\star(t),t)}{\mathcal{N}(t)}\int_{m^\star(t)}^\infty [n(m-m^\star(t),t) - n(m,t)]\ dm - \dot{m}^\star(t)n(m^\star(t),t),\\ &= \frac{\dot{m}^\star(t)n(m^\star(t),t)}{\mathcal{N}(t)}\left(\int_0^\infty n(s,t)\ ds - \int_{m^\star(t)}^\infty n(m,t)\ dm\right) - \dot{m}^\star(t)n(m^\star(t),t),\\ &= \frac{\dot{m}^\star(t)n(m^\star(t),t)}{\mathcal{N}(t)}\left(\int_{m^\star(t)}^\infty n(s,t)\ ds - \int_{m^\star(t)}^\infty n(m,t)\ dm\right) - \dot{m}^\star(t)n(m^\star(t),t),\\ &= -\dot{m}^\star(t)n(m^\star(t),t),\\ &= -\dot{m}^\star(t)\mathcal{N}(t)\rho(m^\star(t),t).\end{aligned} \tag{6.3.9}$$

Inserting this in (6.3.7) we see that the terms in the square brackets cancel and we are left with,

$$\rho_t(m,t) = \dot{m}^\star(t)\rho(m^\star(t),t)\rho(m - m^\star(t),t). \tag{6.3.10}$$

At this point we note that equation (6.3.10) is invariant under scaling of time, this can easily be seen by considering the scaling $t \to \lambda t$. This reflects the fact that there is no 'real' time involved in the evolution since at each stage the next event happens instantaneously. As a result we are free to prescribe any suitable function to govern the evolution of the free boundary at $m^\star(t)$. We choose to let $m^\star(t) = t$ (see for example [**43**]), in other words we let 'time' evolve in step with the minimum length, which is, as a result of the evolution laws of the system, an increasing function. This leads to the following simplification,

$$\rho_t(m,t) = \rho(t,t)\rho(m-t,t). \tag{6.3.11}$$

6.3.1 Self-Similar Solutions and the Delay Differential Equation

We aim to find self-similar solutions to (6.3.11) of the form,

$$\rho(m,t) = \frac{1}{\langle m\rangle(t)}\hat{\rho}\left(\frac{m}{\langle m\rangle(t)}\right), \tag{6.3.12}$$

where $\langle m\rangle(t)$ is the mean mass at time t, defined by

$$\langle m\rangle(t) = \int_0^\infty m\rho(m,t)\,dm. \tag{6.3.13}$$

We also define the following,

$$x = \frac{m}{\langle m\rangle(t)} \quad \text{and} \quad x^\star = \frac{m^\star(t)}{\langle m\rangle(t)} = \frac{t}{\langle m\rangle(t)}. \tag{6.3.14}$$

Note that since $n(m,t) = 0$ for $m < m^\star$ it follows that $\hat{\rho}(x) = 0$ for $x < x^\star$. We begin by taking the time derivative of the distribution and relating it to the proposed self-similar solution,

$$\begin{aligned}\rho_t(m,t) &= -\frac{\langle m\rangle'(t)}{\langle m\rangle^2(t)}\hat{\rho}(x) - \frac{m\ \langle m\rangle'(t)}{\langle m\rangle^3(t)}\hat{\rho}'(x),\\ &= -\frac{\langle m\rangle'(t)}{\langle m\rangle^2(t)}[\hat{\rho}(x) - x\hat{\rho}'(x)],\\ &= -\frac{\langle m\rangle'(t)}{\langle m\rangle^2(t)}(x\hat{\rho}(x))'.\end{aligned} \tag{6.3.15}$$

It is also easy to read off the following identities,

$$\rho(t,t) = \frac{1}{\langle m\rangle(t)}\hat{\rho}(x^\star) \quad \text{and} \quad \rho(m-t,t) = \frac{1}{\langle m\rangle(t)}\hat{\rho}(x - x^\star), \tag{6.3.16}$$

which together with (6.3.15) give,

$$-\frac{\langle m\rangle'(t)}{\langle m\rangle^2(t)}(x\hat{\rho}(x))' = \frac{1}{\langle m\rangle^2(t)}\hat{\rho}(x^\star)\hat{\rho}(x-x^\star),$$

$$(x\hat{\rho}(x))' = -\frac{1}{\langle m\rangle'(t)}\hat{\rho}(x^\star)\hat{\rho}(x-x^\star), \quad x \geqslant x^\star. \tag{6.3.17}$$

At this point we note that in order for this equation to be self consistent $\langle m\rangle'(t)$ must be a constant, i.e. not a function of time. Further to this, to be consistent with the scaled co-ordinates given in (6.3.14) we must have that $\langle m\rangle'(t) = 1/x^\star$. Substituting this into equation (6.3.17) we have,

$$(x\hat{\rho}(x))' = -x^\star\hat{\rho}(x^\star)\hat{\rho}(x-x^\star), \quad x \geqslant x^\star, \tag{6.3.18}$$

$$\hat{\rho}(x) = 0, \quad 0 \leqslant x < x^\star.$$

Making use of the conditions (6.3.8) and (6.3.13) we have,

$$\int_0^\infty \rho(m,t)\;dm = 1,$$

$$\int_0^\infty \frac{1}{\langle m\rangle(t)}\hat{\rho}\left(\frac{m}{\langle m\rangle(t)}\right)\;dm = 1,$$

$$\int_0^\infty \frac{1}{\langle m\rangle(t)}\hat{\rho}\left(x\right)\;\langle m\rangle(t)dx = 1,$$

$$\int_0^\infty \hat{\rho}(x)\;dx = 1,$$

and similarly,

$$\int_0^\infty m\rho(m,t)\;dm = \langle m\rangle(t),$$

$$\int_0^\infty \frac{m}{\langle m\rangle(t)}\hat{\rho}\left(\frac{m}{\langle m\rangle(t)}\right)\;dm = \langle m\rangle(t),$$

$$\int_0^\infty x\hat{\rho}\left(x\right)\;\langle m\rangle(t)dx = \langle m\rangle(t),$$

$$\langle m\rangle(t)\int_0^\infty x\hat{\rho}(x)\;dx = \langle m\rangle(t),$$

$$\int_0^\infty x\hat{\rho}(x)\;dx = 1.$$

Combining these two integral constraints together with (6.3.18) leads to the following delay differential equation,

$$(x\hat{\rho}(x))' = -x^\star\hat{\rho}(x^\star)\hat{\rho}(x-x^\star), \quad x \geqslant x^\star, \tag{6.3.19}$$

$$\hat{\rho}(x) = 0, \quad 0 \leqslant x < x^\star,$$

$$\int_0^\infty \hat{\rho}(x)\;dx = 1 \quad \text{and} \quad \int_0^\infty x\hat{\rho}(x)\;dx = 1.$$

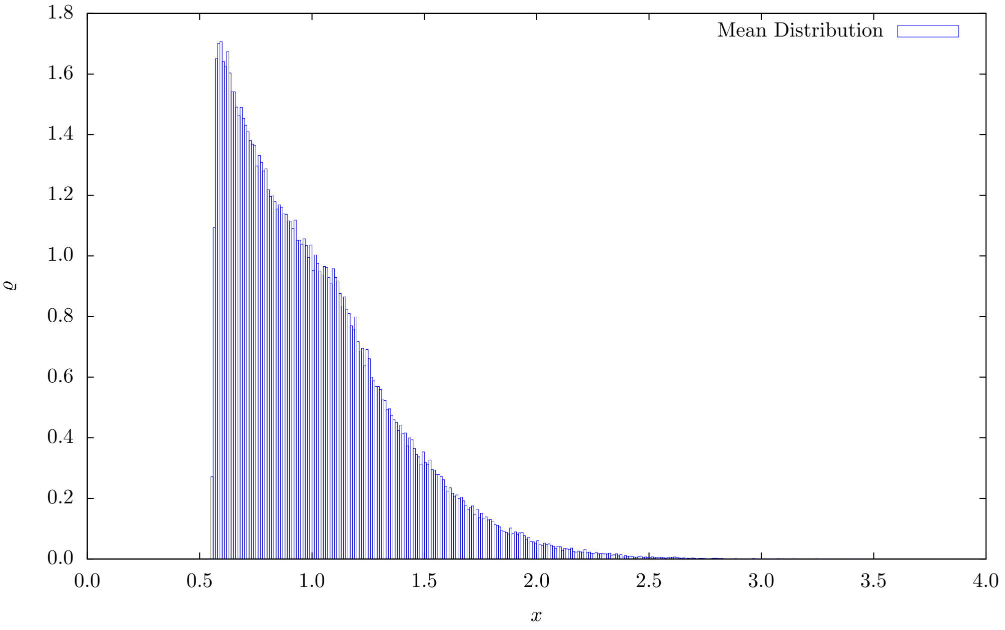


Figure 2.10 – Scaled distribution for the discretely distributed mass - one sided Paste-all model. Data was gathered from five thousand independent simulations before calculating the mean distribution seen here.

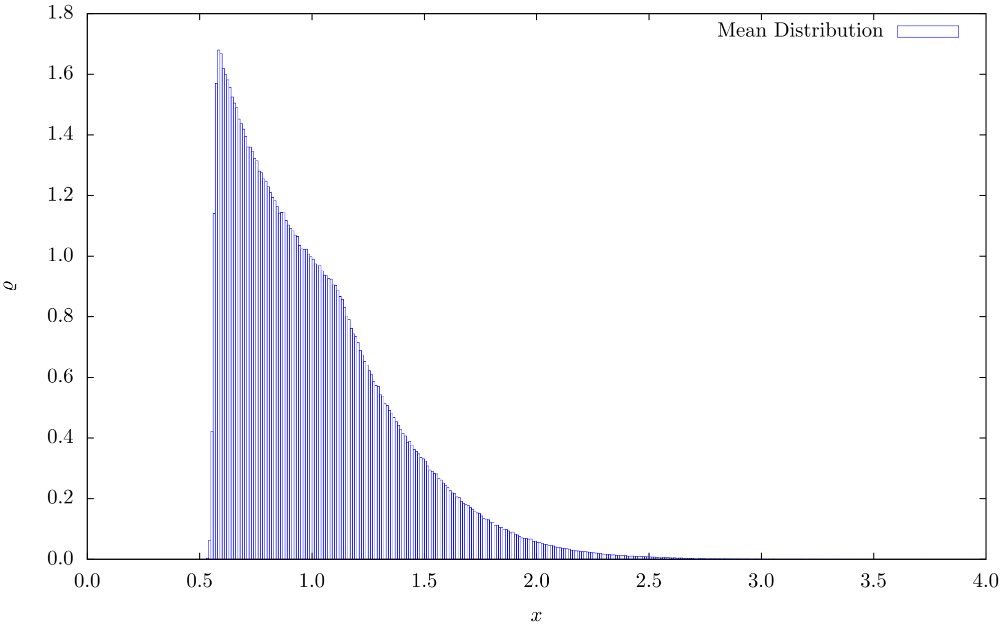


Figure 2.11 – Scaled distribution for the continuously distributed mass - one sided Paste-all model. As in the discrete case we again gather data from five thousand simulations and then generate the mean distribution here.

6.4 Solution to the Delay Differential Equation

We see that both the discrete and continuous versions lead to the same delay differential equation, (6.2.12) and (6.3.19) respectively, so it is this which we aim to solve. We proceed by considering the Laplace transform of $\hat{\rho}(x)$,

$$R(s) = \int_0^\infty e^{-sx}\hat{\rho}(x)\,dx. \tag{6.4.1}$$

The moment conditions in (6.2.12), or equivalently (6.3.19), are related to limits of the transform as follows,

$$\lim_{s\to 0^+} R(s) = \int_0^\infty \hat{\rho}(x)\,dx = 1, \tag{6.4.2}$$

$$\lim_{s\to 0^+} -\frac{dR(s)}{ds} = \int_0^\infty x\hat{\rho}(x)\,dx = 1. \tag{6.4.3}$$

If we now take the Laplace transform of the delay differential equation in (6.2.12) or (6.3.19) we have,

$$\int_0^\infty e^{-sx}\frac{d(x\hat{\rho}(x))}{dx}\,dx = -\int_0^\infty e^{-sx}x^\star\hat{\rho}(x^\star)\hat{\rho}(x-x^\star)\,dx,$$

which, given our knowledge of $\hat{\rho}(x)$, can be written,

$$\begin{aligned}
\int_{x^\star}^\infty e^{-sx}\frac{d(x\hat{\rho}(x))}{dx}\,dx &= -\int_{x^\star}^\infty e^{-sx}x^\star\hat{\rho}(x^\star)\hat{\rho}(x-x^\star)\,dx,\\
[e^{-sx}x\hat{\rho}(x)]_{x^\star}^\infty + s\int_{x^\star}^\infty e^{-sx}x\hat{\rho}(x)\,dx &= -e^{-sx^\star}x^\star\hat{\rho}(x^\star)\int_0^\infty e^{-sX}\hat{\rho}(X)\,dX, \quad X = x - x^\star,\\
-e^{-sx^\star}x^\star\hat{\rho}(x^\star) + s\int_0^\infty e^{-sx}x\hat{\rho}(x)\,dx &= -e^{-sx^\star}x^\star\hat{\rho}(x^\star)\int_0^\infty e^{-sX}\hat{\rho}(X)\,dX,\\
-e^{-sx^\star}x^\star\hat{\rho}(x^\star) - s\frac{dR(s)}{ds} &= -e^{-sx^\star}x^\star\hat{\rho}(x^\star)R(s).
\end{aligned}$$

Hence we have the following transformed equation,

$$s\frac{dR(s)}{ds} = -e^{-sx^\star}x^\star\hat{\rho}(x^\star)(1-R(s)). \tag{6.4.4}$$

We make the following observation on $R(s)$ that as $s\to\infty$ the Laplace transform is dominated by the value of $\hat{\rho}(x)$ at 0,[5] which we know is well behaved and equal to 0.

[5] Consider that as $s\to\infty$ the Laplace transform tends to 0 everywhere except where the large value of s is cancelled out in the exponent, which only happens when x is close to or exactly 0.

Hence, we can solve for $R(s)$ as follows,

$$\begin{aligned}
\int_\infty^s \frac{1}{1-R}\frac{dR}{d\tilde{s}}\,d\tilde{s} &= -x^\star\hat{\rho}(x^\star)\int_\infty^s \frac{e^{-\tilde{s}x^\star}}{\tilde{s}}\,d\tilde{s},\\
\int_0^R \frac{1}{1-\tilde{R}}\,d\tilde{R} &= -x^\star\hat{\rho}(x^\star)\int_\infty^s \frac{e^{-\tilde{s}x^\star}}{\tilde{s}}\,d\tilde{s},\\
-\log(1-R(s)) &= -x^\star\hat{\rho}(x^\star)\int_\infty^{sx^\star} \frac{e^{-u}}{u}\,du,\\
R(s) &= 1-\exp\left[x^\star\hat{\rho}(x^\star)\int_\infty^{sx^\star} \frac{e^{-u}}{u}\,du\right],\\
\text{or equivalently,}\quad R(s) &= 1-\exp\left[-x^\star\hat{\rho}(x^\star)\int_{sx^\star}^\infty \frac{e^{-u}}{u}\,du\right]. \qquad (6.4.5)
\end{aligned}$$

We now make use of the moment conditions (6.4.2) and (6.4.3). The integral on the right hand side blows up as $s \to 0^+$ leaving us with,

$$\lim_{s\to 0^+} R(s) = 1,$$

as expected. Worryingly, perhaps, we gain no information from this limit, however, we push on regardless and consider the other condition, namely (6.4.3). Taking the derivative of (6.4.5) gives,

$$R'(s) = -x^\star\hat{\rho}(x^\star)\frac{\exp\left[-sx^\star - x^\star\hat{\rho}(x^\star)\int_{sx^\star}^\infty \frac{e^{-u}}{u}\,du\right]}{s}. \qquad (6.4.6)$$

Our aim is to take the limit as $s \to 0^+$. We note that on the right hand side the limit returns $0/0$ and so we could try and proceed with l'Hôpital's rule. Unfortunately taking the derivative of the fraction returns the same integral in the numerator and so we enter a loop. To avoid entering this loop we instead make use of the *exponential integral* and its relatives, in particular considering the asymptotic properties of these integrals. We make use of the fact that,

$$\int_x^\infty \frac{e^{-u}}{u}\,du = -\gamma - \log x + \ldots, \qquad (6.4.7)$$

as $x \to 0^+$ to leading order and where γ is the *Euler-Mascheroni* constant,

$$\gamma = 0.577215664901532860606 5\ldots$$

Making use of (6.4.7) in (6.4.6) we have, following some rearrangement,

$$R'(s) = -x^\star\hat{\rho}(x^\star)s^{x^\star\hat{\rho}(x^\star)-1}\exp[-sx^\star + x^\star\hat{\rho}(x^\star)(\gamma+\log x^\star)]. \qquad (6.4.8)$$

Now we consider the limit as $s \to 0^+$, which imposes certain conditions in order for the limit to match (6.4.3). First, we can't have any dependence on s or this would force the limit to 0 and so the exponent on s must vanish, equivalently $x^\star\hat{\rho}(x^\star) = 1$. With this in mind, we must have the exponential having 0 as its exponent, which implies $\gamma + \log x^\star = 0$ or $x^\star = e^{-\gamma}$. As a result we can write down the final form of the delay equation,

$$\begin{aligned} (x\hat{\rho}(x))' &= -\hat{\rho}(x - x^\star), \quad x \geqslant x^\star, \\ \hat{\rho}(x) &= 0, \quad 0 \leqslant x < x^\star, \\ x^\star = e^{-\gamma} \quad &\text{and} \quad \hat{\rho}(x^\star) = \frac{1}{x^\star}. \end{aligned} \tag{6.4.9}$$

We now proceed to find the solution $\hat{\rho}(x)$ to the delay equation. We employ the standard technique for such equations by building up the solution piece by piece. Since we know the solution $\hat{\rho}(x) = 0$ for $0 \leqslant x < x^\star$ we begin by considering the region $x^\star \leqslant x < 2x^\star$.

$$\begin{aligned} (x\hat{\rho}(x))' &= -\hat{\rho}(x - x^\star), \\ x\hat{\rho}(x) &= -\int_0^x \hat{\rho}(X - x^\star)\, dX + C. \end{aligned}$$

Making use of the information we already have we can easily find the following,

$$\begin{aligned} x\hat{\rho}(x) &= -\int_0^{2x^\star} \hat{\rho}(X - x^\star)\, dX + C, \\ &= -\int_{-x^\star}^{x^\star} \hat{\rho}(\tilde{X})\, d\tilde{X} + C, \qquad \tilde{X} = X - x^\star, \\ &= C. \end{aligned}$$

We know that $x^\star\hat{\rho}(x^\star) = 1$ and so clearly $C = 1$, hence,

$$\hat{\rho}(x) = \frac{1}{x}, \qquad x^\star \leqslant x < 2x^\star.$$

Continuing this process for subsequent regions yields the following solution,

$$\begin{aligned} \hat{\rho}(x) &= 0, \quad 0 \leqslant x < x^\star, \\ \hat{\rho}_1(x) := \hat{\rho(x)} &= \frac{1}{x}, \quad x^\star \leqslant x < 2x^\star, \\ \hat{\rho}_2(x) := \hat{\rho}(x) &= \frac{1 - \ln(x - x^\star) + \ln x^\star}{x}, \quad 2x^\star \leqslant x < 3x^\star, \\ \hat{\rho}_3(x) := \hat{\rho}(x) &= \frac{1 - \ln\left(\frac{x}{x^\star} - 1\right)}{x} + \frac{1}{x}\left[\frac{\pi^2}{12} + \operatorname{dilog}\left(\frac{x}{x^\star} - 1\right) + \ln\left(\frac{x}{x^\star} - 2\right)\ln\left(\frac{x}{x^\star} - 1\right)\right], \\ &\qquad 3x^\star \leqslant x < 4x^\star, \end{aligned} \tag{6.4.10}$$

where,

$$\operatorname{dilog}(x) = \int_1^x \frac{\ln t}{1 - t}\, dt.$$

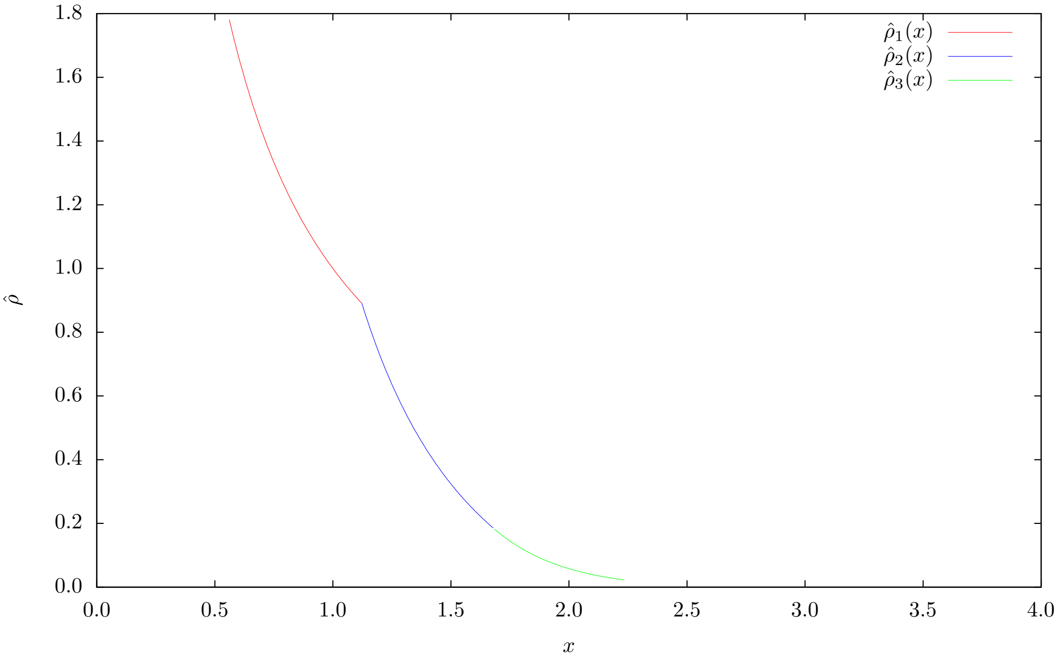


Figure 2.12 – Solution to the delay differential equation, (6.2.12) or equivalently (6.3.19). Different coloured sections correspond to the different sections of the solution shown in (6.4.10).

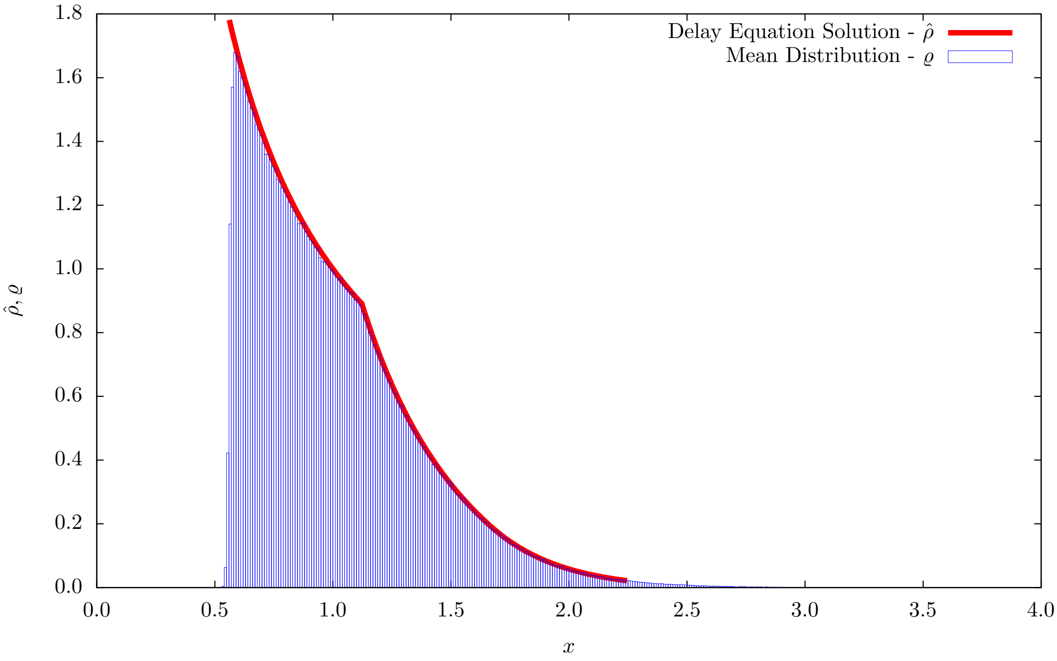


Figure 2.13 – Solution to the delay differential equation together with numerical data from continuously distributed simulations, shown in Figure 2.11, for comparison. A lack of significant correlations in the system means our predicted distribution should closely match the data, as we see here.

7 Model Comparisons

In the previous sections we have detailed two models, one a continuously evolving particle model and the other a probabilistic, discrete particle model. Here we draw several conclusions including discussion of the connections between the two models.

7.0.1 Vacuum Interval

In Figure 2.7 we see that the vacuum interval increases in size as $p \to \infty$ and we estimated that the asymptotic value at infinity was 0.557397. This compares well with the paste-all model which has a vacuum interval which ends at $e^{-\gamma} \approx 0.561459$. We also mention, as noted before, that the distribution appears to become more like a jump to the peak at the end of the vacuum interval, which again matches well with the paste-all model.

7.0.2 Support of the Distribution

If we again return to Figure 2.7 and consider the 5% cut-off value, we estimated there that the asymptotic value was in the region of 1.65. Calculation of the cut-off value for the paste-all model distribution returns the value 1.67684, which we again see compares favourably.

7.0.3 Peak of the Distribution

If we now compare the size of the peak of the distribution as p is increased we see that it tends to level off at some fixed value. In fact, if we perform the same analysis as we have for the size of the vacuum interval we find an asymptotic value at infinity of 1.655214. Whilst this compares reasonably well with the paste-all value of 1.781072 it doesn't match quite as closely as the vacuum interval or distribution support values. A possible source of the discrepancy is found by noting that the simulations we were able to carry out were unable to move far enough away from low p values to recreate the sharp jump in the distribution found in the paste-all case. This results in the appearance of several smaller bars of the histogram to the left of the peak which don't wash out as we increased p as far as possible.

7.0.4 'Bump' on Distribution Tail

We make only a brief note that as p was increased a 'bump' on the tail of the distribution was observed. Its position was seen to be around twice the minimum mass. We see that the paste-all model has a definite change in the slope, in other words a 'bump', at precisely twice the minimum. This appears to match up well between the models.

7.0.5 System Dynamics

In Section 5 we saw that when p is large enough the system behaves in a very particular way, namely the minimum mass particle(s) collide next and that in the time to that collision the others remain essentially fixed. This analysis was of course done with the paste-all model in mind, since this behaviour for large p is exactly how the paste-all model behaves.

7.1 Overall Conclusions

We have considered two distinct particle models, one an aggregating, ballistic particle model and the other a variant of the so-called 'paste-all' model [**30**]. In both cases we studied the scale invariant distributions and several of the features therein. In the case of the paste-all model we solved for the exact scale invariant distribution (Figure 2.12) and in Figure 2.14 we see that the distributions, for $p \geqslant 10$, already match closely to the delay differential equation solution. In particular we mention the emergence of the 'bump' on the tail of the distribution. The emergence and associated location of this as p is increased is very interesting, and further to this, the similarity to the paste-all model is remarkable.

We have found asymptotic values for the vacuum interval, the peak and the 5% cut-off values, all of which agree well with the exact values found in the paste-all case. We also showed that the dynamics of the ballistic system for large enough p becomes exactly the dynamics of the paste-all model, in other words despite being a continuously evolving ballistic model, when p is large enough it behaves like a statistical, min-driven system.

In summary, we have shown that for large enough p the aggregating, ballistic particle model becomes almost identical to the paste-all model in dynamics and associated distribution. We suspect that in the limit $p \to \infty$ that the aggregating, ballistic particle model becomes exactly the paste-all model. Consideration of the analysis in Section 5 in the limit also supports this claim.

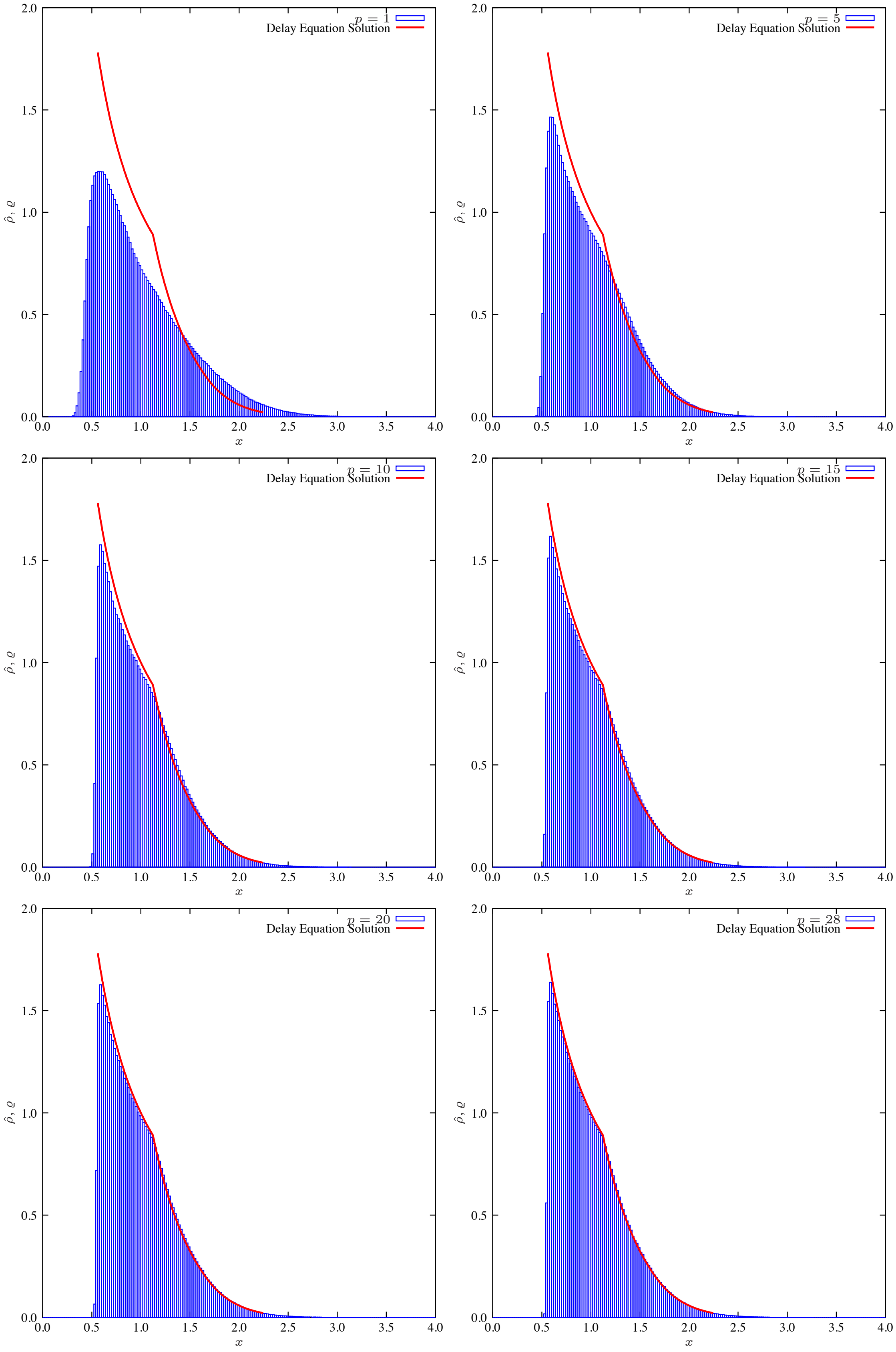


Figure 2.14 – Delay differential equation solution, (6.4.10), overlaid onto several $\mathcal{ABP}$ distributions for various p. As expected there is an increasingly better fit as p is increased since the system behaves more like the paste-all model for large p.

Chapter 3

CKS Equation, Numerics and Dynamical Systems

1 Introduction and Background

The theory of crystal surfaces below the *roughening temperature* has seen significant progress recently [**73**, **105**] and as such has seen a large number of theoretical predictions emerge [**5**, **10**, **90**]. The general overall picture of this research is to take fundamental knowledge of the atomic structure of the surface and find a continuum prediction of the complete evolution of the surface. The type of problems which we consider frequently involve the introduction of new matter, adatoms, to the surface from the surrounding system which once on the surface move in several possible manners, which can be predicted by the structure of the underlying crystal. From these predictions and often in the case of several assumptions, which we shall discuss later, the hope is that we will discover a governing equation which predicts the evolution of the surface from a known initial state.

Similar dynamics to those of crystal surfaces appear in other distinct contexts [**102**, **111**], so an understanding of the method and processes in this context could prove to be insightful in other contexts too. Even within crystal surfaces there are several possible growth procedures possible, such as moving steps and island nucleation/growth, [**10**, **67**, **94**, **107**, **108**, **125**], we limit ourselves here to so-called *step meandering* and *step bunching.*

The typical scenario in which these step effects are exhibited is that of a vicinal crystal surface. A vicinal surface is one which overall has a slight miscut in the angle of the surface relative to the planar structure which the material sits and as a result the surface must make a series of jumps by a single atom height (or multiple thereof) as we move across the entire surface. As already mentioned, it is frequently the case that we then introduce new material to the system. These new atoms hit the surface and then proceed to diffuse on the surface before hitting a step and then generally attaching at some location. The process with which this new material is deposited is often *molecular beam epitaxy* (MBE), which is one type of process used when growing crystal structures, although other methods do exist. Individual atoms can also spontaneously be ejected from the surface back into the surroundings, this is known as desorption.

The two types of growth we detail are step meandering and step bunching. Step meandering, put simply, is the ability of the steps on the crystal surface to move around in a wave like manner. First detailed by Bales and Zangwill in [**5**] this type of step movement is found in many circumstances, for example [**63**, **88**, **113**]. The second type of growth is step bunching which is essentially where microscopic processes allow each step to move in a different manner to the others and as a result the steps can bunch together, rather than stay apart, e.g. [**23**, **95**, **109**]. Generally speaking the types of process which occur at the atomic level differ and dictate the type of instability present, significant detail on vicinal crystal surfaces in general can be found in [**73**].

We give some brief detail on one aspect of step growth since it has relevance to the specific context we study later, the so-called *Ehrlich-Schwoebel* (E-S) effect [**32**, **97**]. As mentioned we deposit individual atoms onto the surface of the crystal. We also noted that these then move until they attach to one of the existing steps. What wasn't clear is the ability for an atom to join either of the two[1] steps it sits between, more specifically the up step or the down step? The E-S effect is then essentially a characterisation of the possible asymmetry in an adatom's ability to go up or down a step. Such an asymmetry was shown to produce a step bunching instability by Shcwoebel [**97**]. It is often the case that the extreme case of one-sided processes are assumed, that is individual atoms can only join onto one of the two possible steps around it or, indeed, that this effect is presumed to be negligible, that is there is no little or no asymmetry in an adatoms ability to join either step.

1.1 The Conserved Kuramoto-Sivashinsky Equation

With a wealth of examples of such step effects, for example see [**23**, **24**, **63**, **73**, **87**, **88**, **90**, **95**, **105**, **109**, **113**, **126**], we choose to focus our attention on one particular

[1] Based on a simplified train of steps structure.

example of step meandering. Taking the growth of a Si(001) vicinal surface under the assumptions of negligible desorption and Ehrlich-Schwoebel effect, together with a lack of elastic interaction between steps, Frisch and Verga [**39**, **40**] conclude that the unstable mode for a wandering step satisfies the following equation,

$$u_t = -[u + u_{xx} + u_x^2]_{xx}. \tag{1.1.1}$$

This equation is known as the *conserved Kuramoto-Sivashinsky* (CKS) equation, where $u(x,t)$ describes the location of the step above some axes and x and t denote the coordinate along the step and time respectively. A typical example of a step and the notation used is shown in Figure 3.1.

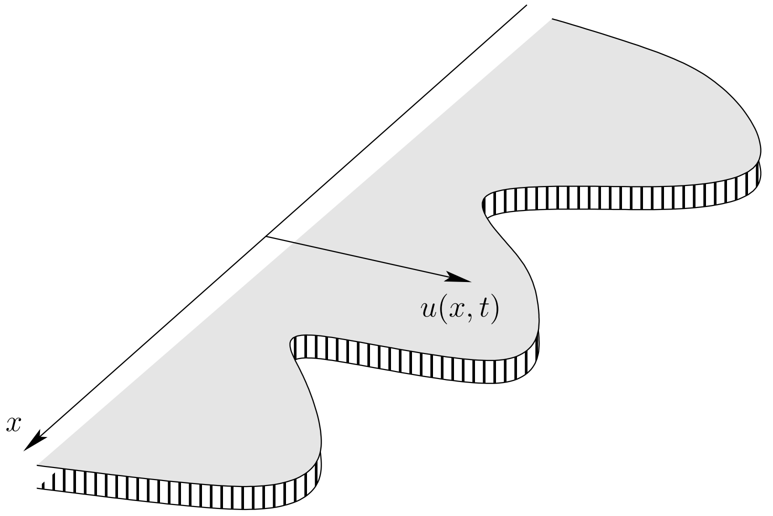


Figure 3.1 – Typical configuration of a step on the material surface indicating the notation used throughout, namely, $u(x,t)$ is the step location, x the spatial coordinate and t time.

The assumptions which lead to this conclusion essentially amount to considering the case when there is no loss from the surface (desorption) and that each step is effectively isolated in space or, more specifically, has no effect on other steps (no elastic interaction). Neglecting the E-S effect simply allows for adatoms to join a preexisting step from either the step above or below with no presumption of favourability to one type of step (up or down). These assumptions allow us to consider the evolution of one single step alone and that the height of this step is modelled by the CKS equation, (1.1.1).

For comparison and justification of the conserved characterisation we mention the well studied *Kuramoto-Sivashinsky* (KS) equation,

$$u_t = -[u + u_{xx}]_{xx} + u_x^2,$$

which was originally studied in the context of reaction-diffusion systems [**64**] and flame propagation [**102**]. Another related equation worth mentioning is identical to the CKS equation with the addition of a term of the type u_{xxx}, [**44**], which arose in a similar context to the CKS equation but for step bunching on the surface of Si(111) and once again neglecting desorption. The original work in [**64**, **102**] has inspired many streams of research across a spectrum of different fields, including the materials science context present. It is not surprising, therefore, to see the conserved Kuramoto-Sivashinsky equation be named as such, based on this rich history.

Let us return to the CKS equation and consider the solution profile $u(x, t)$. We first draw our attention to the overall emerging structure. Simulations carried out in [**39**, **40**] and here (detail in Sections 2 and 3) indicate the emergence of a two scaled solution with large arced regions separated by small connecting regions. A simple similarity argument predicts a growth in the typical size of the large structures that behaves like $t^{1/2}$, [**39**]. We note that in the presence of the additional u_{xxx} term the typical size of step bunches also scales like $t^{1/2}$ [**44**]. A typical simulation of the CKS equation can be seen in Figure 3.2 where we can see the evolution of the solution profile and as the number of arcs reduces, the average size of each increases as expected.

Further analysis of the structure of the solution profile shows that a single stationary parabola, $u(x, t) = a - (x - \bar{x})^2/4$, for some a and $\bar{x}$, is a solution on $\mathbb{R}$ and that the large regions of the solution profile are essentially independent of the fourth derivative term [**39**]. Trying a similarity solution in the reduced equation (without the fourth order term) admits bounded parabolas, $u(x, t) = -x^2/4$ for $|x| < x_0(t)$, and zero outside this range. The small connecting regions between parabolas are shown to be governed by the separate reduced equation, $u_{xx} + u_x^2 = k^2$, where k is a constant, and results in an expected $\log[\cosh k(x - x_0(t))]$ type structure at the edge of the parabola. The parameter k can be determined by matching the inner solution to the outer parabolas to find that $k = x_0(t)/2 \sim \sqrt{t}$, or roughly quarter the width of the neighbouring parabolas, and as a consequence it can be shown that the curvature, κ, in these connecting regions behaves linearly in time, $\kappa \sim t$. From here it is concluded that the solution profile can be thought of as a series of parabolas of the typical form described earlier connected by small regions

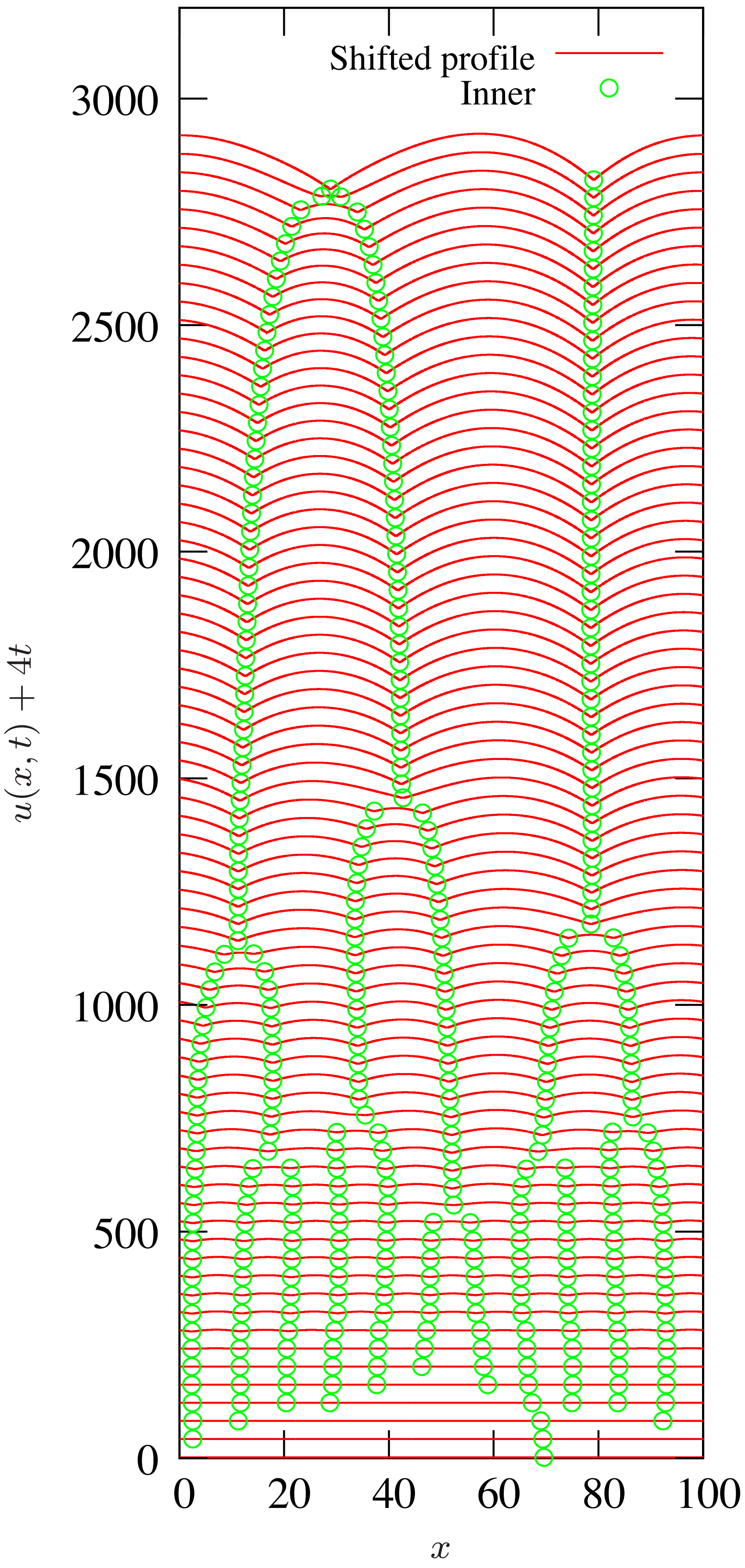


Figure 3.2 – Typical simulation of the CKS equation where the profile has been shifted vertically in time for ease of visualisation. Green points indicate the location of the small joining regions which then give some idea of the scale of each arc, showing a general increase in time.

with linearly increasing curvature. This is the scenario which leads into the derivation of a simple dynamical system which, hopefully, effectively captures the entire evolution.

1.2 Proposed Dynamical System

With the emerging profile now somewhat understood, Politi and ben-Avraham discuss a particle model which they propose captures the essential aspects of the solution profile evolution [**89**]. The derivation of this model presumes the solution profile can be thought of as a series of translations of arcs of the universal parabola $p(x) = -x^2/4$; an assertion also made in [**39**]. Letting x_i denote the join location between such successive parabolic arcs, the authors of [**89**] argue that,

$$\frac{dx_i}{dt} = \frac{1}{(x_{i+1} - x_i)} - \frac{1}{(x_i - x_{i-1})}.$$

Setting $l_i := x_{i+1} - x_i$ we can write this particle model in equivalent length form as,

$$\frac{dl_i}{dt} = \frac{1}{l_{i+1}} - \frac{2}{l_i} + \frac{1}{l_{i-1}}. \tag{1.2.1}$$

1.3 Objectives and Approach

Our main objective is to provide a critique of the proposed dynamics, (1.2.1), by directly comparing simulations of the CKS equation with the predicted results of the dynamical system.

We shall begin our exploration in Section 2 by conducting extensive numerical simulations of the CKS equation. Several methods have been adopted and we detail these and associated background material where appropriate. Following on from this in Section 3 we shall benchmark the code against the expected analytical results to check that our code is accurate and reliable. Finally, in Section 4 we shall consider the coarsening pathways within the system and carry out our final comparisons between the PDE dynamics and those predicted in [**89**], shown in (1.2.1).

2 Numerical Solution of CKS Equation

We begin by presenting the numerical procedures adopted for solving the CKS equation (1.1.1). Three such methods were attempted with varying levels of success. We begin by using a simple finite difference method on a fixed, uniform grid together with an explicit update of the nodal values. The data gathered here lead us to consider two distinct numerical methods, first adaptive grid methods and second a hybrid implicit-explicit update rule on a fixed, uniform grid.

2.1 Explicit update - Fixed Grid

The first method employed to numerically solve the CKS equation was a standard finite difference approach on a fixed, uniform grid with an explicit update rule. Formally, we have our equation together with an initial condition, $u(x,0) := u_0(x)$, and want to solve on a domain of length L with periodic boundary condition.

First let us choose a integer $n > 0$ and set $h = L/n$. This number h will then represent the size of the grid spacing on our domain and so we have essentially partitioned the domain into n equal parts. Setting $x_i = ih$ for $i = 0, 1, \ldots, n-1$, we now have our discrete spatial grid for the finite difference scheme which we wish to set up. Note that since we aim to solve on a periodic domain the point $x_n := x_0$, and so on; essentially wherever there is a spatial index i we are in fact taking $i \pmod{n}$. This will become important later when we discuss the finite difference expressions near the boundary. We must also discretise time and so we select a time step Δt and set $t_j = j\Delta t$ for $j = 0, 1, \ldots$. Combining these two we then have a set of grid points in the completely discretised domain, which we shall denote (x_i, t_j).

We want to use this grid to numerically solve the CKS equation and so we define several objects. First we define the function $\mathcal{U}$ which takes a spatial node and a time and defines a real number, or explicitly,

$$\mathcal{U} : \{0, 1, \ldots, n-1\} \times \{0, 1, \ldots\} \longrightarrow \mathbb{R}.$$

If we were to fix a choice of grid point and time we could write this as $\mathcal{U}(i,j)$, however, we shall define this to be $\mathcal{U}_i^j := \mathcal{U}(i,j)$. From here we then define the vector of $\mathcal{U}_i^j$ values as,

$$\mathbf{U}^j := (\mathcal{U}_0^j, \mathcal{U}_1^j, \ldots, \mathcal{U}_{n-1}^j)^T$$

We then proceed to discretise each of the elements of the CKS equation on the numerical grid. First we must approximate the derivatives in the CKS equation by making use of finite differences. We define four operators on the uniformly spaced numerical grid values $\mathcal{U}_i^j$, one finite difference type expression for each spatial derivative present. The four operators are,

$$(D\mathcal{U})_i^j := \frac{1}{2}(\mathcal{U}_{i+1}^j - \mathcal{U}_{i-1}^j), \tag{2.1.1}$$

$$(D_2\mathcal{U})_i^j := \mathcal{U}_{i+1}^j - 2\mathcal{U}_i^j + \mathcal{U}_{i-1}^j, \tag{2.1.2}$$

$$(D_3\mathcal{U})_i^j := \frac{1}{2}(\mathcal{U}_{i+2}^j - 2\mathcal{U}_{i+1}^j + 2\mathcal{U}_{i-1}^j - \mathcal{U}_{i-2}^j), \tag{2.1.3}$$

$$(D_4\mathcal{U})_i^j := \mathcal{U}_{i+2}^j - 4\mathcal{U}_{i+1}^j + 6\mathcal{U}_i^j - 4\mathcal{U}_{i-1}^j + \mathcal{U}_{i-2}^j. \tag{2.1.4}$$

On a uniform grid, such as we have here, the full finite difference expressions are calculated using the Taylor series in x and are as follows,

$$\frac{\partial u}{\partial x}(x,t) = \frac{u(x+h,t) - u(x-h,t)}{2h} + O(h^2), \tag{2.1.5}$$

$$\frac{\partial^2 u}{\partial x^2}(x,t) = \frac{u(x+h,t) - 2u(x,t) + u(x-h,t)}{h^2} + O(h^2), \tag{2.1.6}$$

$$\frac{\partial^3 u}{\partial x^3}(x,t) = \frac{u(x+2h,t) - 2u(x+h,t) + 2u(x-h,t) - u(x-2h,t)}{2h^3} + O(h^2), \tag{2.1.7}$$

$$\frac{\partial^4 u}{\partial x^4}(x,t) = \frac{u(x+2h,t) - 4u(x+h,t) + 6u(x,t) - 4u(x-h,t) + u(x-2h,t)}{h^4} + O(h^2), \tag{2.1.8}$$

where each expression is of second order. If we neglect higher order terms and make use of our four defined operators on a uniform grid we have,

$$\frac{\partial u}{\partial x}(x_i,t_j) \approx \frac{1}{h}(D\mathcal{U})_i^j, \tag{2.1.9}$$

$$\frac{\partial^2 u}{\partial x^2}(x_i,t_j) \approx \frac{1}{h^2}(D_2\mathcal{U})_i^j, \tag{2.1.10}$$

$$\frac{\partial^3 u}{\partial x^3}(x_i,t_j) \approx \frac{1}{h^3}(D_3\mathcal{U})_i^j, \tag{2.1.11}$$

$$\frac{\partial^4 u}{\partial x^4}(x_i,t_j) \approx \frac{1}{h^4}(D_4\mathcal{U})_i^j, \tag{2.1.12}$$

Note that if we make use of the $i \pmod{n}$ argument these expressions are valid at all the grid points. In a similar manner we can calculate the appropriate first order expression for the time derivative from the standard first order finite difference expression,

$$\frac{\partial u}{\partial t}(x,t) = \frac{u(x,t+\Delta t) - u(x,t)}{\Delta t} + O(\Delta t),$$

which expressed using our numerical values and having neglected the error terms is approximated as,

$$\frac{\partial u}{\partial t}(x_i, t_j) \approx \frac{1}{\Delta t}(\mathcal{U}_i^{j+1} - \mathcal{U}_i^j).$$

Collecting all these expressions together and substituting them into the CKS equation, (1.1.1), we have,

$$\frac{1}{\Delta t}(\mathcal{U}_i^{j+1} - \mathcal{U}_i^j) = -\frac{1}{h^2}(D_2\mathcal{U})_i^j - \frac{1}{h^4}(D_4\mathcal{U})_i^j - 2\left(\left(\frac{1}{h^2}(D_2\mathcal{U})_i^j\right)^2 + \frac{1}{h}(D\mathcal{U})_i^j \frac{1}{h^3}(D_3\mathcal{U})_i^j\right). \tag{2.1.13}$$

This equation contains only one expression at the updated t_{j+1} time and so we rearrange this to solve for $\mathcal{U}_i^{j+1}$,

$$\begin{aligned}\mathcal{U}_i^{j+1} &= \mathcal{U}_i^j + \Delta t\left[-\frac{1}{h^2}(D_2\mathcal{U})_i^j - \frac{1}{h^4}(D_4\mathcal{U})_i^j - 2\left(\left(\frac{1}{h^2}(D_2\mathcal{U})_i^j\right)^2 + \frac{1}{h}(D\mathcal{U})_i^j \frac{1}{h^3}(D_3\mathcal{U})_i^j\right)\right], \\ &= \mathcal{U}_i^j - \frac{\Delta t}{h^4}\left[h^2(D_2\mathcal{U})_i^j + (D_4\mathcal{U})_i^j\right] - \frac{2\Delta t}{h^4}\left(\left((D_2\mathcal{U})_i^j\right)^2 + (D\mathcal{U})_i^j(D_3\mathcal{U})_i^j\right). \qquad (2.1.14)\end{aligned}$$

We now have one equation of this form for each of the spatial grid points and can use this formula to update each nodal value in time. We can rewrite all of these n equations in one simple matrix form as,

$$\mathbf{U}^{j+1} = \left(I - \frac{\Delta t}{h^4}A\right)\mathbf{U}^j - \frac{2\Delta t}{h^4}\mathbf{F}(\mathbf{U}^j), \tag{2.1.15}$$

where

$$A := \begin{bmatrix} 6-2h^2 & -4+h^2 & 1 & 0 & \cdots & \cdots & 0 & 1 & -4+h^2 \\ -4+h^2 & 6-2h^2 & -4+h^2 & 1 & 0 & \cdots & \cdots & 0 & 1 \\ 1 & -4+h^2 & 6-2h^2 & -4+h^2 & 1 & 0 & \cdots & \cdots & 0 \\ 0 & 1 & -4+h^2 & 6-2h^2 & -4+h^2 & 1 & 0 & \cdots & 0 \\ \vdots & & \ddots & & \ddots & & \ddots & & 0 \\ 0 & \cdots & \cdots & 0 & 1 & -4+h^2 & 6-2h^2 & -4+h^2 & 1 \\ 1 & 0 & \cdots & \cdots & 0 & 1 & -4+h^2 & 6-2h^2 & -4+h^2 \\ -4+h^2 & 1 & 0 & \cdots & \cdots & 0 & 1 & -4+h^2 & 6-2h^2 \end{bmatrix}$$

and $\mathbf{F}$ defines a nonlinear function on the current state and returns the value of the nonlinear parts of the equation, explicitly

$$\mathbf{F} : \mathbb{R}^n \longrightarrow \mathbb{R}^n,$$

and the ith component is given by,

$$\left((D_2\mathcal{U})_i^j\right)^2 + (D\mathcal{U})_i^j(D_3\mathcal{U})_i^j.$$

We now have an equation which is valid for all the grid points in both spatial and temporal dimensions and can be used to update the solution, in terms of the points $\mathcal{U}_i^j$, by a single time step to the updated solution, the points $\mathcal{U}_i^{j+1}$. The initial condition gives us the value of the discrete solution at the grid points as,

$$\mathcal{U}_i^0 := u_0(x_i),$$

for each of the spatial grid points, $i \in \{0, 1, \ldots, n-1\}$. From here we can simply march forward in time making use of (2.1.14) at each step since we know the previous time step values iteratively starting from the initial condition.

2.1.1 Discussion of Explicit Method

Unfortunately two factors arose during these simulations, both of which hindered the progress of the method. The first was a time step restriction and second was the need for large numbers of points to resolve the emerging boundary layers.

Time stepping

Explicit schemes are often hindered by time step issues and this method is no different. The fourth order nature of the equation in question forced us to consider very small time steps to ensure stability. In particular, if we consider only the linear parts of equation (2.1.15) involving the matrix term,

$$\left(I - \frac{\Delta t}{h^4} A\right), \tag{2.1.16}$$

we can find a rough estimate of the type of restriction we must observe. This matrix is a circulant matrix and so we can easily calculate the eigenvalues as follows. First let us define six quantities of use to us, where n is the number of grid points used,

$$\begin{gathered} \omega = e^{\frac{2\pi i}{n}}, \\ c_0 = 1 - \frac{\Delta t}{h^4}(6 - 2h^2), \quad c_1 = \frac{\Delta t}{h^4}(4 - h^2), \quad c_2 = -\frac{\Delta t}{h^4} \\ c_{n-1} = \frac{\Delta t}{h^4}(4 - h^2) \quad \text{and} \quad c_{n-2} = -\frac{\Delta t}{h^4}, \end{gathered}$$

where ω is therefore an nth root of unity and the subscripts on the c quantities represent the respective location of the term along the first row of the matrix in question, running from 0 through to $n-1$. This is the standard way of indexing a circulant matrix since each subsequent row is simply a shift of the first. Note that these numbers $c_\star$ come from the entries of the matrix (2.1.16) above. From here we can write down an expression for the eigenvalues λ_j, namely,

$$\lambda_j = c_0 + c_{n-1}\omega^j + c_{n-2}\omega^{2j} + c_2\omega^{-2j} + c_1\omega^{-j}, \quad j \in \{0, 1, \ldots, n-1\}.$$

Inserting the appropriate expressions from above we then have,

$$\lambda_j = 1 - \frac{\Delta t}{h^4}(6 - 2h^2) + \frac{\Delta t}{h^4}(4 - h^2)(\omega^j + \omega^{-j}) - \frac{\Delta t}{h^4}(\omega^{2j} + \omega^{-2j}),$$

which can be simplified by collection of terms and use of the identity,

$$\cos x = \frac{e^{ix} + e^{-ix}}{2},$$

to,

$$\lambda_j = 1 + \frac{\Delta t}{h^4}\left[8\cos\left(\frac{2\pi j}{n}\right) - 2\cos\left(\frac{4\pi j}{n}\right) - 6\right] + \frac{\Delta t}{h^2}\left[2 - 2\cos\left(\frac{2\pi j}{n}\right)\right]$$

In order to ensure stability, at least as far as the linear terms go, we must ensure that the spectral radius of the matrix shown in (2.1.16) is less than 1. More specifically this means the largest eigenvalue in modulus must be less than one, and hence we must consider under what circumstances can we guarantee the following,

$$\left|1 + \frac{\Delta t}{h^4}\left[8\cos\left(\frac{2\pi j}{n}\right) - 2\cos\left(\frac{4\pi j}{n}\right) - 6\right] + \frac{\Delta t}{h^2}\left[2 - 2\cos\left(\frac{2\pi j}{n}\right)\right]\right| \leqslant 1, \quad \forall j.$$

We simplify this expression for two related reasons, first, we only expect to find some rough idea of the kind of restriction necessary for stability and note, again, that we are only considering the linear parts here so it isn't guaranteed to be a complete restriction on the whole equation anyway. Second, since we expect the grid spacing to be much smaller than one ($h \ll 1$) we can assume that the leading order behaviour will be sufficient to gain reasonable insight into the issue. As a result we consider the simplified case,

$$\left|1 + \frac{\Delta t}{h^4}\left[8\cos\left(\frac{2\pi j}{n}\right) - 2\cos\left(\frac{4\pi j}{n}\right) - 6\right]\right| \leqslant 1, \quad \forall j.$$

The expression on the left of the inequality attains a maximum at $j = 0$ or $j = n$ of 1, so there is no constraint there. We must therefore consider the case when,

$$-1 \leqslant 1 + \frac{\Delta t}{h^4}\left[8\cos\left(\frac{2\pi j}{n}\right) - 2\cos\left(\frac{4\pi j}{n}\right) - 6\right].$$

The term inside the square brackets is minimal when $j = n/2$ and inserting this gives,

$$-1 \leqslant 1 - 16\frac{\Delta t}{h^4},$$

which in turn gives,

$$\Delta t \leqslant \frac{h^4}{8}.$$

This is our rough estimate of the type of restriction we must observe if we are to have any chance of achieving numerical stability in the method. It is now clear that with small grid spacings we must use very small time steps to remain stable.

Time steps of this size are therefore already a problem for the efficiency of the system but this problem is, in fact, only worsened by an increase in domain length. Naively an increase in domain length whilst maintaining the number of grid points would increase h, the grid step size, and hence increase Δt. On larger domains, however, to maintain enough spatial resolution we must increase the number of points in the system to reflect the size of the domain, and so we cannot expect any increase in the possible time step from this point of view. For example, suppose we are now required to double the number of grid points, as a result of doubling the domain, we must then perform twice as many calculations per time step. We then see that for large domains where we may require vast grid point numbers we need to perform significantly more calculations to update the system.

Further to this, if we suppose we require a doubling in the number of grid points to maintain numerical accuracy on a fixed domain we see that the time step required is reduced by a factor of 16. This means we must perform 16 times more time steps to reach the same fixed future time T, say. Combined with the fact we need to perform twice as many calculations per time step and we see a rough 32 times increase in computational effort to reach the time T. These types of restrictions unfortunately result in a strong playoff between maintaining accuracy and returning results in a reasonable time frame.

Under resolved boundary layers

The second issue which arose during simulations was an inability to capture the inner boundary layers. As the solution profile coarsens the number of parabola structures decreases and the overall size of those which remain increases. This causes the jump in slope across the inner layers to increase with time. Similarly the curvature in these regions grows larger as the number of such boundary layers drops. If we try to simulate with too few points in our grid there is a chance that the scale of the jump in slope and size of the curvature will become too much for the grid to handle and lead to numerical instability or indeed collapse. At first a straightforward method to combat this would be to increase the number of points in the grid but sadly, in light of the first time stepping issue, we see that we are now trying to balance two competing influences on our method. Increasing the number of grid points to combat the need for precision in the boundary layers results in huge increases in computational time.

As a result of these issues we have found this method to be unsuitable for providing robust numerical data and so seek out alternative methods. The inability to accurately capture the inner solution structure without the system collapsing whilst also outputting results in a reasonable time frame proved too much for the method to overcome. Our first attempt at an alternative shall be to try and counter both criticisms of the explicit method at once. In particular we try to keep the total number of grid points as low as possible whilst making sure there are a reasonable number of points in the boundary layers. This is achieved by use of so-called *adaptive* grid techniques.

2.2 Adaptive Grid Methods

In the previous section we saw that the explicit method was able to capture the evolution of the solution profile but in order to properly capture the full solution, including the inner boundary layers, we require a large number of grid points, which subsequently penalises us in terms of the size of time step required. This playoff left us with a method that was capable but incredibly slow. To combat this we realised that the number of grid points required to capture the outer regions is fairly small, there is no large change in the solution structure here. We do, however, require resolution in and around the inner boundary layers, but since these move we cannot simply use a fixed non-uniform grid. This lead us to consider so-called *adaptive grid methods.*

We will present a brief overview of adaptive methods first, including types of adaptivity, equidistribution and MMPDEs, before giving detail of the particular method we have adopted to solve the CKS equation.

2.2.1 Overview of Adaptive Methods

Adaptive grid methods all share one common theme in that the grid, location or density, or aspects of it, such as interpolating polynomial (Finite Element Method), are not fixed throughout the numerical procedure. Instead these aspects are allowed to evolve or change in some prescribed manner throughout the solution procedure. By doing so, the aim is that these numerical methods have a better chance of accurately producing a numerical solution. Typical areas where fixed grid solutions struggle include boundary layers, shocks and ‘front tracking’, and so many of the methods devised have been developed with one or more of these types of event in mind. In the case of the CKS equation we have multiple moving boundary layers which, based on our fixed grid simulations, require an increased resolution of grid points versus the, respectively, more tame outer regions. In this case, and in many others, where sharp changes in the solution are present, fixed grid methods often fail to properly resolve the area unless a very high resolution grid is used. In particular, if these boundary regions become comparable to the size of the grid spacing there is a tendency for collapse of the numerical method. Adaptive methods aim to combat these types of instability by, for example, introducing more grid points in appropriate areas.

There are three main types of adaptivity and so we shall begin by briefly detailing these together with some common advantages and disadvantages of each before settling on the preferred method for our purpose.

2.2.2 Types of adaptivity

There are three main types of adaptive grid method, named *h-, p-* and *r-adaptivity.* Each method takes a different approach to the way in which they adapt to the problem being solved but in all three cases the number of grid points, the location of the grid points or the interpolating polynomial degree is not fixed. It should also be noted that combinations of the three methods can be applied in many cases, for example an *hp-adaptive* method, or indeed an *hpr-adaptive* method, if so required. As we shall see, the likelihood of certain combinations being used is unlikely or perhaps even unnecessary. We summarise the main features here for discussion, in all cases we don't specify the method by which we actually implement each method but simply give an idea of the approach.

h-adaptivity

Very briefly *h-adaptivity* can be described as a method which introduces(removes) grid points where we need more(less) resolution. More precisely, this method involves selecting an initial, perhaps non-uniform grid to perform the solution procedure on and fixing both the number of grid points and their locations. We then evolve the system until we feel that there are not enough(too many) grid points in a particular location, perhaps due to the emergence of a boundary layer or shock front, and then pause the system evolution. In the area(s) where we feel there is not enough grid points we insert more, increasing the total number, and interpolate the solution onto the new grid points. In this way we have increased the resolution in the affected areas and hope that by doing so we can avoid large errors in the solution due to grid spacing effects. The system then continues to evolve on this new grid with possible repeats of the procedure as necessary.

We also note that should an area have 'too many' grid points, for example a straight line section of the solution, we can remove points from the grid in these areas. In this way the adaptivity of the method comes from the varying number of grid points. The naming of this method is perhaps clear as we are adapting the grid spacing, commonly denoted h.

Advantages/Disadvantages

The advantages of this particular method lie in its ability to provide resolution to complex areas but since a rewrite of the grid requires an interpolation step and also for computational efficiency, we do not want to be updating the grid at every time step. Careful consideration must be made as to when to rewrite the grid. In problems which develop fixed boundary layers or shocks, for example, this method can be particularly effective as it has the ability to start on a coarse grid, which will be efficient, and then add in points in the appropriate region at a later time. Thus we aren't required to use a fine grid for the entire numerical simulation and so can expect a more efficient method in some cases. There is a play-off between grid rewrite efficiency/accuracy and a fixed fine grid. Fine grids require more computational time so the intention here is that by making use of occasional grid rewrites we can improve computational speed.

p-adaptivity

This type of adaptivity is essentially a type of finite element method where the adaptivity comes from varying the degree of the basis functions being used. In general one would select a set of basis functions all of the same order, often linear, and the proceed with the numerics. The p-adaptive method allows for this degree to vary on each element and therefore we can, for some desired reason, alter the degree as time evolves.

Advantages/Disadvantages

This method sees much use in various settings, such as electrostatics [**84**], and can have a significant improvement over sticking with one type of basis function or indeed fixing the basis. This method isn't in keeping with the finite difference method employed earlier but is included here for completeness.

r-adaptivity

This method can be described as a movement of the grid points to areas that require higher resolution. The method begins in the same manner as the other types and sets up an initial grid which may or may not be uniform. The key point here is that the number of grid points remains fixed throughout so we must ensure that we start with a 'reasonable' number of grid points. We now perform a single time step update of the nodal values. Simultaneously to the solution update we also move the grid points in a prescribed manner; for now we just assume that they move by some means to areas of interest. So, unlike the other types, we simply move the grid points into areas which require more resolution and since the number of points is fixed we have less in areas which require fewer grid points.

The evolution then proceeds in this way by repeating the whole process. Both the solution values at the grid points and the locations of the grid points themselves are updated at every time step. In this way we force grid points into areas which require more resolution and so we are better able to monitor any boundary layers or shocks, for example.

Advantages/Disadvantages

This method is particularly useful in dealing with moving boundary layers and shocks. Since the system can move the grid in step with areas of interest we expect to be able maintain a high level of accuracy in our solution. The method is also advantageous as it involves a fixed number of grid points so grid relations are fixed and it is often not necessary to use any interpolation. This method is generally much more capable at 'front tracking' in that it can continuously adapt the grid to take care of a moving front.

The main disadvantages of the method are a reduction in the computational speed associated with additional equations to deal with the movement of the grid and, particularly in higher dimensions, 'mesh tangling'. As far as speed is concerned it is expected, however, that in many problems the speed loss from additional equations will be negated by an overall increase in accuracy, and that since we generally require far fewer grid points than a fixed fine grid we do make significant computational improvements. Mesh tangling, whilst always an issue, is a real problem in 2 or more dimensions. We are only ever really dealing with one-dimension so this isn't a major drawback in our case.

2.2.3 Selection of *r-adaptivity* for CKS equation

For the equation we are considering, (1.1.1), and by considering the numerical simulations carried out using fixed grids detailed earlier, we now make the selection of an *r-adaptive* method from the choices listed above as our next approach. Given that the PDE displays multiple moving boundary layers which also appear and disappear over the course of evolution it is likely that this will be the most efficient method in our case. The *h-adaptive* method would require almost constant grid rewrites in order to track the moving boundary layers and further rewrites as these then disappear from the system, these additional overheads would appear to make the *h-adaptive* method computationally inefficient. Since we are using a standard finite difference approach the *p-adaptive* method is not appropriate and so we won't consider its use here.

2.2.4 Overview of Pure Equidistribution and Monitor Functions

Now that we have introduced the basic idea of the *r-adaptive* method we give some detail on the specific implementation of the method. First we give a brief overview of the equidistribution principle and the method it induces. This principle underpins much of the r-adaptive methodology and so it is worth noting the technique. We will only consider the theory in one spatial dimension as this is all we require presently, however, it should be noted that extensions to higher dimensions can be made. The idea of equidistribution essentially stems from an idea by de Boor for approximating numerical solutions with splines [**28**].

To start we must consider the *physical domain*, Ω_p, on which the PDE for $u(x,t)$, say, is to be solved, for simplicity of the argument we shall assume this domain is simply the unit interval and hence $x \in [0,1] =: \Omega_p$. Suppose now that we introduce a *computational domain*, Ω_c, with coordinate ξ and suppose, again for simplicity, that this domain is also the unit interval $\Omega_c := [0,1]$. The idea now is that we define a continuous, strictly increasing function $\Psi(\xi,t) : \Omega_c \times [0,\infty) \to \Omega_p \times [0,\infty)$ and that this map in fact describes the adaptivity of the grid. Assume for now that we know $\Psi(\xi,t)$, then if we uniformly partition the computational space into grid points ξ_i and then find the image of these ξ_i under Ψ we will have a collection of points $x_i(t)$ in the physical domain. These points form our adaptive grid in the physical domain, in particular we have $x_i(t) := \Psi(\xi_i,t)$. Note that by insisting the function is strictly increasing we preclude the possibility of mesh tangling, that is, points cannot switch ordering. What is left to consider is how we derive, interpret and understand the function $\Psi(\xi,t)$.

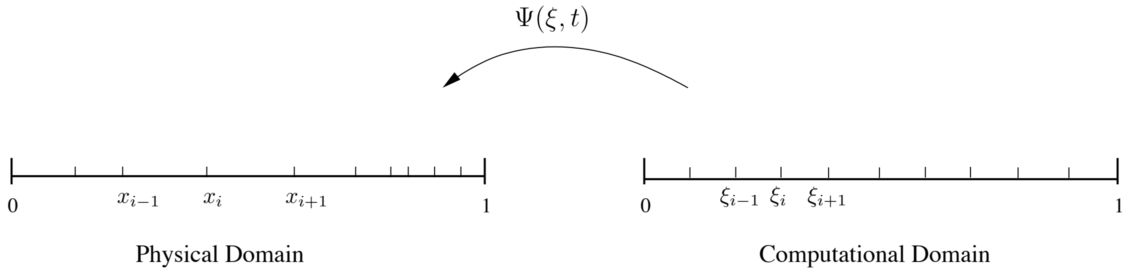


Figure 3.3 – Illustrative example of the physical domain, the computational domain and the associated map $\Psi(\xi,t)$ between the two. Note that the *uniform* grid in the computational domain is translated to a *non-uniform* grid in the physical domain.

As already mentioned we can interpret this mapping Ψ as the map which ‘distorts’ the domain in such a way so as to push grid points in the computational domain into specific areas in the physical domain as necessary. It is a time evolving function and this is done so that we can continuously adapt the grid as we see fit. In the example above, Figure 3.3, we note that the boundaries are fixed and, in general, we would rather not impose a 'twist' of sorts by inverting the boundaries, so we have the following constraints on Ψ,

$$\Psi(0,t) = 0 \quad \text{and} \quad \Psi(1,t) = 1, \quad \forall t \in [0,\infty).$$

In Figure 3.4 we see an example of the function Ψ at some time and have also placed a very coarse grid on the domain to show the effect on the resulting grid positions in the physical domain. Note that the function has fixed end points and is strictly increasing as demanded.

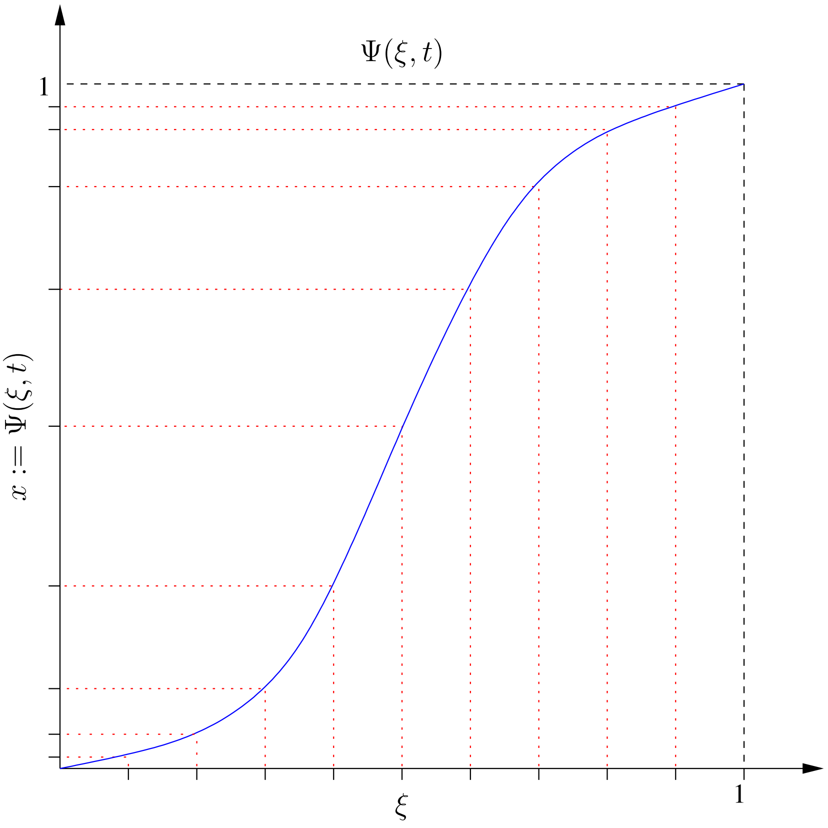


Figure 3.4 – Example of a transformation function Ψ showing the key features, namely, fixed end points and strictly increasing. Note that the values of the *uniform* grid in the computational coordinate, ξ, are transformed into a *non-uniform* grid in the physical, x, space.

Now that we have some idea of the function Ψ we proceed to the actual method of calculating the function and how it adapts with time. To achieve this we introduce the so-called *monitor function*, $M(x,t)$, on the physical domain. The choice of monitor function can have dramatic effect on the solution procedure and overall effectiveness of the approach. An example choice of monitor function could be arc length, i.e. $M(x,t) := \sqrt{1 + u_x(x,t)}$, although many other choices exist and it is often the case that the monitor must be tailored to suit the particular problem being solved, [**31**, **56**].

We then have the following definition, which is known as the *equidistribution principle*,

$$M(x,t)\Psi_{\xi}(\xi,t) = C(t), \quad \forall t \in [0,\infty), \tag{2.2.1}$$

where $C(t)$ is some constant at each time and, as before, $x := \Psi(\xi,t)$. Essentially we are insisting that the product of this monitor function at the point x and the ξ-derivative of the function $\Psi(\xi,t)$ at the inverse image of x is constant. This may seem to be arbitrary in nature but its use becomes clearer when we approximate the ξ derivative on the same computational grid as earlier to find the following,

$$M(x,t)\Psi_{\xi}(\xi,t) \approx M(x,t)\left(\frac{\Psi(\xi_{i+1},t)-\Psi(\xi_i,t)}{\Delta\xi}\right) = M(x,t)\left(\frac{x_{i+1}-x_i}{\Delta\xi}\right) = C(t). \tag{2.2.2}$$

At this point we note that the original definition of the equidistribution principle, (2.2.1), is defined at every point in the domain, however, now we have discretised the derivatives we must decide at which point in the subinterval $[x_i, x_{i+1}]$ the point x is located. Various modifications could be made at this stage, such as central differences at each grid point so that the monitor function would be defined at each grid point, however, we simply choose $X_i := x = (x_i + x_{i+1})/2$, i.e. the midpoint of the interval.

Taking the last part of (2.2.2) and defining $\hat{C}(t) = \Delta\xi C(t)$ we have,

$$\begin{gathered} M(X_i,t)(x_{i+1}-x_i) = \hat{C}(t), \\ M(X_i,t)\Delta x_i = \hat{C}(t), \quad \text{where,} \\ X_i = \frac{x_i + x_{i+1}}{2} \quad \text{and} \quad \Delta x_i = x_{i+1} - x_i. \end{gathered} \tag{2.2.3}$$

From here we see the immediate impact of the equidistribution principle on the moving mesh as where we have $M(X_i,t)$ ‘large’ we must have Δx ‘small’, and vice versa. In this manner the map Ψ, which is encoded in the statement (2.2.3), together with the associated monitor function determine the areas which require increased/decreased resolution in the mesh. This also explains the choice of the name *monitor function* as, in some sense, this function monitors the current numerical solution and ‘informs’ the function Ψ that the mesh needs to be adapted, equivalently the function Ψ needs to evolve.

In many circumstances the output using pure equidistribution can result in the grid moving in an abrupt manner, which isn’t ideal since this lack of smoothness can lead to jumps in the solution itself. Various attempts to smooth or otherwise limit the movement of the grid have been investigated, some examples include [**31**, **42**, **106**, **127**].

In practise, the implementation of the equidistribution method is much simpler than the detail above as we are only required to select a monitor function in advance of the simulations[2] and then ensure that at each time step the mesh is updated in keeping with the solution so as to maintain the condition in (2.2.3). Thus at each time step we must solve the original PDE and update the grid in the physical domain at the same time. The particular method we will adopt involves solving for the updated solution on the existing grid, then updating the grid and finally interpolating the new solution onto the new grid.

Along the same lines as smoothing the grid are so-called *MMPDEs*, or explicitly, Moving Mesh PDEs. These have a strong presence in the area of adaptive methods and so we give some detail of their origin and use. We also give brief detail on the so-called, *geometric conservation method* which is similar in nature to MMPDEs.

2.2.5 Overview of MMPDEs

MMPDEs [**55**, **56**], have their origins in the equidistribution principle, (2.2.1), and are essentially taking the principle and 'relaxing' the evolution of the mesh so as to avoid any 'jerkiness' or un-smoothness of the mesh evolution. The idea here is that it is not at all crucial to the numerical procedure that the mesh is perfectly equidistributed but instead we would be satisfied that it attempts to be close but also evolves in a smooth manner. For our purpose we will simply state several well known MMPDEs, Table 3.1, which are all derived by taking various temporal and spatial derivatives combined with well justified relaxation of the resulting equations. In doing so, various PDEs can be created each of which can be shown to increase the smoothness of the grid update whilst also closely resembling a perfectly equidistributed grid. More detail on these PDEs and significant analysis can be found in, for example, [**8**, **9**, **15**, **16**, **51**, **54**–**57**, **93**, **106**, **127**, **128**].

2.2.6 Geometric Conservation

Another possible approach, which closely resembles an MMPDE, is called geometric conservation [**4**, **20**]. This method considers the equidistribution principle, as in (2.2.1), but

[2] Choice of a monitor function is not always obvious. Fixed grid simulations or some other *a priori* information may be used to select an appropriate monitor.

MMPDE1	$\frac{\partial^2}{\partial\xi^2}(M\dot{\Psi}) = -\frac{\partial}{\partial\xi}\left(\frac{\partial M}{\partial t}\frac{\partial\Psi}{\partial\xi}\right)$
MMPDE2	$\frac{\partial^2}{\partial\xi^2}(M\dot{\Psi}) = -\frac{\partial}{\partial\xi}\left(\frac{\partial M}{\partial t}\frac{\partial\Psi}{\partial\xi}\right) - \frac{1}{\tau}\frac{\partial}{\partial\xi}\left(M\frac{\partial\Psi}{\partial\xi}\right)$
MMPDE3	$\frac{\partial^2}{\partial\xi^2}(M\dot{\Psi}) = -\frac{1}{\tau}\frac{\partial}{\partial\xi}\left(M\frac{\partial\Psi}{\partial\xi}\right)$
MMPDE4	$\frac{\partial}{\partial\xi}\left(M\frac{\partial\dot{\Psi}}{\partial\xi}\right) = -\frac{1}{\tau}\frac{\partial}{\partial\xi}\left(M\frac{\partial\Psi}{\partial\xi}\right)$
MMPDE5	$\dot{\Psi} = \frac{1}{\tau}\frac{\partial}{\partial\xi}\left(M\frac{\partial\Psi}{\partial\xi}\right)$
MMPDE6	$\frac{\partial^2\dot{\Psi}}{\partial\xi^2} = -\frac{1}{\tau}\frac{\partial}{\partial\xi}\left(M\frac{\partial\Psi}{\partial\xi}\right)$
MMPDE7	$\frac{\partial}{\partial\xi}\left(M\frac{\partial\dot{\Psi}}{\partial\xi}\right) - 2\frac{\partial}{\partial\xi}\left(M\frac{\partial\Psi}{\partial\xi}\right)\frac{\partial\dot{\Psi}}{\partial\xi}/\frac{\partial\Psi}{\partial\xi} = -\frac{1}{\tau}\frac{\partial}{\partial\xi}\left(M\frac{\partial\Psi}{\partial\xi}\right)$

Table 3.1 – Table of common MMPDEs together with their numbering as in [**56**]. τ is called the '*relaxation time*' and its value affects how quickly the mesh approaches equidistribution.

we adjust the definition of the monitor so that it is normalised, in other words the integral of the monitor over the physical domain is constant and hence $C(t)$ is constant. We proceed by taking the derivative with respect to time. We then have,

$$\begin{aligned} M_t(x,t)\Psi_\xi(\xi,t) + M_x(x,t)\Psi_t(\xi,t)\Psi_\xi(\xi,t) + M(x,t)\Psi_{\xi t}(\xi,t) &= 0 \\ (M_t + (MV)_x)\Psi_\xi &= 0, \end{aligned}$$

where $V = \Psi_t$ and we have suppressed arguments in the last line. This expression V is the velocity of the grid which we aim to find as this will tell us where the points of the grid need to be at the next time step. Since we know $\Psi_\xi > 0$, a constraint on the function as defined, we must have,

$$M_t + (MV)_x = 0, \tag{2.2.4}$$

which is now the equation solved to find V. Once we have calculated the velocity of the grid points we can very simply update the grid and then repeat the whole procedure.

We now proceed to discuss the particular method we have chosen to solve the CKS equation, (1.1.1), and in particular mention slight modifications made to account for periodicity. Further to this we will give detail of the monitor function chosen, which is based on our understanding from the fixed grid simulations.

2.2.7 Choice of Geometric Conservation

There are several methods which we could have adopted for updating the numerical grid underlying the numerical solution. For our purposes here we have chosen to employ the geometric conservation approach. This approach maintains the same advantage as the various MMPDE in that it creates a smoother update of the numerical grid but we feel it has a slight advantage in that we need not specify a relaxation time τ for the grid. Ultimately finding a satisfactory output is the goal and we feel that this approach has a good chance of obtaining that. Numerically solving (2.2.4) proceeds as follows; first we simply rearrange the equation,

$$(MV)_x = -M_t,$$

before then integrating both sides with respect to the dummy variable $\bar{x}$ between 0 and x,

$$M(x,t)V(x,t) = -\int_0^x M_t(\bar{x},t)\, d\bar{x} + C,$$

where C is a constant. If we now divide through by $M(x,t)$ we then have the following expression for $V(x,t)$,

$$V(x,t) = -\frac{\displaystyle\int_0^x M_t(\bar{x},t)\, d\bar{x} + C}{M(x,t)}.$$

This is the basic form of equation which we will solve numerically to find the function V. In order to characterise the constant C we must understand the effect that having a periodic domain has on the adaptive grid method.

2.2.8 Accounting for Periodicity

Everything we have discussed so far has assumed that we are on a fixed domain of some size, L, say. For our purposes we wish to consider a periodic domain since this is more appropriate for the CKS problem. We still have a domain of some fixed size L but we now have periodic boundary conditions on the domain and as a result, since we must perform integrals over the domain during the procedure, we are forced to privilege a point in the domain, the initial point for the integrals. This is straightforward in the code since we simply select the point with index 0, x_0, as the start of the domain. With this point selected we see that the function Ψ is defined from this point which corresponds to the point ξ_0 in the fixed computational domain, and so we must be strictly increasing from this point at all times and then the function would have a discrete jump at the periodicity point. This presents no issues to the implementation of the method but is more of a subtle change.

We now proceed to characterise the constant C. One method of figuring out the value of C is to fix one physical grid point for all times, i.e. to set $V_0 = 0$, and use this to find that $C = 0$. This is not really appropriate since we must select one point to fix, which is somewhat arbitrary in choice, and so we have placed some unwanted rigidity on the grid.

To remove this rigidity we instead insist that the integral of V over the domain x is 0,

$$\int_0^L V(x,t)\, dx = 0,$$

i.e. the average velocity is zero. This will allow us to deduce a value for C which releases all the points in the grid to adapt and also, usefully, ensures we don't have any unnecessary drift of all the points. Writing the velocity formula in the following manner allows us to simplify the approach,

$$V(x,t) = -\hat{V}(x,t) + \frac{C}{M(x,t)},$$

where,

$$\hat{V}(x,t) = \frac{\displaystyle\int_0^x M_t(\bar{x},t)\, d\bar{x}}{M(x,t)}.$$

Now if we take the integral with respect to x on both sides we have,

$$0 = -\int_0^L \hat{V}(x,t)\, dx + C\int_0^L \frac{1}{M(x,t)}\, dx,$$

and hence,

$$C = \frac{\displaystyle\int_0^L \hat{V}(x,t)\, dx}{\displaystyle\int_0^L \frac{1}{M(x,t)}\, dx}.$$

The code can numerically calculate each of these integrals and then set each grid point's velocity, V_i. We can then update the grid at each time step using the appropriate velocity.

2.2.9 Monitor Choice

The one remaining ingredient necessary before we start simulations is a monitor function. Selection of a monitor function could be considered a bit of an art form since there is no universal choice, instead we must be guided by information on what we expect to happen from either analysis or perhaps simple fixed grid simulations. It is the latter which we mainly rely on in our case since we performed a significant number of fixed grid simulations before settling on using an adaptive grid method. These simulations together with the analysis earlier and in [**39**, **89**] led us to believe that the solution forms large parabolic type regions with essentially fixed curvature joined by small areas of high curvature. Since these small areas are those which caused the fixed grid so much difficulty we choose a monitor function which takes the curvature of the solution at each grid point, κ_i, and then calculates the following quantity as the monitor value,

$$M(X_i, t) = \sqrt{1 + \kappa_i^2}.$$

Since the curvature in the parabola type regions is generally fixed this will have a similar contribution to overall monitor whereas the curvature in the small regions is much higher and so will contribute more to the monitor. Loosely this means that since the monitor is higher in these small regions the grid wants to have more points here to reduce the grid step size there. This is precisely what we would like to happen and so we see the benefit of this monitor choice.

Note that in order to make use of the geometric conservation method we must make use of a normalised monitor function. This is easy to implement since we simply calculate these monitor values at each of the points X_i, the mid points of the intervals, then calculate the integral over the domain, $M_{tot}(t)$ say, and finally actually make use of value $M_i(x,t)/M_{tot}(t)$ as the monitor value at each of the points. Doing so makes sure we are in an appropriate position to use the geometric conservation method and so are safe to proceed.

2.2.10 Update procedure

With the monitor now set we give some brief detail of the actual procedure used to update the solution at each time step. To begin we take the PDE which we aim to solve, in this case the CKS equation, (1.1.1), and take a single Euler time step of size Δt on the current grid. Using this new solution we then calculate the monitor values, $M_i := M(X_i, t)$, and calculate each of the associated velocities for the grid points V_i. Then using the same time step we update the underlying grid again using a simple Euler update with the same time step size, to find the new position of the grid points. We then take the new solution and use linear interpolation to map the new solution onto the new grid points. We then repeat this loop for as long as we require.

Since we are no longer on a uniform grid we must redefine the four finite difference operators from the explicit method. In all four cases we make use of a standard 5-point stencil, that is, we use the nodal value plus the two values to the left and two to the right. Hence we have the following standard operator,

$$(D_k\mathcal{U})_i^j := (A_k)_i^j\mathcal{U}_{i-2}^j + (B_k)_i^j\mathcal{U}_{i-1}^j + (C_k)_i^j\mathcal{U}_i^j + (D_k)_i^j\mathcal{U}_{i+1}^j + (E_k)_i^j\mathcal{U}_{i+2}^j.$$

We must then define each of the coefficients in the expression for $k \in \{1, 2, 3, 4\}$. We make use of the various grid spacings which are indexed based on the distance they represent, for example h_1 is the distance between x_i and x_{i+1}, h_{-1} the distance between x_i and x_{i-1} and so on. The coefficients of each term are as follows,

$$
\begin{aligned}
k=1: \quad (A_1)_i^j &:= \frac{h_{-1}h_1h_2}{h_{-2}(h_{-2}+h_1)(h_{-2}+h_2)(h_{-2}-h_{-1}),} \\
(B_1)_i^j &:= -\frac{h_{-2}h_1h_2}{h_{-1}(h_{-1}+h_2)(h_{-1}+h_1)(h_{-2}-h_{-1})}, \\
(C_1)_i^j &:= \frac{h_{-1}h_1h_2+h_{-2}h_1h_2-h_{-2}h_{-1}h_2-h_{-2}h_{-1}h_1}{h_{-2}h_{-1}h_1h_2}, \\
(D_1)_i^j &:= \frac{h_{-2}h_{-1}h_2}{h_1(h_{-1}+h_1)(h_{-2}+h_1)(h_2-h_1)}, \\
(E_1)_i^j &:= -\frac{h_{-2}h_{-1}h_1}{h_2(h_{-1}+h_2)(h_{-2}+h_2)(h_2-h_{-1})},
\end{aligned}
$$

$$
\begin{aligned}
k=2: \quad (A_2)_i^j &:= -\frac{2(h_{-1}h_1+h_{-1}h_2-h_1h_2)}{h_{-2}(h_{-2}+h_1)(h_{-2}+h_2)(h_{-2}-h_{-1})}, \\
(B_2)_i^j &:= \frac{2(h_{-2}h_1+h_{-2}h_2-h_1h_2)}{h_{-1}(h_{-1}+h_2)(h_{-1}+h_1)(h_{-2}-h_{-1})}, \\
(C_2)_i^j &:= \frac{2(h_{-2}h_{-1}-h_{-2}h_1-h_{-2}h_2-h_{-1}h_1-h_{-1}h_2+h_1h_2)}{h_{-2}h_{-1}h_1h_2}, \\
(D_2)_i^j &:= -\frac{2(h_{-2}h_{-1}-h_{-2}h_2-h_{-1}h_2)}{h_1(h_{-1}+h_1)(h_{-2}+h_1)(h_2-h_1)}, \\
(E_2)_i^j &:= \frac{2(h_{-2}h_{-1}-h_{-2}h_1-h_{-1}h_1)}{h_2(h_{-1}+h_2)(h_{-2}+h_2)(h_2-h_{-1})},
\end{aligned}
$$

$$
\begin{aligned}
k=3: \quad (A_3)_i^j &:= \frac{6(h_{-1}-h_1-h_2)}{h_{-2}(h_{-2}+h_1)(h_{-2}+h_2)(h_{-2}-h_{-1})}, \\
(B_3)_i^j &:= -\frac{6(h_{-2}-h_1-h_2)}{h_{-1}(h_{-1}+h_2)(h_{-1}+h_1)(h_{-2}-h_{-1})}, \\
(C_3)_i^j &:= \frac{6(h_{-2}+h_{-1}-h_1-h_2)}{h_{-2}h_{-1}h_1h_2}, \\
(D_3)_i^j &:= -\frac{6(h_{-2}+h_{-1}-h_2)}{h_1(h_{-1}+h_1)(h_{-2}+h_1)(h_2-h_1)}, \\
(E_3)_i^j &:= \frac{6(h_{-2}+h_{-1}-h_1)}{h_2(h_{-1}+h_2)(h_{-2}+h_2)(h_2-h_{-1})},
\end{aligned}
$$

$$
\begin{aligned}
k=4: \quad (A_4)_i^j &:= \frac{24}{h_{-2}(h_{-2}+h_1)(h_{-2}+h_2)(h_{-2}-h_{-1})}, \\
(B_4)_i^j &:= -\frac{24}{h_{-1}(h_{-1}+h_2)(h_{-1}+h_1)(h_{-2}-h_{-1})}, \\
(C_4)_i^j &:= \frac{24}{h_{-2}h_{-1}h_1h_2}, \\
(D_4)_i^j &:= -\frac{24}{h_1(h_{-1}+h_1)(h_{-2}+h_1)(h_2-h_1)}, \\
(E_4)_i^j &:= \frac{24}{h_2(h_{-1}+h_2)(h_{-2}+h_2)(h_2-h_{-1})}.
\end{aligned}
$$

As we simply make use of the current grid set up we can simply calculate all of these values and then update the associated grid point using a modified version of (2.1.13), namely,

$$\mathcal{U}_i^{j+1} = \mathcal{U}_i^j - \Delta t \left[(D_2\mathcal{U})_i^j + (D_4\mathcal{U})_i^j + 2\left(\left((D_2\mathcal{U})_i^j \right)^2 + (D_1\mathcal{U})_i^j (D_3\mathcal{U})_i^j \right) \right]. \qquad (2.2.5)$$

Once we have updated the nodal values we then perform the linear interpolation step to place the new solution on the new grid.

2.2.11 Discussion of Adaptive Method

We comment on two aspects of this method, first the size of time step used and some reasoning behind this followed by a look at the code's ability to mimic certain aspects of the solution structure which we anticipate to be true.

Time stepping

Unlike the previous explicit-fixed grid method it is harder for us to say something conclusive about the size of time step that must be used. We do, however, make one particular comment on the time step used. As the update mechanism is essentially an explicit rule we expect it to behave in a similar manner to the fixed grid simulations. Indeed, if the grid were forced to be static and uniform it would be identical to the explicit method. With this in mind we expect the adaptive method to have a similar time step restriction to the explicit method, namely,

$$\Delta t \leqslant \frac{h^4}{8}.$$

We expect that at the local level spatially, the areas with small grid spacing must be carefully resolved, since these areas are considered to be areas of interest, and so we anticipate that the time step must reflect the smallest grid spacing present, h_{min}. Taking this into account we expect the time step to be adaptive and restricted in size by the expression,

$$\Delta t \leqslant \frac{h_{min}^4}{8}.$$

Numerically we found this restriction to indeed be applicable and generally chose the time step $\Delta t = h_{min}^4/10$ for our simulations.

Accuracy

We consider the ability of the code to capture the essential features of the evolution and we consider one particular aspect as a check. As mentioned earlier, once the system settles into the parabolic structures we should see a fixed second derivative value of $-1/2$ in the parabolic regions. To check this outcome we take a domain of size $L = 10$ and seed with a simple initial condition which should form, in this case, a single parabola. We then plot the second derivative values once the profile has settled into this shape. We ran three simulations with 40, 50 and 60 grid points respectively and plot the second derivative values in the parabolic region, each at the same physical time, shown in Figure 3.5.

Worryingly for this method we see no evidence of a relationship between the number of points and computational accuracy. Increasing the number of points to 60 appears to make the outer structures less like the $-1/2$ structures than we expect and strangely further from this mark than for 50 points. In all three cases, however, there is quite a discrepancy between the predicted value and the actual numerical value. This discrepancy combined with the ever increasing computational disadvantage of increasing the number of points led us to believe that this method is incapable of providing accurate results in reasonable time frame. Whilst it may be possible to implement an adaptive method for solving the CKS equation we have been unable to do so here.

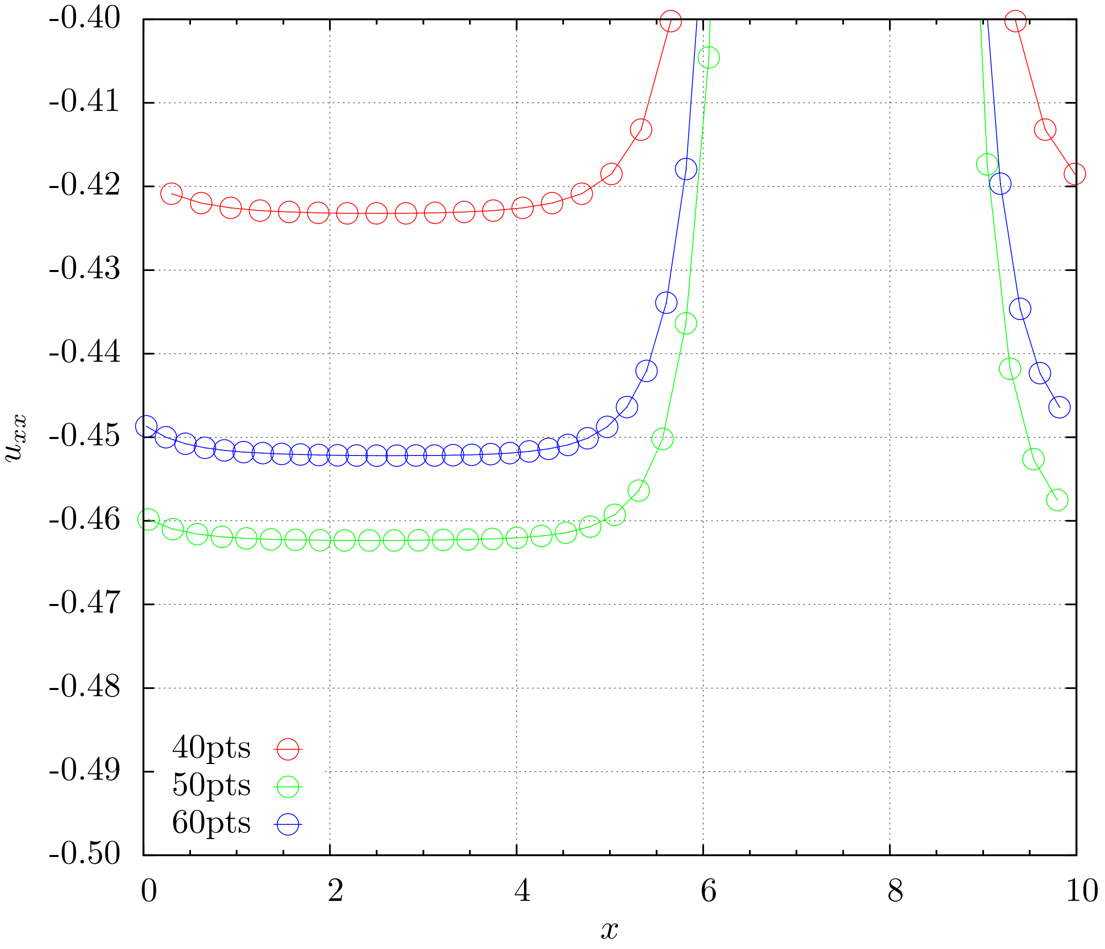


Figure 3.5 – Adaptive simulation of the CKS equation on a domain of size $L = 10$ with varying numbers of grid points. Accuracy varied inconsistently with an increase in the number of points and additionally computational time is adversely affected as we increase the number of points.

2.3 Implicit-Explicit Update - Fixed Grid

Following on from the adaptive method and the problems encountered there, we have chosen to return to a fixed grid system. Since the inner boundary layers translate within the domain it isn't particularly feasible to create a non-uniform grid which will always have an appropriate number of grid points in the right areas, in particular with only one grid used throughout the simulation we run the risk of over resolving some regions whilst potentially being deficient in others. As a result we instead make use of another fixed, uniform grid but aim to improve in other ways on the explicit method shown in Section 2.1. Fundamentally the initial set up is the same as the explicit method, however, it differs in the method with which we update the nodal values. The main limiting factor in the explicit scheme was the small time steps which were required for the system to be numerically stable, such a constraint meant that the method was not efficient for this particular problem. The discretisation of the derivatives, on the other hand, was perfectly adequate for our purposes and so these aspects of the method will be left untouched. First let us return equation (2.1.15) which states,

$$\mathbf{U}^{j+1} = \left(I - \frac{\Delta t}{h^4}A\right)\mathbf{U}^j - \frac{2\Delta t}{h^4}\mathbf{F}(\mathbf{U}^j), \tag{2.3.1}$$

For purely linear PDEs there are two common types of solution method; explicit and implicit. Explicit methods are essentially the same as found in Section 2.1 earlier and update nodal values making exclusive use of the previous nodal values, i.e. use the t_j values to calculate t_{j+1} values. Such methods can also be readily used for non-linear PDEs, which is precisely what we performed earlier. Implicit methods are different in that they express all the spatial finite differences in terms of the updated values which we wish to find, i.e. express spatial derivatives at the t_{j+1} level instead of the t_j level and then solve the system of equations to find the nodal values at the new, updated time. If we take the linear terms in the CKS equation, namely u_{xx} and u_{xxxx}, and use an implicit type scheme instead, which means expressing the derivatives, contained in A, in terms of the updated values, we then have a new expression,

$$\mathbf{U}^{j+1} = \mathbf{U}^j - \frac{\Delta t}{h^4}A\mathbf{U}^{j+1} - \frac{2\Delta t}{h^4}\mathbf{F}(\mathbf{U}^j).$$

Note that the term involving A on the right hand side, which represents the linear terms, is at the updated t_{j+1} time. If we now rearrange this equation to place t_{j+1} values to the left and t_j values to the right we have,

$$\left(I + \frac{\Delta t}{h^4} A\right) \mathbf{U}^{j+1} = \mathbf{U}^j - \frac{2\Delta t}{h^4} \mathbf{F}(\mathbf{U}^j). \tag{2.3.2}$$

This will form the basis for our implicit-explicit method. If we consider the right hand side of equation (2.3.2) we see that as we proceed to try and update the nodal values the entire right hand side is made up of explicit contributions due to the non-linear nature of the terms of the original equation there, hence, they could not be easily treated implicitly. These values can all be calculated since they only require information at the current time.

Note that this expression captures the crucial point that the right hand side is at the 'old' time and the left hand side the 'new', updated time. This matrix equation is what we use to update the system. The vector of right hand side values is calculated first using the current nodal values. The matrix $(I + (\Delta t/h^4)A)$ is fixed throughout the entire simulation and so we are presented with a choice of methods for solving for $\mathbf{U}_{j+1}$. The cyclic pentadiagonal structure of the matrix admits efficient inverse procedures to be carried out. Since the matrix is fixed throughout the simulation we are only required to calculate this inverse at program start-up and then at each time step simply use matrix multiplication to find,

$$\mathbf{U}^{j+1} = \left(I + \frac{\Delta t}{h^4} A\right)^{-1} \left[\mathbf{U}^j - \frac{2\Delta t}{h^4} \mathbf{F}(\mathbf{U}^j)\right]. \tag{2.3.3}$$

Another possibility is to use an iterative solver whereby we leave the equation as in (2.3.2) and solve by repeated use of an iterative algorithm. Many such solvers exist and often these are very efficient, in this case, however, we generally found it to be just as efficient to perform one inversion at start-up and use this throughout, as in (2.3.3), so it is this method which we shall adopt. The method then proceeds much like the explicit method in that we initialise using the specified initial condition and then march forward one step at a time.

2.3.1 Discussion of Implicit-Explicit Method

We first show that by consideration of the linear parts of the equation this method has no restriction on the size of time step used. Following on from this we comment on the influence of the nonlinear parts and what effect those have on the time stepping. Finally we give some brief comments on this method in comparison to the others as well as giving some ideas of possible extensions and improvements.

Time stepping

We follow a very similar analysis to that which was performed on the explicit method in Section 2.1.1 but this time note that we need to instead consider the matrix,

$$\left(I + \frac{\Delta t}{h^4} A\right).$$

Note that we don't immediately consider the inverse but we shall return to this point later. In a similar manner to before we have the following quantities,

$$\omega = e^{\frac{2\pi i}{n}},$$
$$c_0 = 1 + \frac{\Delta t}{h^4}(6 - 2h^2), \quad c_1 = \frac{\Delta t}{h^4}(h^2 - 4), \quad c_2 = \frac{\Delta t}{h^4}$$
$$c_{n-1} = \frac{\Delta t}{h^4}(h^2 - 4) \quad \text{and} \quad c_{n-2} = \frac{\Delta t}{h^4},$$

from which we can write down our expression for the eigenvalues,

$$\lambda_j = 1 + \frac{\Delta t}{h^4}(6 - 2h^2) + \frac{\Delta t}{h^4}(h^2 - 4)(\omega^j + \omega^{-j}) + \frac{\Delta t}{h^4}(\omega^{2j} + \omega^{-2j}), \quad j \in \{0, 1, \dots, n-1\}.$$

Simplifying as was done previously we then have,

$$\lambda_j = 1 + \frac{\Delta t}{h^4}\left[6 + 2\cos\left(\frac{4\pi j}{n}\right) - 8\cos\left(\frac{2\pi j}{n}\right)\right] + \frac{\Delta t}{h^2}\left[2\cos\left(\frac{2\pi j}{n} - 2\right)\right].$$

If we once again consider only the dominant part we are left with the following,

$$\lambda_j \approx 1 + \frac{\Delta t}{h^4}\left[6 + 2\cos\left(\frac{4\pi j}{n}\right) - 8\cos\left(\frac{2\pi j}{n}\right)\right].$$

This has minimum value 1 when $j = 0$ or $j = n$ and for all other values of j this is strictly greater than 1. Therefore all the eigenvalues of this matrix can be considered to be larger than 1 and so since we actually require the inverse matrix,

$$\left(I + \frac{\Delta t}{h^4} A\right)^{-1},$$

we can easily calculate each of the eigenvalues of the inverse matrix as λ_j^{-1} and since each $\lambda_j \geqslant 1$ we immediately have that $\lambda_j^{-1} \leqslant 1, \forall j$. Thus the linear parts of the equation provide no restriction on the time step used, it is unconditionally stable.

Modification of Δt to account for nonlinear parts

Whilst this analysis of the linear parts provides no restriction on the size of the time step used our simulations note that we must still take care in the selection of the time step used. Primarily this is due to the nonlinear parts of the equation. It is harder to exactly quantify any such restriction from the nonlinear parts analytically but from our simulations we note that a time step of at most $h^2/5$ together with a small enough grid spacing seemed to be sufficient to ensure the system does not collapse at some later time. This is clearly a much less severe restriction compared to the explicit method which forced us to consider a time step proportional to h^4. Nevertheless we must take care to ensure the code remains consistent for a specified time step. More detail on this aspect can be found in Section 3.5.

Possible improvements or modifications

This method has been found to be sufficient for our purposes and we shall probe the code's ability in Section 3. Here, for completeness, we give some brief detail on possible improvements that could be made to this method, possibly in conjunction with the previous methods.

One possible modification would be to take all the terms of the equation and treat then implicitly. A simple matrix solve would no longer be appropriate but if we had an equation of the form,

$$\left(I + \frac{\Delta t}{h^4} A\right) \mathbf{U}^{j+1} + \frac{2\Delta t}{h^4} \mathbf{F}(\mathbf{U}^{j+1}) - \mathbf{U}^j = 0,$$

we may be able to then use a fixed point type argument to update the system completely implicitly.

Perhaps the most interesting improvement would be to construct an implicit adaptive scheme whereby we have a large system of moving grid points and solution values, much like the adaptive method earlier, and solve for all the variables simultaneously but this time implicitly. This type of method appears completely plausible and it seems likely that it could be both fast and accurate, a best of both worlds approach.

3 Benchmarking the Code

We now proceed to check the performance of the selected implicit-explicit code against various aspects of the solution which have been analytically derived from the original equation. We begin by checking the instability of the zero state by inputting a very small perturbation to the zero state and considering the growth of the solution profile, i.e. does it grow into the expected solution structure. Following on from this we check that the code forms the expected two-scale solution with large outer regions connected by small inner layers. We then check the predicted structure of both the outer and inner layers against predicted results before finally commenting on the time step used for simulations.

3.1 Linear stability

We first consider the linear stability of the CKS equation and once calculated we verify the outcome using the code. First let us take the CKS equation and consider only the linear parts, this leaves us with the following,

$$u_t = -u_{xx} - u_{xxxx}.$$

We take the standard ansatz solution,

$$u(x,t) = e^{ikx}e^{\sigma t},$$

where $k \in \mathbb{R}^+$, the positive real line, and $\sigma \in \mathbb{C}$, and plug this into the linearised CKS equation. The term σ controls the fate of an initial perturbation with wave number k. If $\mathrm{Re}(\sigma) < 0$ then these contributions shrink in time. Conversely, if $\mathrm{Re}(\sigma) > 0$ then these contributions grow in time. Taking appropriate derivatives of this ansatz we then have,

$$\begin{aligned} u_t &= \sigma e^{ikx}e^{\sigma t}, \\ u_{xx} &= -k^2 e^{ikx}e^{\sigma t}, \\ u_{xxxx} &= k^4 e^{ikx}e^{\sigma t}. \end{aligned}$$

If we now insert these into the linearisation of the CKS equation we have,

$$\sigma e^{ikx}e^{\sigma t} = e^{ikx}e^{\sigma t}(k^2 - k^4),$$

or equivalently,

$$e^{ikx}e^{\sigma t}(\sigma - k^2 + k^4) = 0.$$

From which we can read off the *dispersion relation*,

$$\sigma = k^2(1-k^2). \tag{3.1.1}$$

Note in our case that σ is in fact a real number. We are mainly concerned with which values of k give growth in time and so recognise that we are restricted to the region $k \in (0,1)$. With this in mind we can also calculate which wave number will grow fastest in time, given by the maximum value of σ, which can easily be calculated as being at $k_{max} = 1/\sqrt{2}$. We can now consider the wavelength which will grow the most rapidly in time by considering the trigonometric expansion and focussing on the sine wave. The wavelength will be of size $L = 2\pi/k_{max}$, which in this case means the most excited wavelength is of size, $L = 2\sqrt{2}\pi$. It is clear that the fully non-linear problem will not exactly satisfy this condition, however, we expect there to be a bias towards structures of this size at least initially and as a result we expect to see roughly one parabola for every nine to ten spatial units emerge.

We can easily test this stability condition in our code by initialising several simulations on different domain lengths with very small perturbations to the zero state and considering the output in time. We show one such example here and note that in the next section where we discuss more on the emergence of solution structure that, in fact, both examples there also provide good insight into the codes ability to mimic this instability.

The example which we shall use here is a simple linear sum of sine waves as an initial condition on a domain of size $L = 20$. We select forty numbers, a_k, at random from a uniform distribution on $[0,1]$ and then multiply each by 10^{-12}. Our initial condition is then,

$$u(x,0) = \sum_{k=1}^{40} a_k \sin\left(\frac{2\pi k x}{L}\right). \tag{3.1.2}$$

In Figure 3.6 we have plotted the solution profile on both a fixed and variable vertical scale and have selected a few snapshots during the time period $t \in [0,120]$ as representative of the initial growth. As expected the initial data rapidly grows until around the end of the selected time period where we see the solution has settled into the expected profile of large scale sections separated by small connecting regions. Note that, as we anticipated, the size of the structures is around ten, a good agreement with the analysis presented earlier. We consider the evolution after initial instability in the next section.

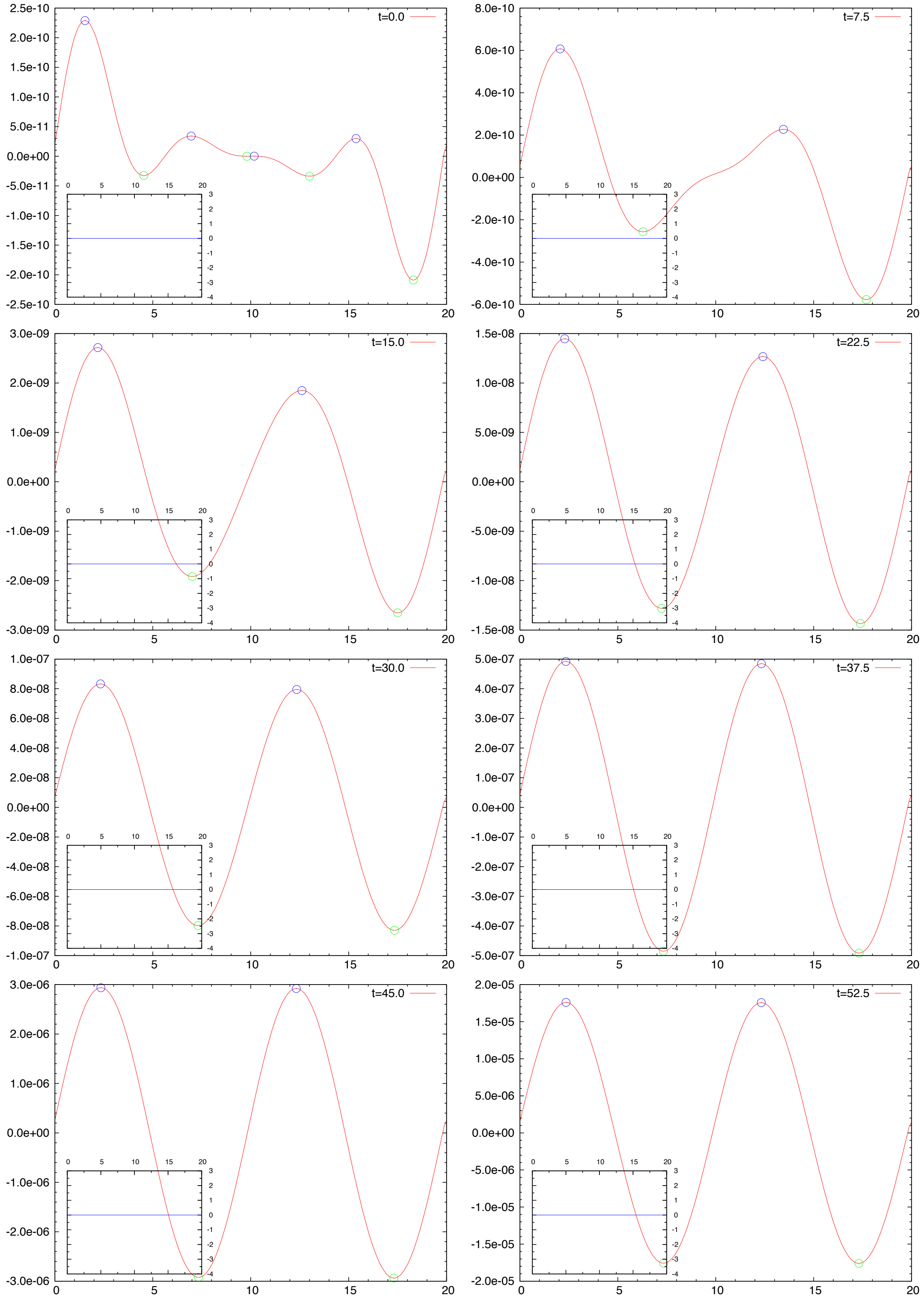


Figure 3.6 – Selected output of solution profile depicting linear instability of the CKS equation. The initial condition is a simple combination of sine waves with varying small amplitude, (3.1.2). Inset diagrams show same profile but on a fixed vertical scale showing the massive growth of the solution. (Continued over page.)

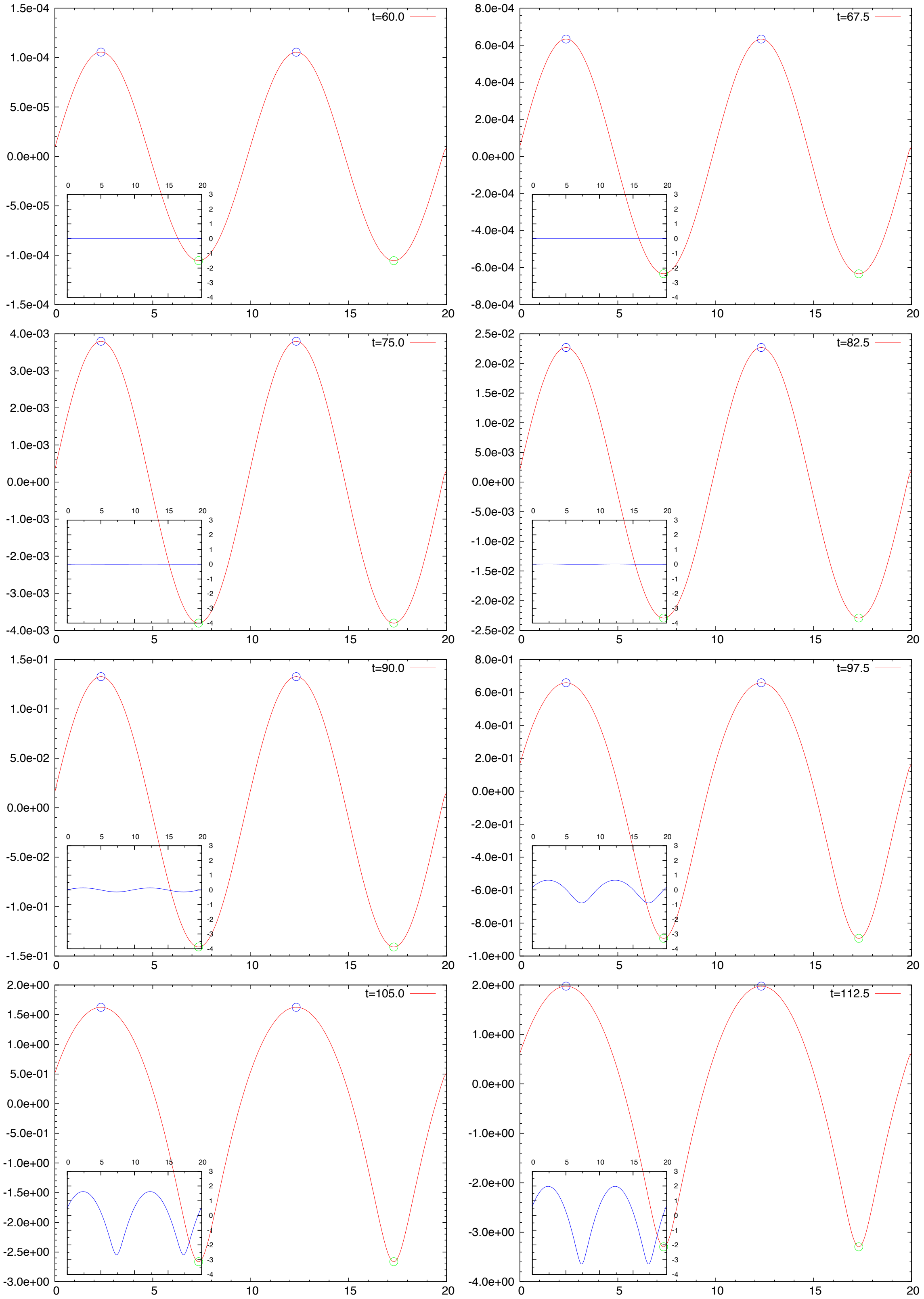


Figure 3.6 Continued – Continuation from previous page showing further into the same simulation.

3.2 Emergence of Two-Scale Solution and Coarsening

The early-time profiles generated by the code have a length scale that agrees with the linear stability analysis. At later times we expect to see the emergence of a two-scale solution with larger outer regions, which should be parabolic in nature, connected by small inner regions which connect the larger structures. Once the solution profile settles into this regime we then expect to see the profile coarsen in time, that is, the number of parabolas should decrease in time and those which are left will grow to compensate. Larger parabolas will appear to eat neighbouring smaller parabolas until ultimately the system is reduced to one large parabolic structure and one inner boundary layer. We again highlight two simulations, from the many carried out, to illustrate the ability of the code to match this expected result, one on a small domain and one on a larger domain.

First we take a domain of size $L = 40$ together with the initial condition,

$$u(x,0) = 0.0001\left(1.0 + e^{-\frac{(x-0.75L)^2}{2}}\right)\sin\left(\frac{\pi x}{L}\right)\sin\left(\frac{2\pi x}{L}\right). \tag{3.2.1}$$

In Figure 3.7 we can see snapshots of the evolution and can clearly see the emergence of a two-scaled solution, large parabola like regions connected by small inner matching regions. This simulation is also another good example of the linear instability of the zero state from the previous section. The initial condition has very small initial minimum and maximum but these grow rapidly until the solution forms the expected two-scale structure as seen at later times.

Once the solution has settled into the two-scaled regime we then see the appearance of the other aspect of the system which we expect to observe, system coarsening. At around $t = 60$ the solution has formed four parabolic type structures, predicted by the linear stability analysis, and we then start to see the smaller two of these begin to shrink whilst the other two grow. In this simulation there are two parabolas which, whilst not identical in size, are smaller in comparison to others and then proceed to shrink and disappear. Following on from this, in the latter half of Figure 3.7, the larger of the two remaining parabolas grows whilst the other shrinks until ultimately we are left with one large structure as we'd expect. This simple example seems to confirm the accuracy of the code, certainly at least in this respect.

The second simulation we have to illustrate the growth into a two-scaled solution is on the larger domain of size $L = 100$ using the initial condition,

$$u(x,0) = 0.0001\sin\left(\frac{\pi x}{L}\right)\sin\left(\frac{2\pi x}{L}\right). \tag{3.2.2}$$

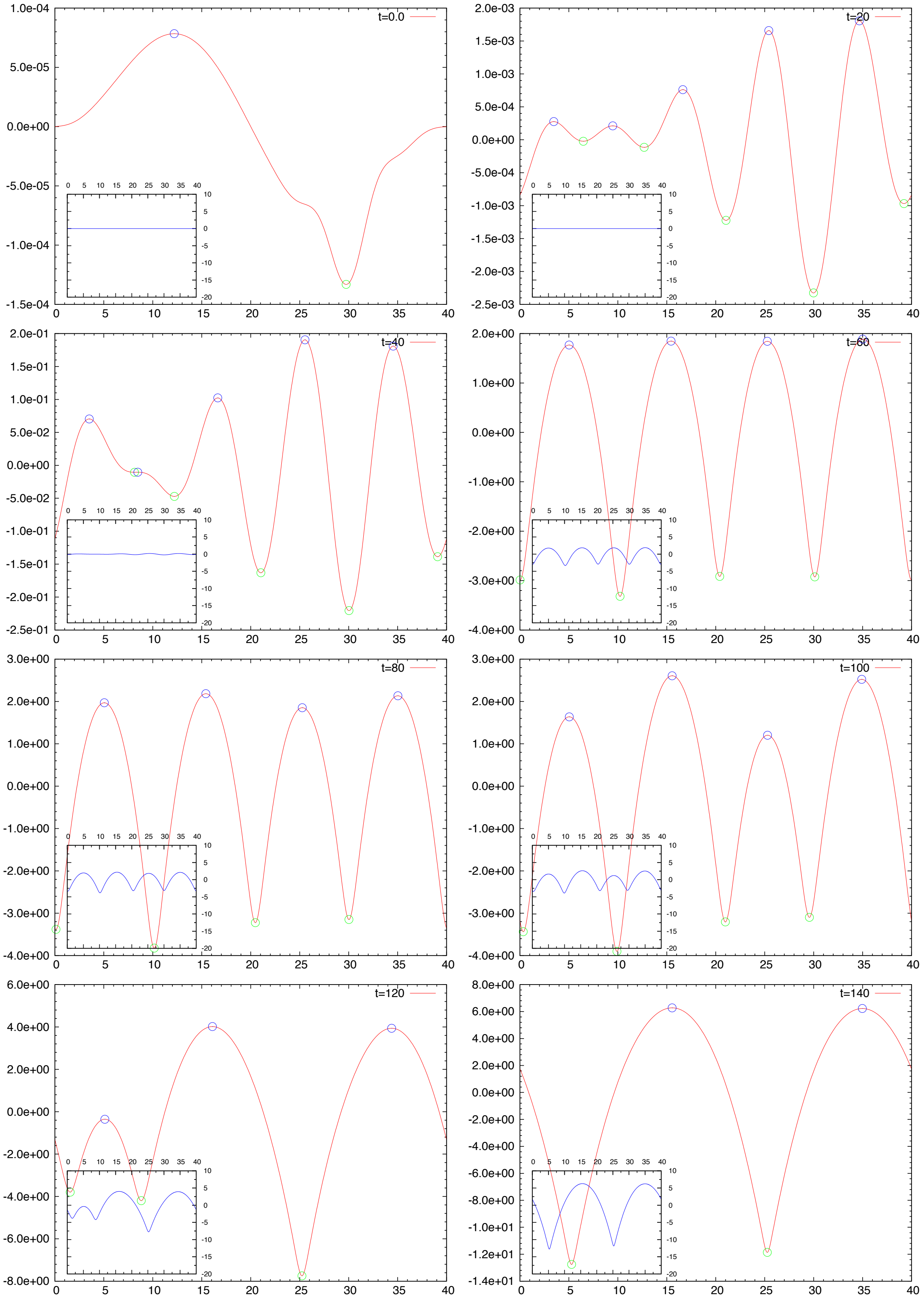


Figure 3.7 – Early evolution of CKS solution on domain of size $L = 40$ using the initial condition shown in (3.2.1), $u(x,0) = 0.0001\left(1.0 + e^{-\frac{(x-0.75L)^2}{2}}\right)\sin\left(\frac{\pi x}{L}\right)\sin\left(\frac{2\pi x}{L}\right)$, showing emergence of two-scale structure and then the coarsening of the solution profile. (Continued over page.)

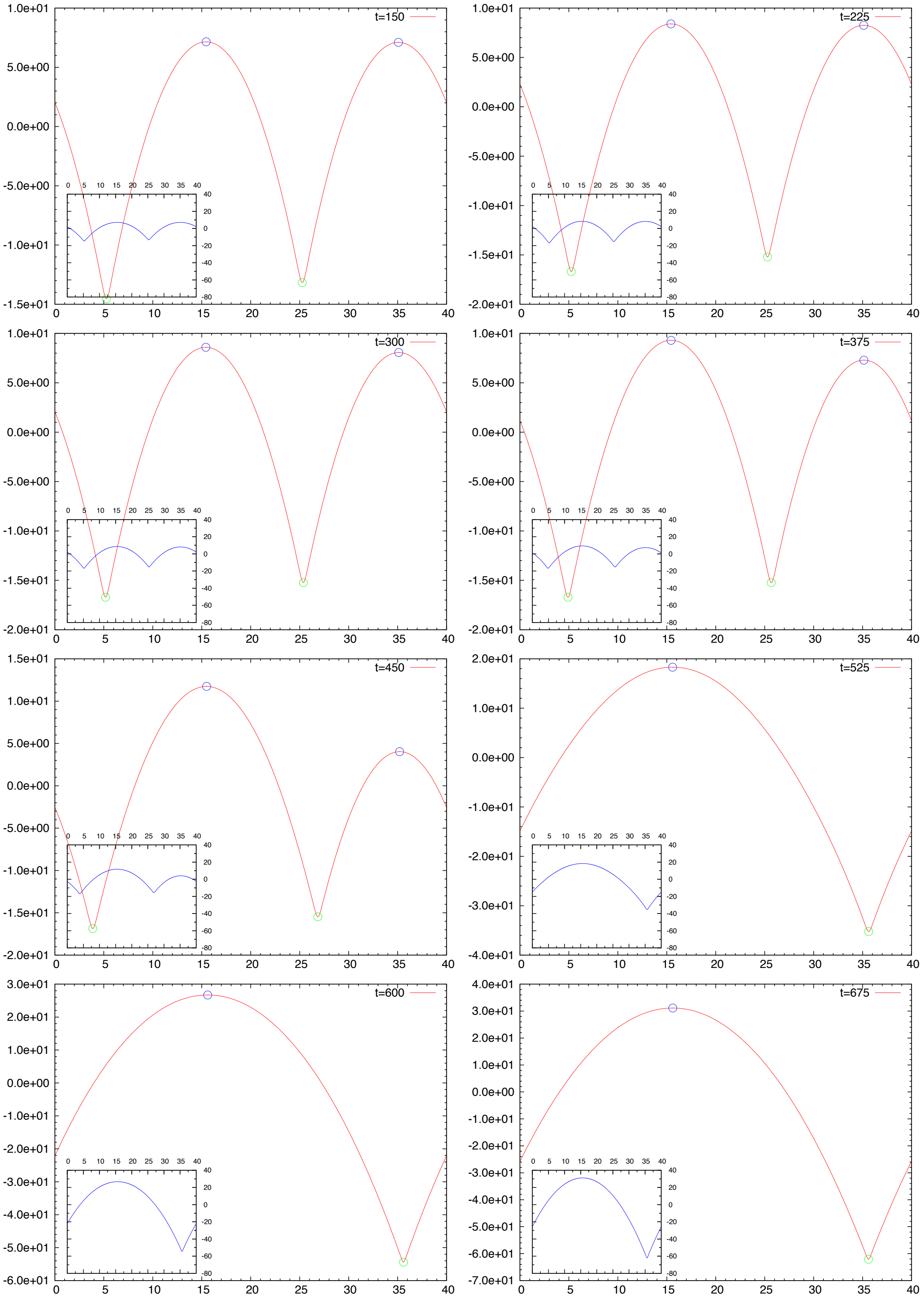


Figure 3.7 Continued – Continuation from above where we have increased the amount of time between snapshots as the profile takes longer to coarsen at later times. We have also reset the inset diagram axes to better reflect the final stages.

Here, much like before, we again see the emergence of several parabolas which then coarsen over time, reducing the number of such structures as we proceed. In Figure 3.8 we have plotted some representative times and see, in much the same way as the previous simulation, the growth of the solution from a very small initial condition into multiple parabolas which then begin to coarsen. In Figure 3.9 we have also plotted the trajectories of the inner locations in time; the paths marked out by the locations of the local minima of the profile, that is, the set of points $\{x_i(t)\}$ where $u_x(x_i, t) = 0$ and $u_{xx}(x_i, t) > 0$. Finally in Figure 3.10 we have plotted the profile, which has been shifted in height by $4t$ to display the evolution of the surface profile. Overlaid on this is the associated plot of the inner trajectories which effectively shows the same information as Figure 3.9 but with the additional detail of seeing the profile at the same time.

3.3 Outer Parabolic Structure

Now that we are satisfied that the code can produce the expected two-scale solutions we proceed to check the structure of the separate regions. In the outer regions at times which are relatively distant from coarsening events we expect the solution to resemble an arc of the parabola $p(x) := a-(x-\bar{x})^2/4$, where a and $\bar{x}$ simply affect the translation and height of the parabola. We propose two methods of checking the consistency of the code, first, a simple plot of a solution profile together with an appropriate parabola(s) superimposed for comparison. Second, we note that the second derivative of each parabola would be fixed, $p_{xx} = -1/2$, so we numerically calculate the second derivative for the entire profile and check to see if this is the case in the regions which we believe to be parabolas. We highlight two simulations on different domain sizes along with distinct initial conditions to compare the solution profile to the expected parabolic structure.

Our first example for comparison comes from a simulation on a domain of size $L = 10$. This simple example, whilst not particularly interesting, is a good judge of the consistency of our code. With relatively few points the system can quickly reach a single stable parabola which we can then easily check the second derivative of and plot the solution with a parabola for comparison. In Figure 3.11(a) we see the final solution profile together with the expected parabola with $a = 2.035$ and $\bar{x} = 2.45847$, which was fitted to the data, showing a very good match. In Figure 3.11(b) we see a plot of the second derivative of the solution profile at the final steady state. As expected we have reached the $-1/2$ mark, a good indicator of consistency in the code. This example certainly supports the consistency

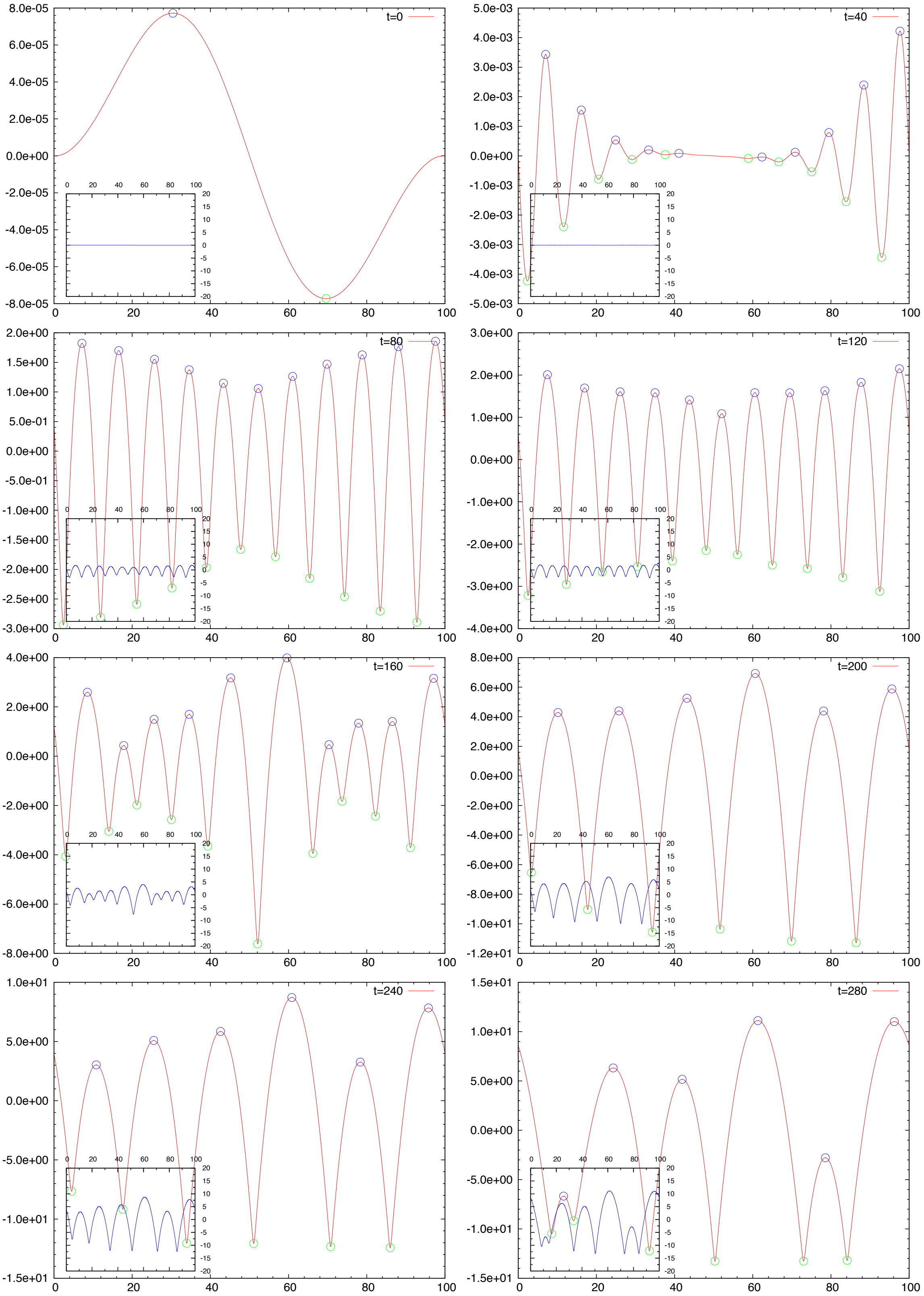


Figure 3.8 – Simulation on larger domain of size $L = 100$ using the initial condition shown in (3.2.2), $u(x, 0) = 0.0001 \sin\left(\frac{\pi x}{L}\right) \sin\left(\frac{2\pi x}{L}\right)$, showing growth and then coarsening of parabolic structures. (Continued over page.)

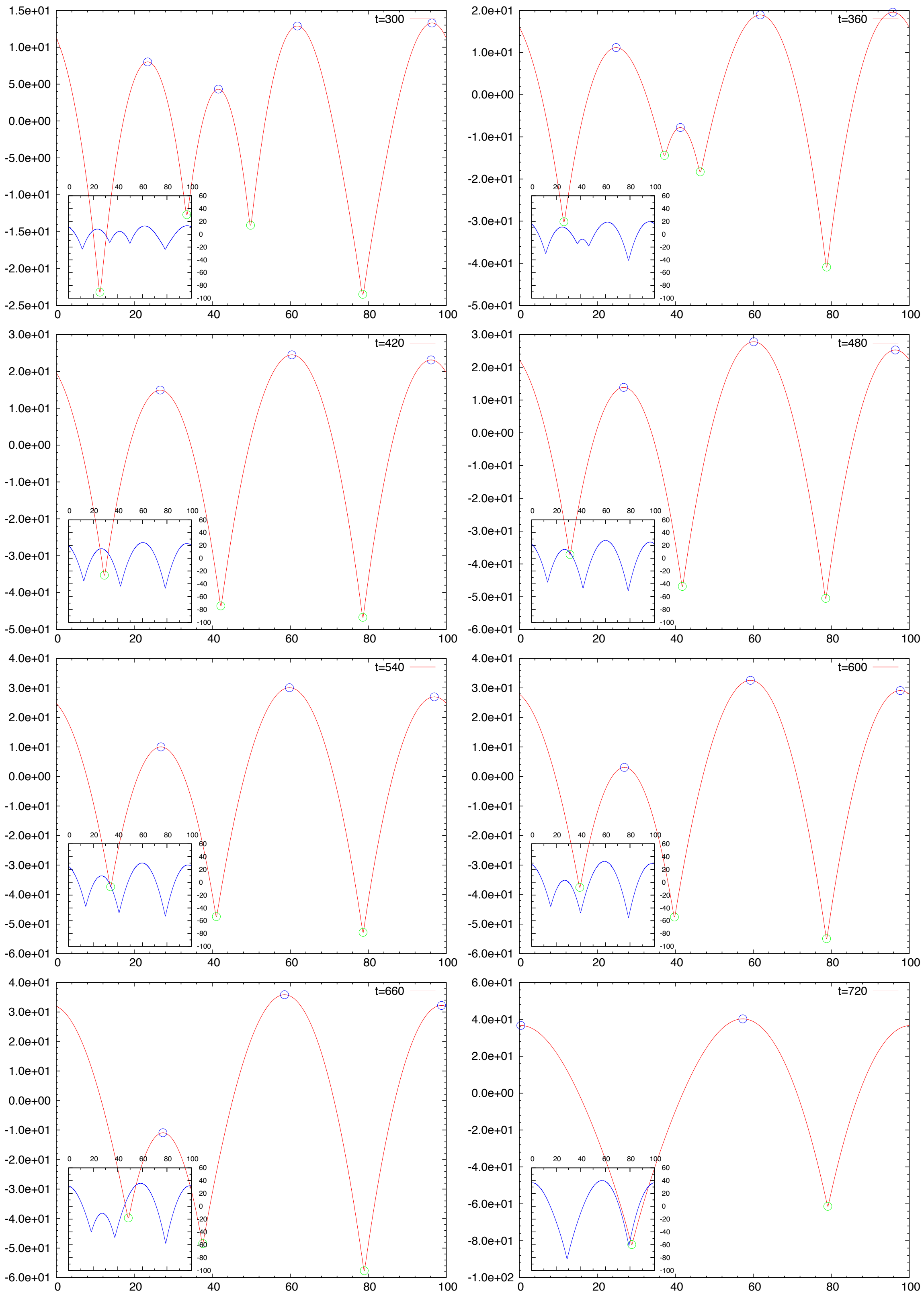


Figure 3.8 Continued – Continuation from previous page showing further into simulation. Note that we take larger time jumps between frames and the inset diagram scale has been altered from the previous page.

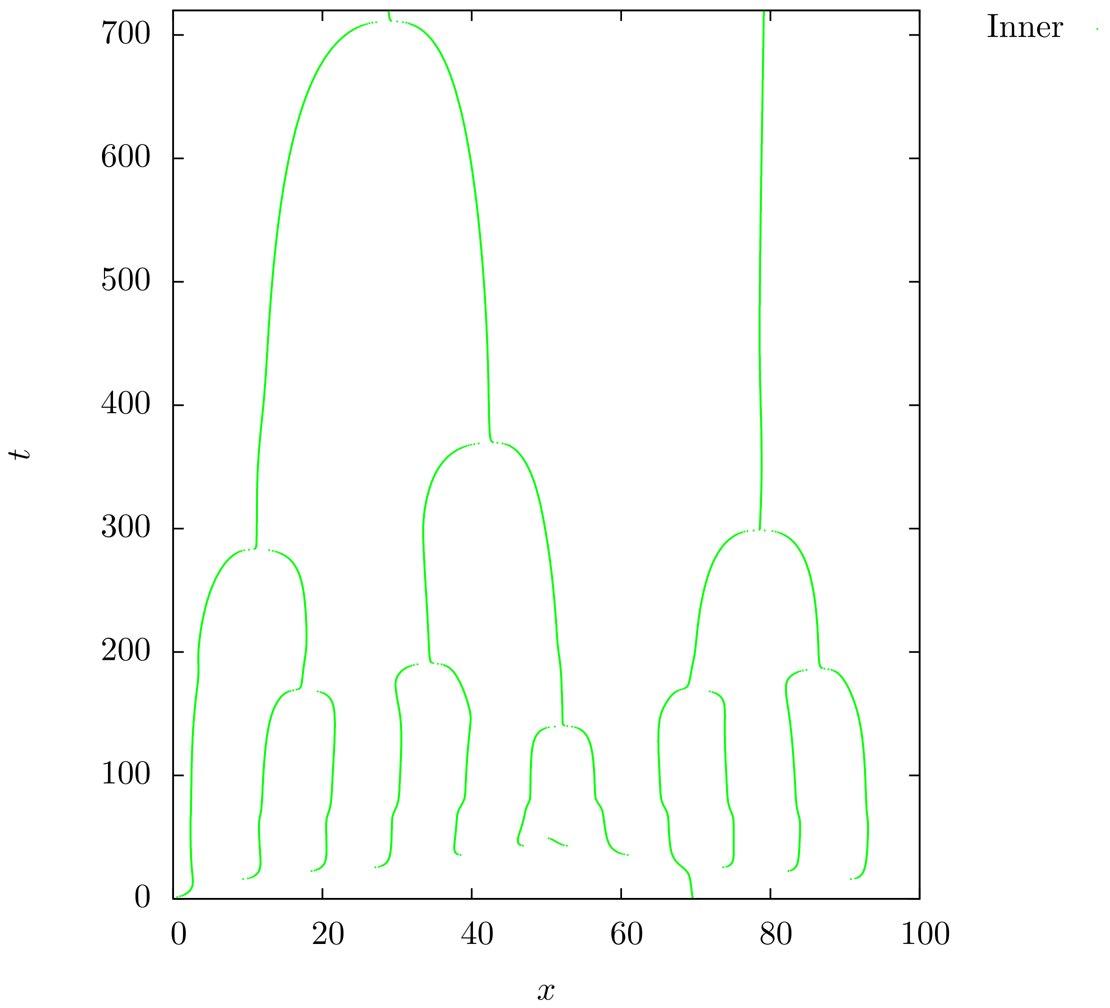
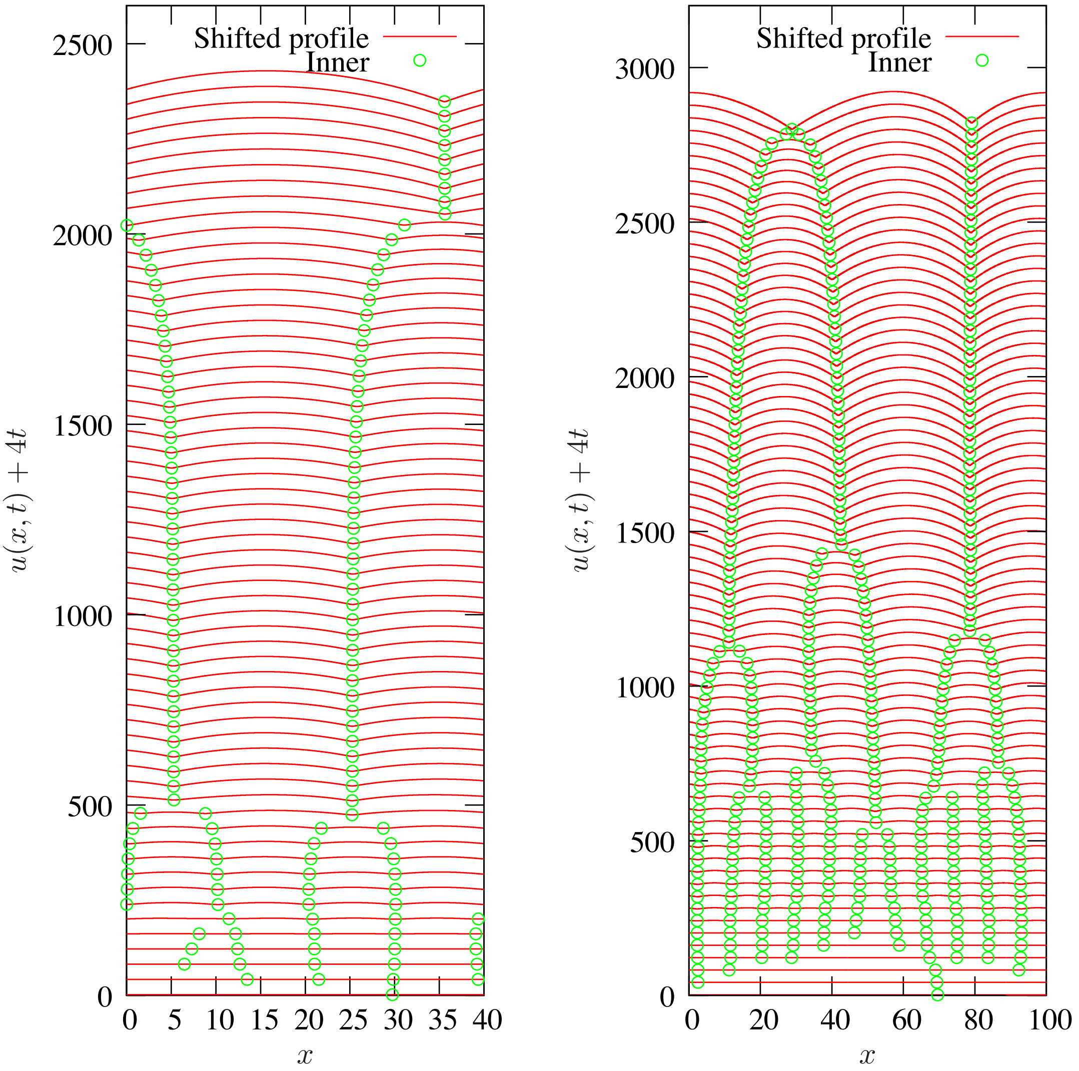


Figure 3.9 – Plot of inner location trajectories from $L = 100$ simulation, shown in Figure 3.8, showing the coarsening events (collision of trajectories) and growth in size of outer structures (gap between trajectories). Inner locations are points $\{x_i(t)\}$ such that $u_x(x_i, t) = 0$ and $u_{xx}(x_i, t) > 0$ in the full solution profile.



(a) $L = 40$ shifted profile and inner trajectories.

(b) $L = 100$ shifted profile and inner trajectories.

Figure 3.10 – Plot of solution profile, $u(x,t)$, shifted in height by $4t$, to aid visualisation, together with the associated inner region trajectories. Note the coarsening events and the overall increase in the mean size of the parabolic regions.

of the code in simulating the CKS equation.

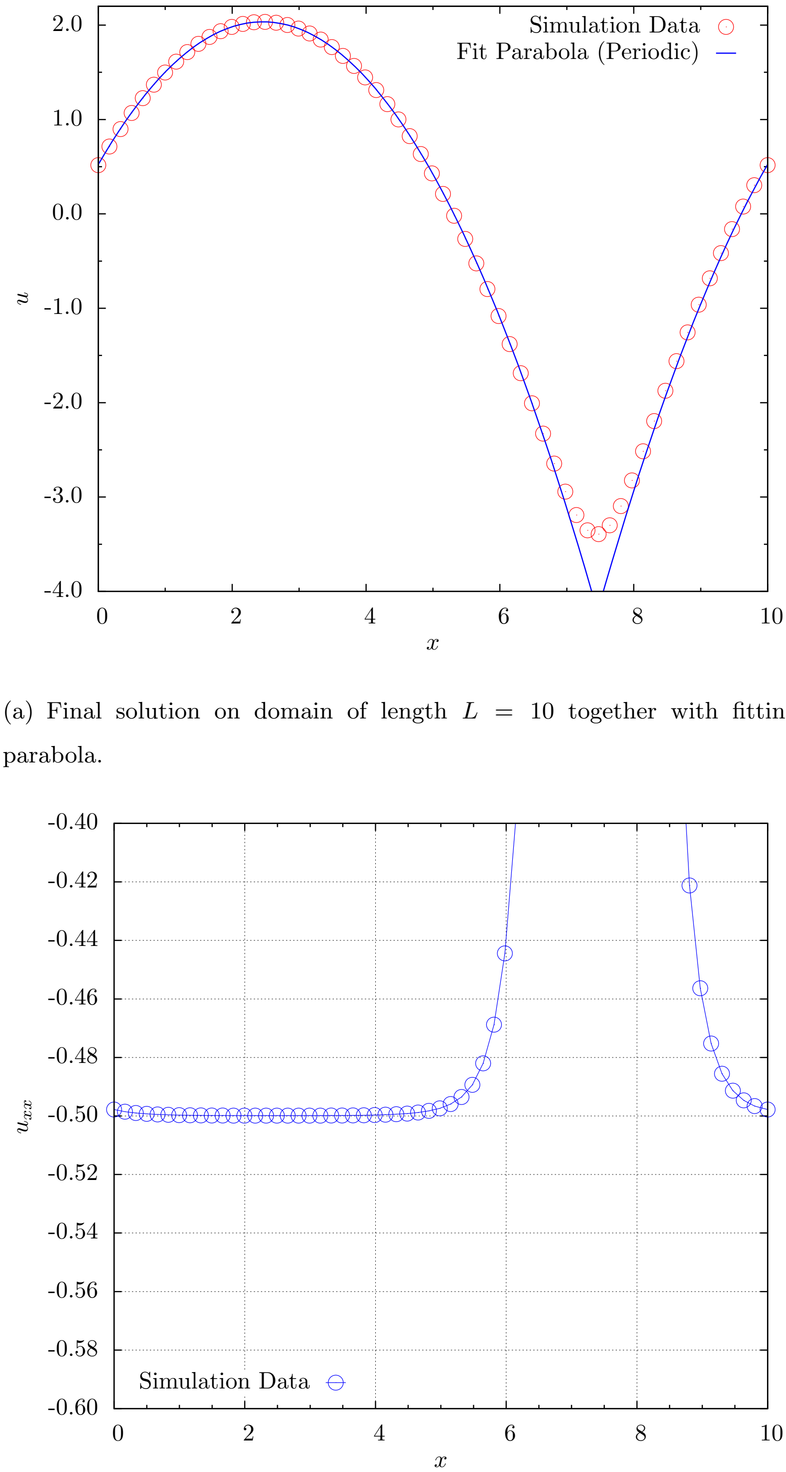
As a further check of the consistency, we look at a simulation on a larger domain and consider a time point which is far from the final, single-parabola state and also try to ensure that the profile is not in the midst of a coarsening event. We do this because we expect the complete dynamics of the system to distort the expected parabolic structure particularly around coarsening events, however, whilst the solution is 'far' from any events we expect the parabolic structure to be relevant. In Figure 3.12 we see two plots from a simulation on a domain of size $L = 60$; the solution profile together with parabolas fit to the surface and the second derivative of the profile showing the neighbourhood of the expected value $-1/2$. On the whole the resemblance at this time point is reasonable, however, there is some discrepancy in the second derivative values. At the later time shown in Figure 3.13 we see the profile is far from the expected steady regime and is, in fact, approaching an event of some type. This appears to cause a significant change in the structure of the outer regions and in particular they appear to lose their parabolic shape. Since a good portion of the evolution time is spent coarsening and hence events are taking place, this simulation leads us to believe that the outer regions are only roughly parabolic and, in fact, coarsening events cause massive deviations from $-1/2$ in the second derivative. Nevertheless we feel the close resemblance to the expected structures whilst away from events is a good indicator of the code performing the evolution process effectively.

3.4 Inner Boundary Layer Structure

We mentioned earlier that the inner regions of the solution profile are expected to have a $\log[\cosh(k(x - x_0))]$ type structure and so we investigate how well the code captures this by considering a few examples of inner regions from the simulations carried out. In the same vain as the outer structure, events will likely distort the expected behaviour, so we do our best to be 'far' from any coarsening events.

First let us briefly describe the derivation of the expected structure. As mentioned earlier, the inner regions were shown to be governed by the reduced equation $u_{xx}+u_x^2 = k^2$, where k is a constant [**39**]. Suppose now, for illustration, that the neighbouring parabolas to the inner region are of the same size, of width $\mathcal{L}$. The inner locations essentially match to the slope of the parabola in the outer regions. Assuming the parabola has equation $u = a - (x - \bar{x})^2/4$ the first derivative is then,

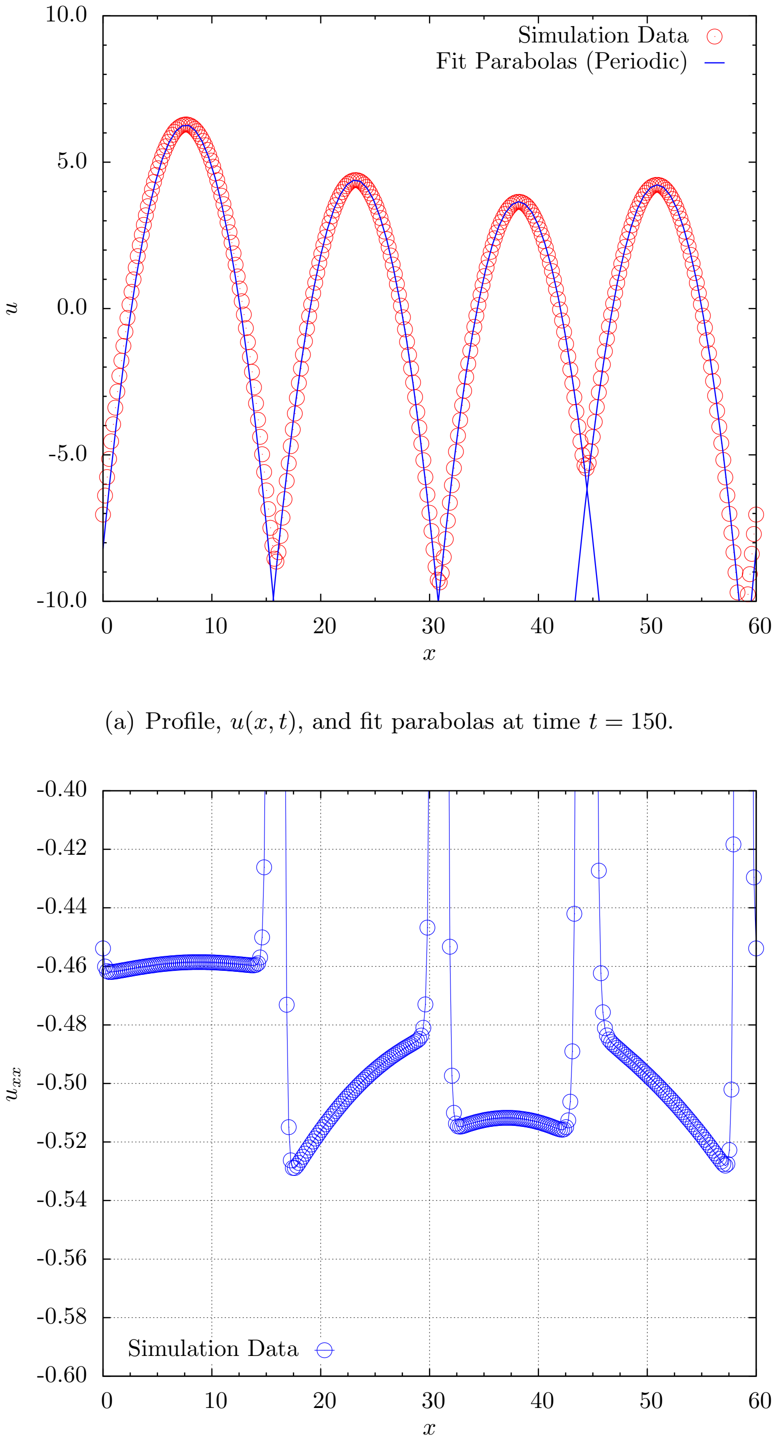
$$u_x = -\frac{(x - \bar{x})}{2},$$



(a) Final solution on domain of length $L = 10$ together with fitting parabola.

(b) Second derivative, u_{xx}, of final steady state.

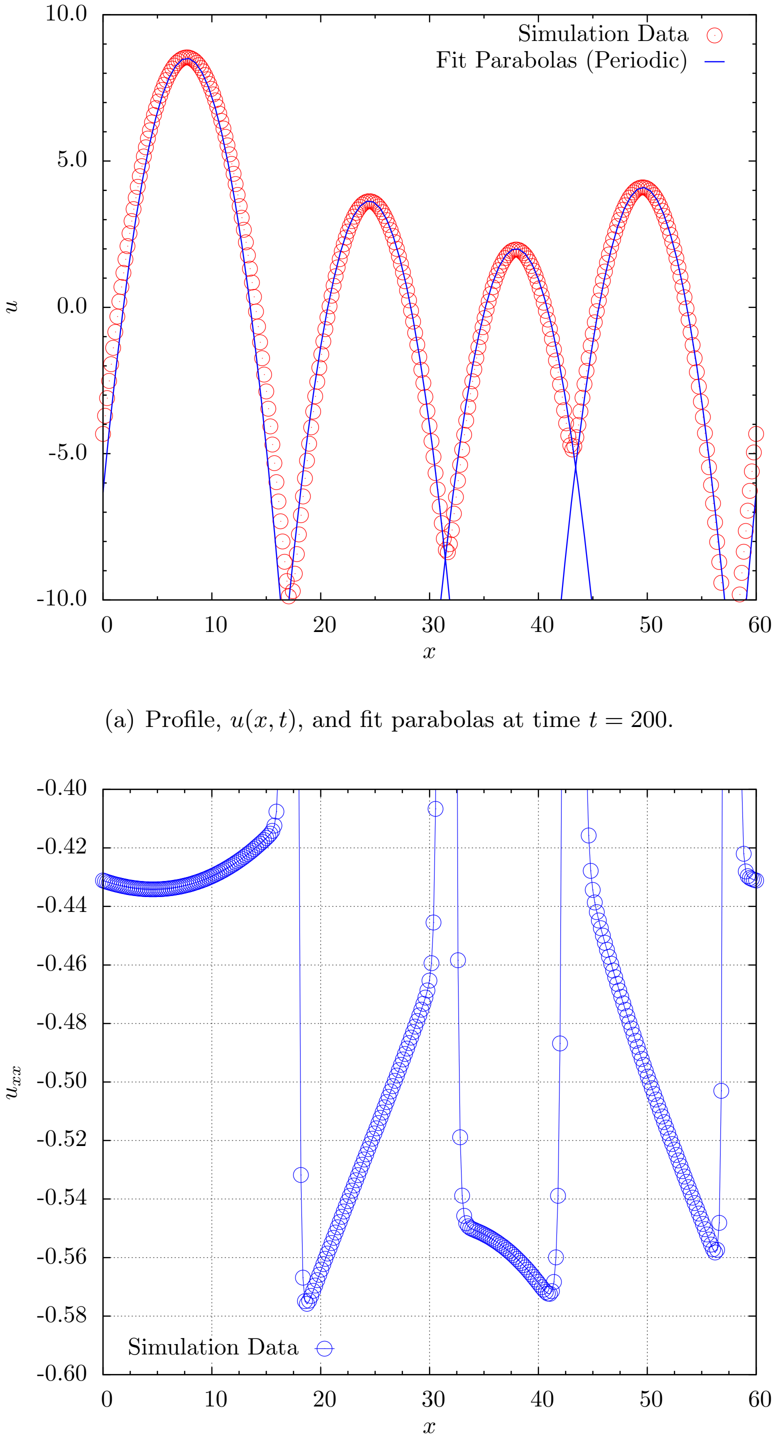
Figure 3.11 – Final steady state solution, $u(x,t)$, of CKS equation on domain of size $L = 10$ showing resemblance to a parabola at the top and good match in expected second derivative value below. Both plots make use of only a fifth of the total number of data points ($n = 300$) for clarity.



(a) Profile, $u(x,t)$, and fit parabolas at time $t = 150$.

(b) Second derivative, u_{xx}, of solution profile at time $t = 150$.

Figure 3.12 – Simulation data and overlay fit parabolas on a domain of size $L = 60$. At this stage we are 'far' from any coarsening events so we see a reasonable agreement with the fit parabolas though the second derivative plot shows there can be a discrepancy, even at such times. Only a tenth the total number of data points ($n = 3200$) have been plotted for clarity.



(a) Profile, $u(x,t)$, and fit parabolas at time $t = 200$.

(b) Second derivative, u_{xx}, of solution profile at time $t = 200$.

Figure 3.13 – Plots from same simulation as Figure 3.12 above but at a later time. As we approach a coarsening event we see a much larger deviation from the parabolic structure, clearly indicated by the large deviation from $-1/2$ in the u_{xx} plot.

and so the slope at the edge of the parabola, where the inner location is, is simply $u_x = \pm\mathcal{L}/4$, with sign depending on which side of the parabola we are considering. Since the inner region matches to a straight line slope condition, the second derivative is simply zero, $u_{xx} \equiv 0$, and so considering the reduced equation we therefore have $u_x^2 = k^2 = (\mathcal{L}/4)^2$. We must, therefore, find a solution to the problem,

$$u_{xx} + u_x^2 = \left(\frac{\mathcal{L}}{4}\right)^2.$$

Setting $q = u_x$ we then have,

$$q_x + q^2 = \left(\frac{\mathcal{L}}{4}\right)^2,$$

which can be rearranged into the following form,

$$\frac{q_x}{\left(\frac{\mathcal{L}}{4}\right)^2 - q^2} = 1.$$

Integrating both sides with respect to x we then have,

$$\int \frac{q_x}{\left(\frac{\mathcal{L}}{4}\right)^2 - q^2}\, dx = \int \frac{1}{\left(\frac{\mathcal{L}}{4}\right)^2 - q^2}\, dq = x - \hat{x},$$

for some constant $\hat{x}$, and by following this by use of the substitution $q = (\mathcal{L}/4)\tanh\theta$ we have,

$$\int \frac{1}{\frac{\mathcal{L}}{4}}\, d\theta = x - \hat{x},$$

from which we can see that $\theta = \mathcal{L}(x - \hat{x})/4$ and $q = (\mathcal{L}/4)\tanh[\mathcal{L}(x - \hat{x})/4]$. Since $q = u_x$ we must integrate once more with respect to x in order to find the solution u. In doing so we have,

$$\begin{aligned} u &= \hat{u} + \log\cosh\left[\frac{\mathcal{L}}{4}(x - \hat{x})\right], \\ &= \hat{u} + \log\cosh\left[k(x - \hat{x})\right], \end{aligned}$$

as expected where the constants $\hat{u}$ and $\hat{x}$ simply set the location and height of the appropriate inner region. The main point to note is that the quantity $k = \mathcal{L}/4$ is related to the size of the neighbouring parabolas and isn't constant throughout the system.

We now consider this inner structure more closely by considering specific simulations to discover the code's ability to capture this feature. Our first example will be to reconsider the $L = 10$ example from the previous section since this reaches a stable state with one well defined inner region and is therefore not featuring in any kind of coarsening event. In

Figure 3.14 we see the solution profile plotted together with the predicted fit curve. The parameter k was selected based on the extent of the parabola which is believed to take up the majority of the domain with the exception of the inner region and thus has size $\mathcal{L} \sim 9$, therefore $k = 9/4$. Overall the shape of the fit curve is good and the selection of k seems consistent with the data.

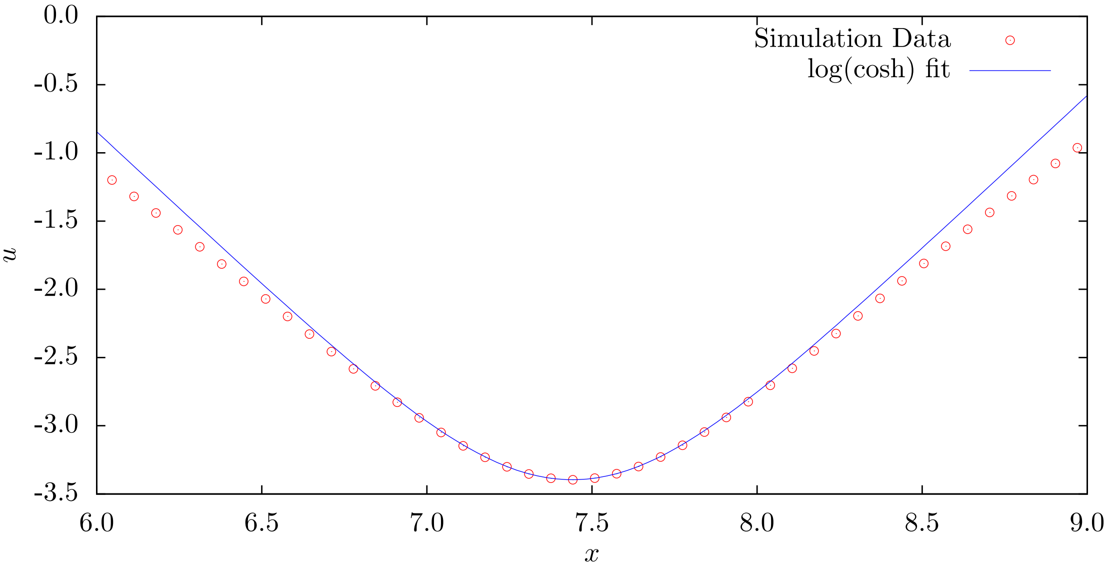
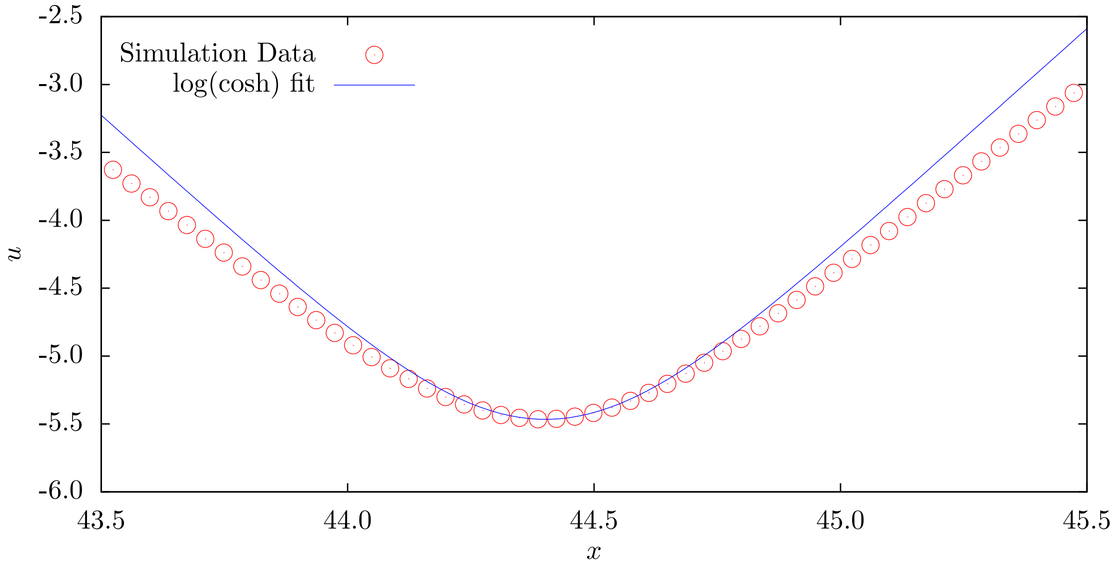
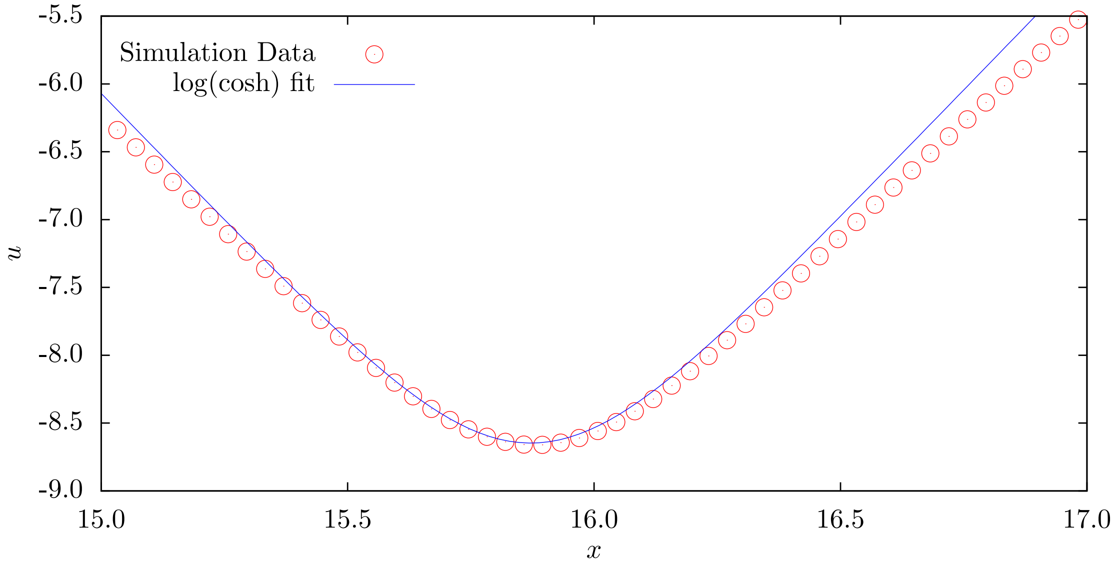


Figure 3.14 – Solution profile from $L = 10$ simulation plotted in inner region together with fitting $\log(\cosh(kx))$ curve where $k = 9/4$, a value predicted by the extent of the parabola which is believed to take up almost the whole domain and thus have size $\mathcal{L} \sim 9$. (Every second data point plotted for clarity.)

For our second example we consider two of the inner regions from the $L = 60$ simulation at $t = 150$ from the previous section. Again we have plotted the appropriate fitting curve where this time based on the rough size of a parabolas either side of the corresponding inner region. In 3.15(a) we have considered that the total extent of the neighbouring parabolas, from Figure 3.12, is approximately 28. Accommodating for the space the inner regions take up we assume the extent of the parabolas to be around 13 each and thus $k = 13/4$. In 3.15(b) the total extent is, as a result of $L = 60$, found to be around 32 giving two parabolas of size approximately 15 plus room for the inner regions, and hence here $k = 15/4$. As before we see overall there is a good agreement in the shape for these predicted values of k.



(a) Profile and fit curve for one inner region at time $t = 150$ on domain $L = 60$ with $k = 13/4$.



(b) Profile and fit curve for a second inner region at time $t = 150$ on domain $L = 60$ with $k = 15/4$.

Figure 3.15 – Two inner regions from the $L = 60$ simulation in the previous section showing the comparison to the predicted $\log(\cosh(kx))$ type structure. In (a) we have used $k = 13/4$ to reflect the fact the parabolas either side of this inner region are less than a quarter of the total domain size. In (b) we have used $k = 15/4$ since the parabolas here are larger in size than in (a). In both cases we see a very good fit of the data and theory. (Every second data point plotted for clarity.)

3.5 Size of Time Step and Grid Spacing

One final area of interest is to consider the size of time step used and check that we are performing suitably accurate time steps. In order to check the consistency of the update for a particular time step we propose a set of simulations, each with identical parameters except the size of the time step, grid spacing or both. In each case we shall use various sizes of time step together with various sizes of grid spacing. Whilst not a rigorous conclusion this does at least give us faith in the output of data from the code. For comparison we shall simply consider the trajectories of the inner regions as our gauge of accuracy.

In Figure 3.16 we see the expected paths for our chosen simulation on a domain of size $L = 30$ with grid spacing $h = 1/25$ and $\Delta t = h^2/10$. This will be our benchmark simulation as we can readily reproduce these results with smaller grid spacings and smaller time steps, so we consider this to be representative of the correct paths.

In Figure 3.17 we see the paths which emerge by changing only one of the control parameters, either increasing the grid spacing or increasing the time step. In both cases we see the collapse of the system at later times. In each case, however, the paths agree at early times, whilst the system has no small inner regions, but begin to disagree as the size of the inner regions reduce. This leads us to believe that care must be take to ensure stability in the system. If we perform one final simulation with $h = 1/25$ as before but this time with $\Delta t = h^2/5$ we see the same paths as the original case, shown in Figure 3.18 together with the 'correct' paths. This is what lead us to using $\Delta t = h^2/5$ in our simulations.

We fix this time step size and ensure stability on different sized domains by modifying the number of grid points used. Whilst often able to undercut this number, we found $n = L^2$ to be approximately the right number of grid points necessary for each domain. At the very least this number gave a good starting point for fine tuning the number required for both efficiency and accuracy. It should be noted that instability tended to occur when the parabolas became larger in size. Larger parabolas force large jumps in the slope to occur across the inner boundary layers, which need to be properly resolved. As a result it is perfectly sufficient to undercut the L^2 number if we only required early evolution data but more care is needed if we hope to coarsen all the way to a single parabola on a large domain.

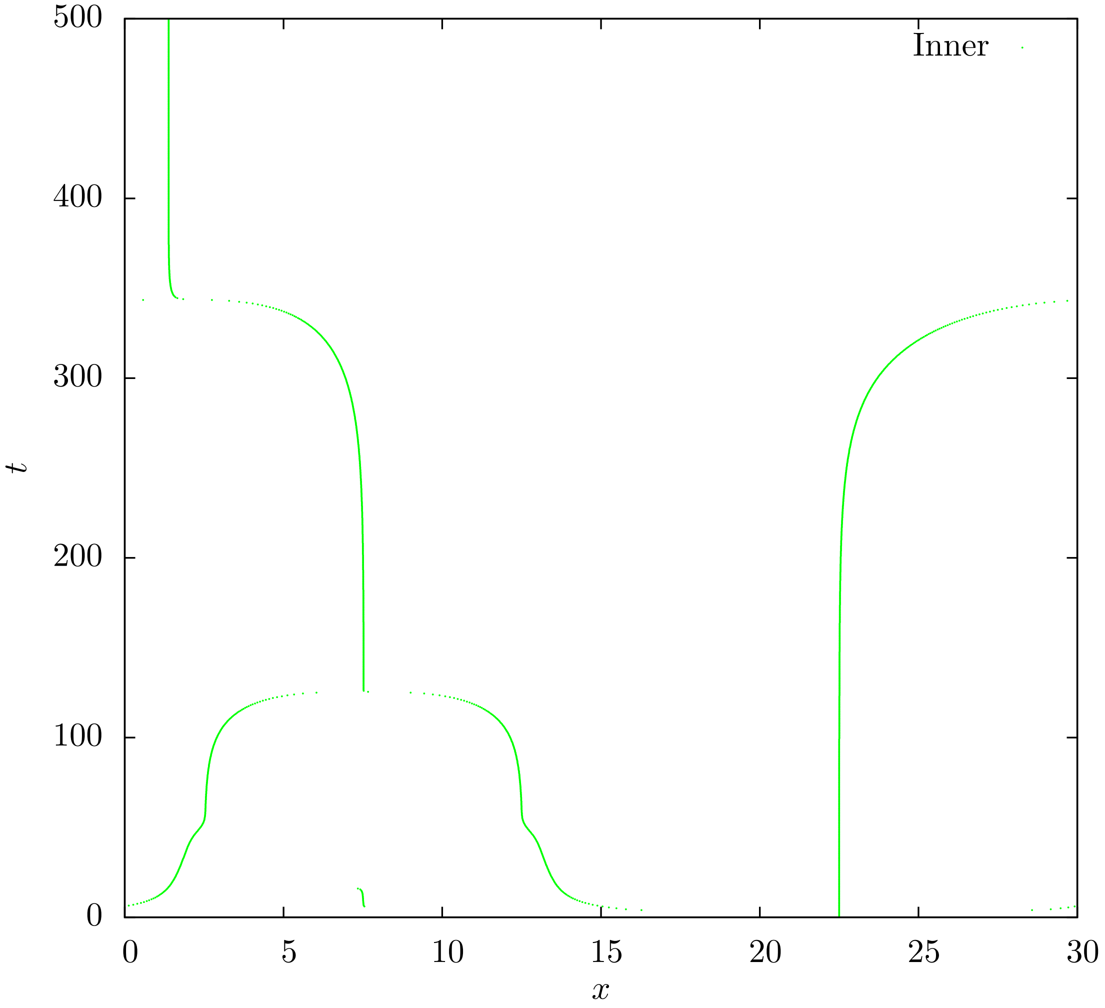


Figure 3.16 – Typical plot showing the correct, stable inner region trajectories from a general simulation on a domain of size $L = 30$. Here we have chosen $h = 1/25$ and $\Delta t = h^2/10$.

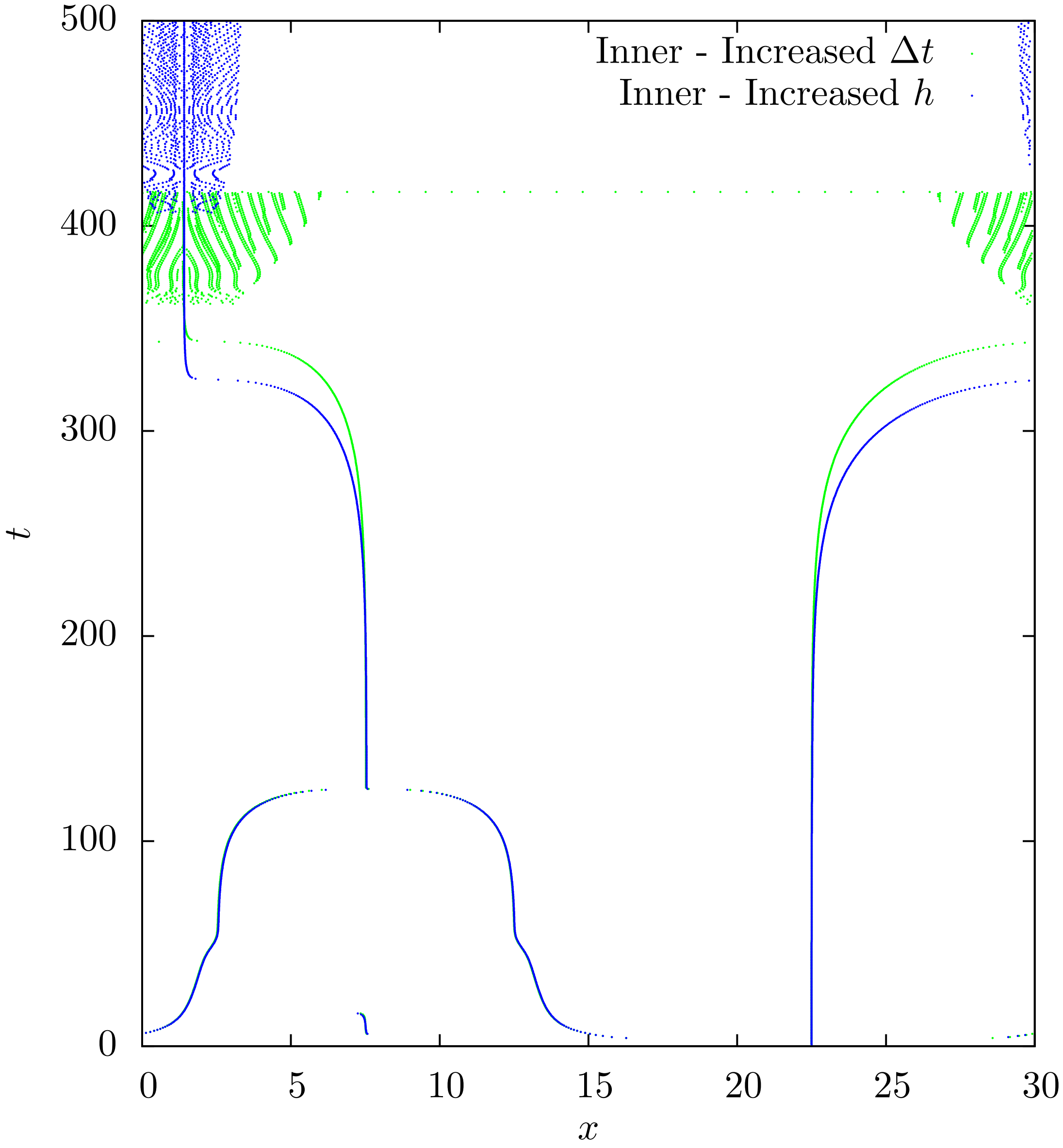
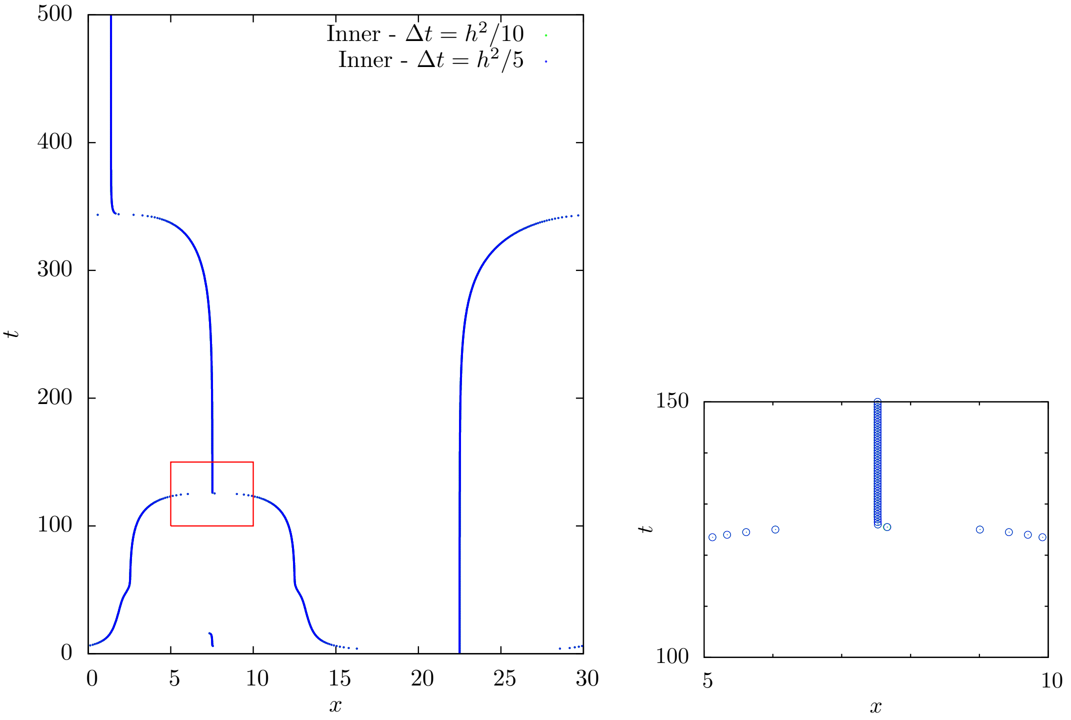


Figure 3.17 – Inner region trajectories for both increased Δt ($\Delta t = h^2$) and increased h ($h = 3/40$). Note that the system breaks down in both cases as signified by the sudden increase in inner regions at later times (the solution develops a wave like structure with multiple peaks and troughs.) In the case of the increased Δt the system completely collapses and doesn't reach the desired final time. In the increased h case the trajectories do seem to match well at early stages but break down occurs as the size of the inner regions is reduced to or approaches the size of the grid. In both cases the code copes well in the initial stages but loses accuracy as the inner region becomes narrower at later times.



(a) Plot of trajectories.

(b) Enlarged plot of red box from (a).

Figure 3.18 – Inner region trajectories for the time step $\Delta t = h^2/5$, which became our regularly chosen time step, showing the match to the correct paths from Figure 3.16. Inset diagram shows the event in the red box enlarged with the green $h^2/10$ data barely visible, showing essentially no discrepancy.

4 Coarsening Pathway and the Dynamical System

In this section we aim to characterise the dynamics of the system by first considering the types of event that can take place before comparing these circumstances to that predicted by the dynamical system proposed by Politi et al. in [**89**], and shown in (1.2.1). We shall see that there is some similarity in the outcome but significant reason to believe that this dynamical system is incapable of capturing the true dynamics of the solution profile.

4.1 Types of event

From our simulations we believe there to be two types of event which can take place during coarsening of the solution profile. One is much more prevalent than the other and corresponds to the shrinking and removal of a single parabola over time. The second is more rare and corresponds to the merging of two parabolas into one, or equivalently the sudden loss of an inner region from the solution profile.

4.1.1 Prevalence of the $2 \to 1$ event

If we consider Figures 3.7 and 3.8 we can clearly see the system coarsening as it reduces the number of parabolic type structures, ultimately ending up with one large structure spanning the whole of the domain. During this process several smaller parabolas must vanish in order for this to happen. The first type of disappearance examined is what we would consider to be the 'standard' method of loss. This involves a parabola shrinking in size until it disappears and has effectively been consumed by the two larger neighbouring parabolas. Several of these types of event can be seen in the figures. On the larger domain shown in Figure 3.8 alone we can see nine such events have taken place. These types of events are very common as the solution coarsens on its way to the final state.

If we consider the trajectories of the inner regions shown, for example, in Figure 3.9 we see the typical signature of this type of event. As a parabola shrinks in size the two inner regions to the left and right become closer and closer before eventually the parabola is removed and the inner regions merge to form one region separating the two remaining parabolas. These events can clearly be seen as the collision of two inner paths and one single inner remaining. As this type of event corresponds to the loss of one inner region from two, we shall call this type of event a $2 \to 1$ event.

4.1.2 The exceptional $1 \to 0$ event

The second type of event we have encountered appears only in very few circumstances, in particular it appears from our simulations that a great deal of symmetry must be present in the solution profile for this type of event to take place. Nevertheless it is possible under certain circumstances for the solution profile to form several parabolas and then for two to shrink simultaneously and in fact merge before they disappear. This type of event can clearly be seen in Figure 3.19 where we see a merging of parabolas at $x = 20$ and t between 87.5 and 100.0.

The signature of this type of event in the inner locations can clearly be seen in Figure 3.20. In terms of purely these inner locations we see that this corresponds to a sudden removal of an inner region and so we shall call this a $1 \to 0$ type event. This is in stark contrast to the more typical type of event in the previous section where we saw two inner regions come together and one remain. An example of this first type of event can also be clearly seen at the later time of approx. $t = 125$ in Figure 3.20 where once the two small parabolas have merged to become one, the size of this parabola is still small in comparison to its neighbours and so it is consumed in the 'standard' $2 \to 1$ manner.

4.2 Comparison with the Dynamical System

To compare the proposed dynamical system (1.2.1), to the evolution of the PDE solution profile we propose three distinct methods. Each of these methods will focus on the particle locations from the dynamical system, which as mentioned earlier correspond to the x location of the inner regions in the PDE solution. If the dynamical system is to truly capture the PDE evolution then it must succeed in capturing these basic events. The first such comparison we shall make is to consider the $1 \to 0$ events and their appearance, or more specifically lack of, in the dynamical system. Following on from this we will consider the particle trajectories and finally we shall explore the possibility of multiple PDE solutions existing with the same locations of the inner regions.

4.2.1 No signature for the $1 \to 0$ event

In Section 4.1.2 we drew attention to the possibility of two parabolas merging to form one larger parabola (See Figure 3.19), and although we remarked that this situation was generally quite rare it is still a feature of the evolution of the PDE which we would expect the dynamical system to capture.

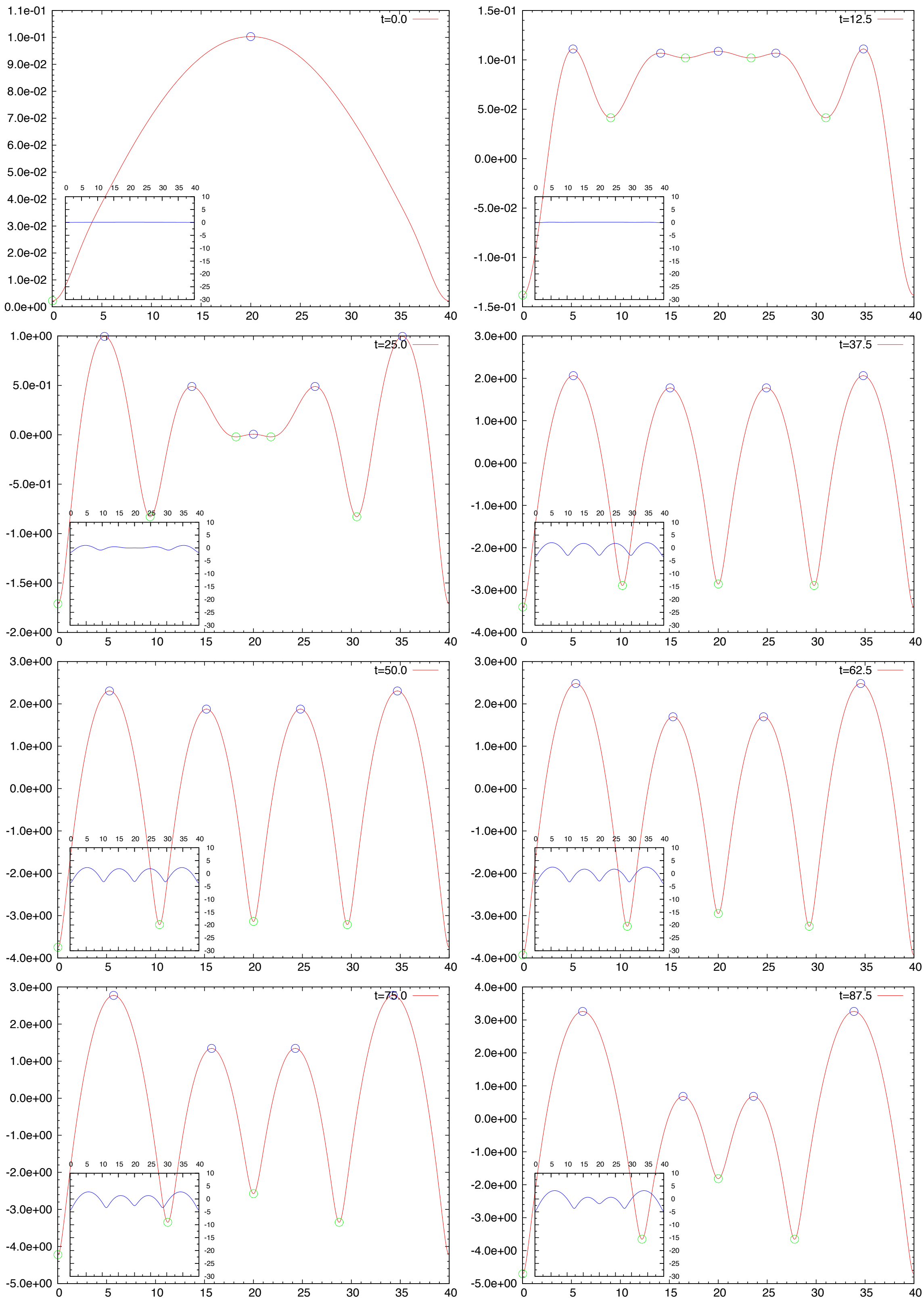


Figure 3.19 – Evolution of solution showing the, comparatively rare, 1 → 0 event whereby two parabola type structures merge into one larger parabola. This type of event is unique in that it corresponds to the sudden disappearance of an inner location, shown here somewhere between $t = 87.5$ and $t = 100.0$. (Continued over page.)

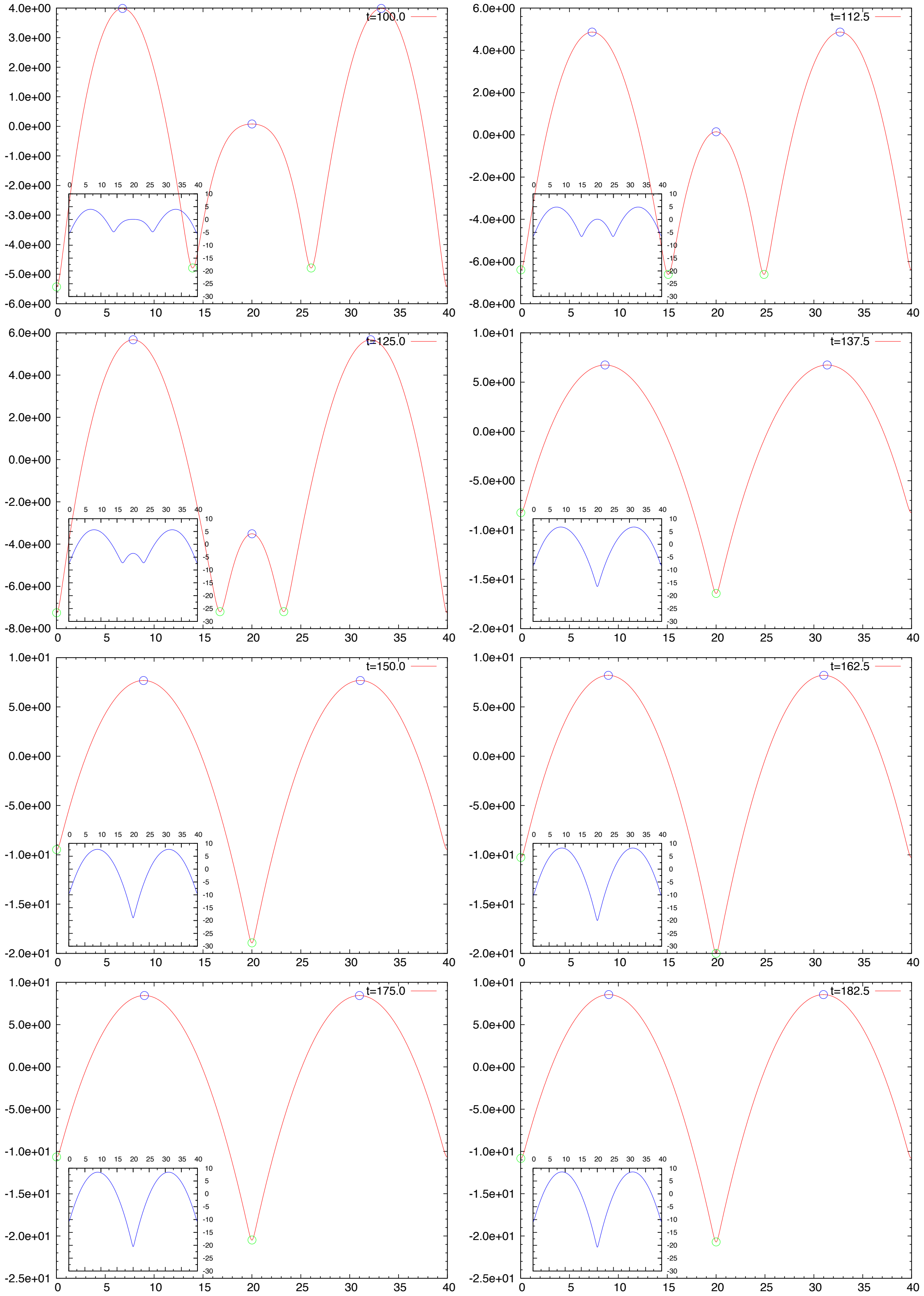


Figure 3.19 Continued – Continuation from previous page.

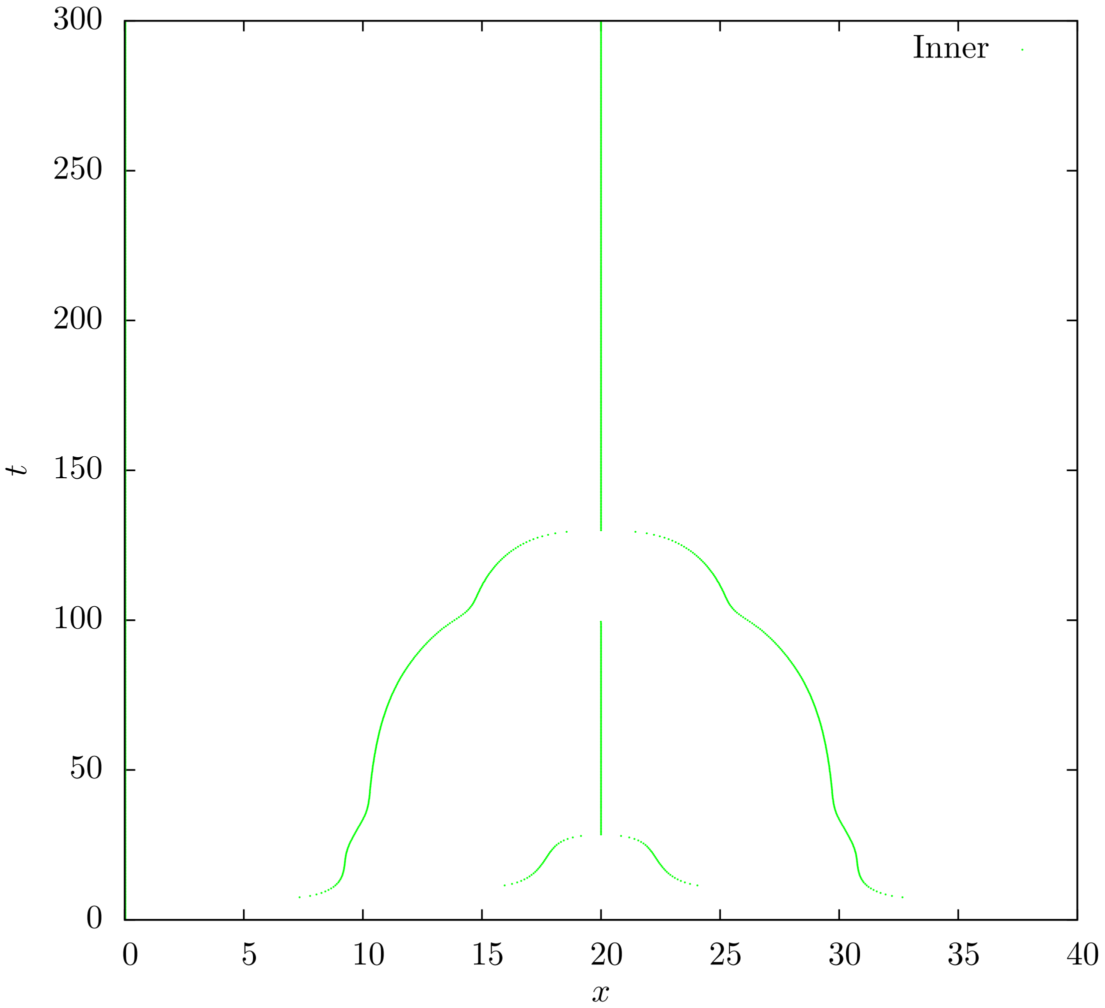


Figure 3.20 – Trajectories of inner points from Figure 3.19 showing the disappearance of the inner region at $x = 20$ shortly before $t = 100.0$. This is quickly followed by the more common $2 \to 1$ event where the newly formed parabola shrinks and vanishes somewhere around $t = 125.0$.

Unfortunately such a situation was not considered in the derivation of the dynamical system. In order to proceed through the analysis in [**89**] the authors makes use of a 2-parabola set-up and consider the case when one parabola grows and the other shrinks. This situation captures the more general type event but does not contain the ability for a particle to essentially vanish as we would see in this circumstance. The dynamical system would need to contain some mechanism by which two lengths join together under particular circumstances and this type of mechanism is certainly not present. A typical example of a fading particle trajectory together with the trajectories the dynamical system predicts can be seen in Figure 3.21.

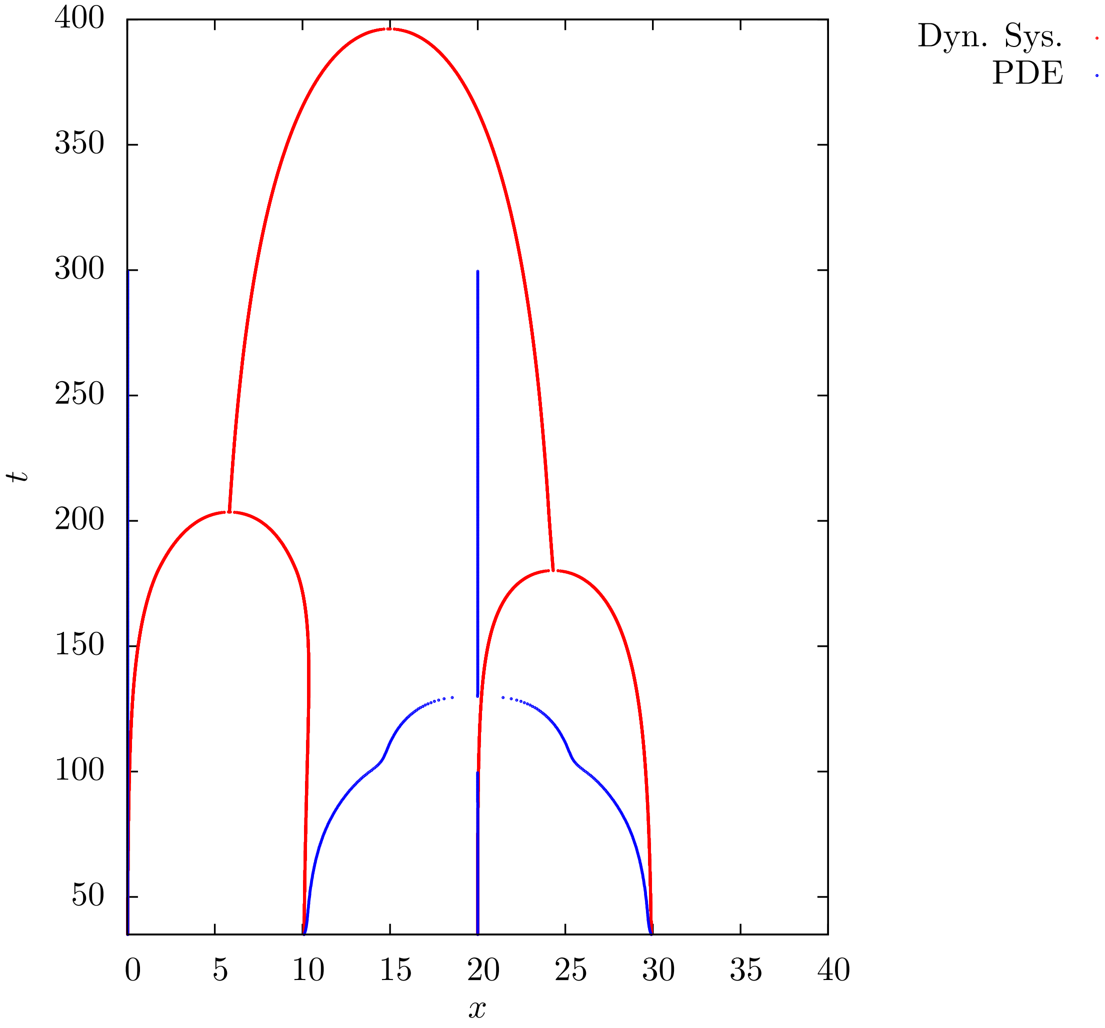


Figure 3.21 – Comparison of PDE inner locations and dynamical system showing the dynamical system's inability to capture the vanishing inner region at around $x = 20$, $t = 100$. This discrepancy results in a dramatically different outcome.

4.2.2 Other Discrepancies

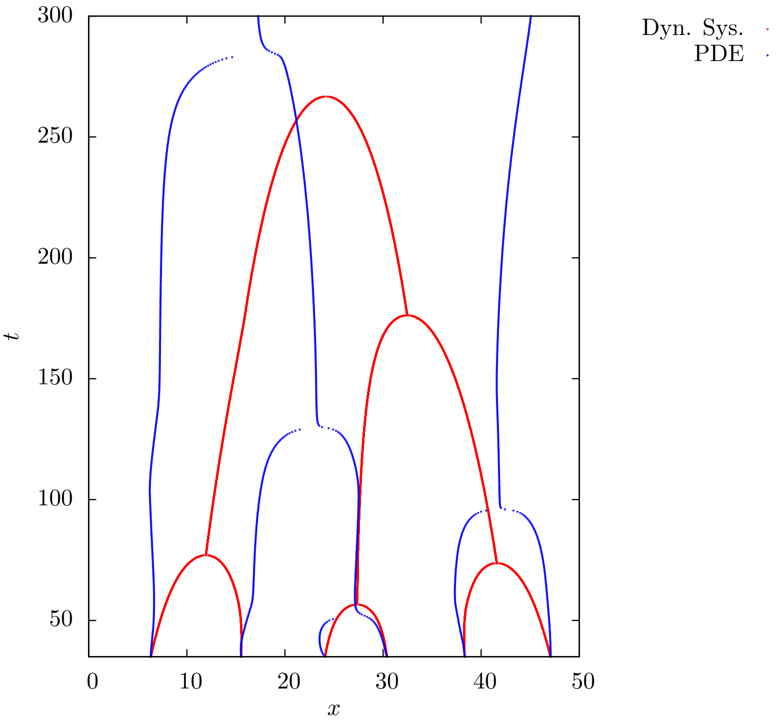
The next method of comparison we shall consider will be to check the trajectories of the inner locations from both the PDE simulations and the dynamical system. This is a very straightforward method of comparison since we simply take a PDE simulation, which ideally contains a reasonable number of parabolas, and consider the inner region trajectories. Since we have already been gathering data on the inner locations we can simply pause time, read off the locations, feed these to the dynamical system and then continue the PDE evolution. In this way we can guarantee that the initial conditions of the two systems are identical and so represent a fair comparison of the inner trajectories. Two examples of the pairs of trajectories can be seen in Figure 3.22.

In these two examples, and many others not shown, we can clearly see discrepancies between the two sets of trajectories. Particular mention should be made of the different order of events shown in Figure 3.22(a). This would seem to suggest that the distance between inner regions is not the only governing factor in the evolution of the system and that other system characteristics play an important role. Figure 3.22(b) does seem to show a relatively good agreement between the two methods suggesting that in some circumstances the dynamical system captures the evolution of the solution well.

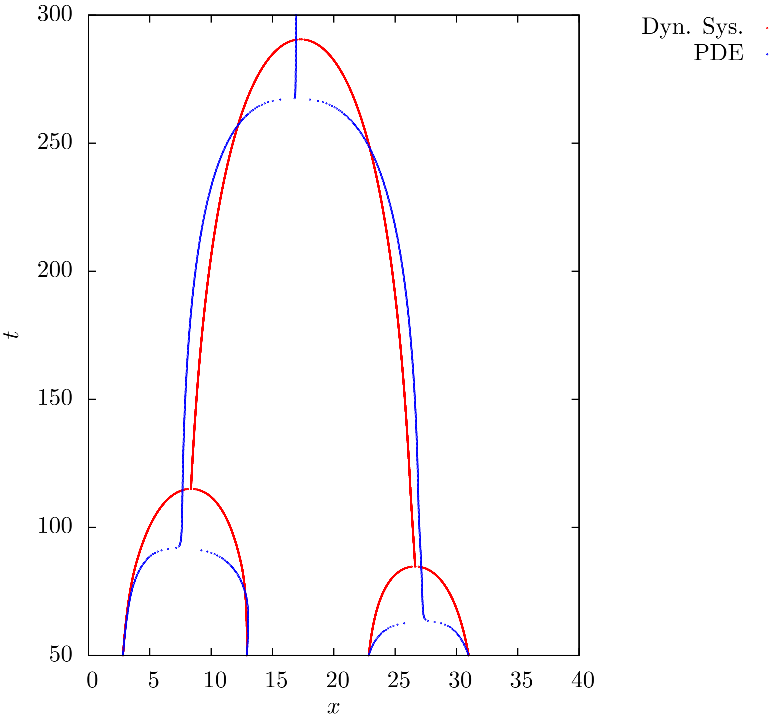
Our final comparison comes from considering the possible outcomes from one initial condition. The dynamical system is as stated and so from a given initial condition is only capable of one particular outcome. The PDE on the other hand is capable of supporting multiple profiles whilst maintaining the same location of the inner regions. This is primarily due to our ability to vary the height at which each inner region is on the vertical axes, a degree of freedom which is projected out when transforming the system into a particle model, and still maintain a valid continuous profile. Removing this information could have dramatic consequences on the evolution of the solution profile, in particular possibly leading to a difference of outcome from a given initial condition.

5 Overall Conclusions

We have probed the CKS equation and gone to length to find a suitable, accurate numerical method of solving the equation. The information gathered here allowed us to explore the dynamics predicted and we have shown that this dynamical system is incapable of fully



(a) $L = 50$ simulation inner trajectories.



(b) $L = 40$ simulation inner trajectories.

Figure 3.22 – Plot of trajectories showing comparison between inner regions directly from PDE simulation and proposed dynamical system for two different sized systems. Both the PDE and dynamical system used identical initial data. In (b) overall we see a reasonable agreement between the two types of data. In (a), however, we see significant discrepancies including a change in the order of events.

capturing the evolution of the surface profile. In particular there are a number of events, as detailed here, which are not or indeed cannot be captured by the dynamical system as stated. We believe the cause for this is that the dimension of the dynamical system, namely the number of inner regions, is too low to fully contain all the necessary information to build up a solution profile. Even if parabolas are assumed to have fixed curvature in the outer regions we see that by assuming parabolas only join in a one-dimensional line, i.e. the x-axis, we have projected out information about relative heights of individual parabolas. We believe this loss of information is what leads to the discrepancy. By possibly increasing the n inner region degrees of freedom to $2n$ (position and height) we believe there is a better chance of being faithful to the original problem. Such a scenario can be visualised by considering that each inner location also has some associated 'mass' and that the dynamics reflects both spatial locations, as currently, and some relation to this new mass.

Chapter 4

Hill-Valley Facet Model

1 Introduction and Background

Faceted surfaces are present in a wide variety of systems and have been subject to a wide array of theoretical and numerical research. Examples include the evolution of phase boundaries between two elastic solids [**49**], evaporation or condensation during thermal annealing [**121**] and many other solidification systems, e.g. [**33**, **52**, **78**, **79**, **99**, **103**, **122**, **124**]. Such systems can display a range of dynamics dependent on the system in question and as a result exhibit a wide variety of evolutionary structure. A general faceted system consists of a large, finite or perhaps infinite, set of individual facets which are commonly constrained in some way, for example in one-dimension the facets may be limited in their choice of slope. Facets are connected together in such a way so as to form a continuous surface in space, in one spatial dimension this simply means the facets form a long, continuous chain. The system of facets then evolve subject to some velocity law which is generally based on geometric properties of the individual facets, e.g. length, slope, area, or perhaps configurational quantities such as height. This facet velocity law then governs the entire evolution of the system and so these models are deterministic in nature, that is, from a given initial condition and with a specified velocity law the dynamics is completely determined. The origins of many faceted models are found to be PDEs which govern physical systems such as [**19**, **46**, **120**, **122**]. Analysis of these PDEs then leads to both the faceted surface, which may be an approximation of some kind, and an effective velocity law which determines facet evolution. Once a perhaps highly complex system is reduced to a faceted model with associated facet velocity, large scale simulations become much more amenable, by reduction of a general surface evolution to that of a set of ODEs, and can often lead to a deeper understanding of the original problem by probing the simplified faceted model. These large scale simulations allow us to consider the underlying statistics and probability distributions with the aim to predict information that matches and informs the original problem and in turn gives insight into the general surface statistics.

Many such faceted systems display a general loss of facets from the system in finite time, facets either shrink and disappear from the system or merge with one-another, e.g. [**79**, **122**, **124**]. The faceted surface, however, will still occupy all of the specified initial space and so we see an increase in the mean size of the remaining facets with a reduction of the overall number of facets, the typical signature of a coarsening system. The exact mechanism of loss will be highly dependent on the particular velocity law applied and other factors within the faceted system, such as the constraints on individual facets. Note, however, that these constraints are often determined by the analysis of the underlying PDE problem so, in fact, the appearance of and structure in and around coarsening events is purely determined by the original problem. Further to this overall concept of coarsening we see the emergence of dynamic scaling, that is, statistics of the surface remain constant when scaled by the characteristic length scale, often the mean size. This scaling parameter is also frequently shown to obey some form of power law scaling in time, again something consistent with general coarsening systems. As we have already mentioned we aim to understand and predict the system statistics so the presence of dynamic scaling is a property which is frequently exploited to further our understanding.

1.1 Faceted Systems: A Coarsening Dynamical System

We now see that facet dynamics models provide another context for coarsening systems and in particular we note that in place of a probabilistic evolution of the system seen in other examples the evolution here is completely deterministic with the dynamics governing the evolution and eventual coarsening. With the dynamics set and a coarsening law obtained from the underlying problem we see in this context that we have a *coarsening dynamical system* ($\mathcal{CDS}$) [**122**] which signifies that the system combines both dynamics and coarsening.

These faceted $\mathcal{CDS}$ are amenable to the kind of dynamic-scaling analysis which has proved successful in other coarsening contexts such as Ostwald ripening [**68**, **91**, **117**, **118**] and thin film dynamics [**45**, **47**, **48**, **83**, **92**]. In very much the same way as discussed earlier, and in many of the listed facet examples, we can probe the overall system by exploring the dynamic scaling properties, scaling laws and indeed consider the associated distributions which arise, in particular making note of the presence of dynamic scaling. The importance of understanding the probability distributions for physical problems leads us naturally to the goal of their prediction.

1.2 Theoretical Predictions and Mean-Field Models

Predicting the evolution of the statistics within a facet dynamics model can be performed in much the same way as other dynamical systems and in this broader context is the subject of much research in the field [**11**, **30**, **47**, **48**]. As already mentioned, understanding this evolution can provide us with a greater insight into the underlying physical problem. These types of predictions closely follow the overarching ideas presented in the seminal papers by Lifshitz & Slyozov [**68**] and Wagner [**118**] (LSW theory) who carried out independent analyses of Ostwald ripening. LSW theory assumed that there we no collision of objects, however, care must be taken in a general facet model to ensure that all sinks and sources of facets are properly accounted for in any theory or predictions made. For example, certain dynamics allow for the loss of one facet and the merging of its neighbouring facets and so care must be taken to ensure any prediction of the distribution evolution takes account of such events [**80**]. Characterisation of these events is perhaps more akin to the sink-source characterisation found in Smoluchowski's coagulation equation [**116**]. Dynamic scaling permits the use of distributions as representative of the system at any relevant time.

Given a faceted system and an associated velocity law it is then the aim to predict the evolution of distributional quantities from some given distribution, which is simply representing the initial data. Of course, an initial condition which has a finite number of facets is amenable to simulation but the outcome would only be valid for that particular initial data, we want something more general than that and so try to predict the distributional makeup of the surface theoretically. Whilst this wouldn't tell us *exactly* what the surface would look like at some later time for any given data it would give us a good idea of the general structure of the surface. For instance, if we can predict the growth rate of the average size of a facet and know the distribution which the surface satisfies then we have something which is far more general than could be obtained by just running a finite simulation.

In many similar coarsening systems theoretical predictions on the growth rate of the system are often found and frequently have a power law type form [**60**, **62**, **83**]. For multi-scale continuum partial differential equations, such power laws emerge from spatio-temporal scale separation resulting in an effective *defect dynamics*. For a physics review on such dissipative PDE and their resulting $\mathcal{CDS}$, see Bray [**14**], and for a unifying analytical framework, utilising the Principle of Maximum Dissipation, see Watson [**121**]. The scaling laws then follow from simple scaling properties of these effective dynamics. We note that a *soft-analysis*, and so-called ansatz-free, version of this approach, which achieves only upper-bounds on the scaling laws, and provides no details on *defect dynamics* was initiated by [**60**], with other subsequent work appearing in [**27**, **61**, **83**].

The prediction of the scaling functions which govern the scale invariant distributions of a dynamically scaling $\mathcal{CDS}$ is a key theoretical challenge. The *Fokker-Planck* equation [**50**] governing an evolving distribution $\rho(l,t)$ is designed to do exactly this. It takes account of deterministic components of the flow but also takes into account any random perturbations that might be present such as noise or general fluctuations in object size. The one-dimensional Fokker-Planck equation for a distribution $\rho(l,t)$ is then,

$$\frac{\partial}{\partial t}\rho(l,t) = -\frac{\partial}{\partial l}[D_1(l)\rho(l,t)] + \frac{\partial^2}{\partial l^2}[D_2(l)\rho(l,t)],$$

where $D_1(l)$ characterises the drift or equivalently the deterministic part and $D_2(l)$ characterises a diffusional or 'noise' component. In our model each facet is driven in a purely deterministic manner and so in this circumstance we shall set the coefficient $D_2(l) \equiv 0$. All that would appear to remain is to characterise the coefficient $D_1(l)$. Unfortunately a problem with this very simple approach arises. Implicit in the equation as defined is that the evolution of the distribution of a particular length l, say, depends only on the current distribution of length and a coefficient, $D_1(l)$, which itself only depends on the individual length. Since the function $D_1(l)$ is only dependent on one length it cannot possibly take account of any neighbour relations or effects that surroundings have on the evolution of lengths, this function must capture what happens to all lengths of size l regardless of surroundings. One can interpret this as the function $D_1(l)$ containing information on what happens to a object of size l 'on average'.

All is not lost, however, as modifications of this basic form of the equation can be made which introduce coefficients depending on the wider surroundings as well as considering higher order distributions, $\rho(l_1, l_2, t)$ for example. This can however lead to an infinite hierarchy of such equations, namely the one-dimensional depends on the two-dimensional, two on three and so on [**11**, **47**]. In order to 'close' the system, in other words cut off this hierarchy at some point, it is generally necessary to make some form of mean-field type assumption.

In this context a mean-field assumption generally follows the same pattern as previously mentioned in Chapter 1, that is, we assume that neighbouring objects are independent or in some sense average. If we take a Fokker-Planck type equation for the one-point distribution, $\rho(l, t)$, and in the infinite hierarchy it depends on the two-point, $\rho(l_1, l_2, t)$, we might assume that the lengths within the two-point distribution are independent, and as a direct result of independence we then have $\rho(l_1, l_2, t) = \rho(l_1, t)\rho(l_2, t)$. Further to this, if due to the velocity law we require information about the surroundings to understand the evolution of an object but we want to have a closed system, then it may be necessary to assume that neighbours are in fact average in size to allow us to proceed with deriving predicted distributions. This kind of idea is quite common in many systems [**11**, **47**] including faceted ones [**80**] and fits neatly into the general concept of a mean-field model.

As a somewhat more concrete example, consider a system where a facet's evolution depends on the length of its immediate neighbours. If we want to develop a one-point theory that governs the distribution evolution then we need to understand how a facet moves on average and since we cannot make direct reference to the size of the neighbours for a closed theory we can then assume that the neighbours are, in an appropriate way, average in size.

Mean-field models in this sense provide exactly the kind of outcome we'd expect from the general concept. By assuming neighbours or indeed the surroundings are independent and/or mean in size we remove any dependence or correlation between the objects in the system and hope the reduced system following these assumptions is more amenable to prediction. It may well be, and often is the case, that without some sort of assumption we are left with the infinite hierarchy of equations or no sensible equations at all and so progress could be severely limited without some form of assumption. Of course in making such assumptions we may well be filtering out important information about the system and so we must take care that we don't impose too strict an assumption on the system, otherwise we risk generating a transport equation predicting the probability distribution evolution which bares no resemblance to the distribution we are trying to predict.

In some circumstances it may be inappropriate to perform the kind of independence assumption seen earlier. Such a situation may arise where the correlation between neighbours is high and as a result this approximation is going to be too strict, possibly limiting the usefulness of the prediction. Calculation of correlation coefficients is an efficient way to probe the relationships between objects and their surroundings. Mean-field type assumptions tend to be useful or appropriate when correlations are small or zero (zero would seem to indicate true independence) and so care must be taken to not be too quick to assume objects are independent. Later when we consider a particular facet model we will see exactly the situation where imposing 'full' independence on objects would severely limit the usefulness of the output.

1.3 Objectives and Approach

Our aim is to take a particular facet dynamics, which has its roots in a geometric PDE [**120**], and to predict the evolution of the system probability distributions, in particular we will focus on the two-point distribution of pairs of lengths. We begin by detailing the system in question which will essentially entail a system of facets with slope alternating between 1 and -1 which will create a saw-tooth like structure of hills and valleys. Following on from this we discuss the exact form of the facet velocity law where we see that the rate of growth of a facet is determined by the lengths of immediate neighbours with no reference to a facets own length. The particular form of the velocity law is shown, via both large scale simulation and heuristic argument, to exhibit coarsening. These coarsening events are not entirely random so we explore the precise detail of this loss from the system, and confirm that loss occurs in specific pairs, namely valleys [**120**].

We find that there exist two distinct two-point distributions, a 'hill' distribution and a 'valley' one. Arguing directly from the velocity law we assume dynamic scaling, and thereby confirm the scaling law of the average length to be $\langle l\rangle(t) \sim t^{1/2}$. Large scale numerical simulations of the system are then carried out which further reveal the shape and structure of the two-point probability distributions and confirms their dynamic scaling properties. We give detail on the method of simulation together with presentation and analysis of the data acquired. Probing the two-point distributions further we see that in both the hill and valley cases an assumption of independence of the side lengths is inconsistent with the numerical data; neither bares any resemblance to the product of the one-point distribution. Consideration of the correlation coefficients shows that pairs of adjacent lengths are highly correlated, which is reflected in the distributions found and helps explain the lack of resemblance.

Following on from our exploration of the model and associated data we proceed to our main aim, the prediction of the evolution of the probability distributions or, equivalently, the prediction of the universal scaling function. We show that, based on typical techniques, in our context a one-point theory becomes degenerate. Primarily this is because evolution proceeds by direct reference to neighbouring lengths. If both neighbours are deemed to be of average length then, in fact, there is no dynamics at all. Each and every length will remain the same in any prediction given which ultimately leads to spurious results. This outcome forces us to consider higher order distributions, in particular we focus on the two-point distributions. Here, unlike in one-dimension, there is hope of a general theory since the aforementioned degeneracy isn't apparent. We focus our attention on the valley distribution and by assuming that adjacent valleys are independent, our chosen mean-field assumption, we are able to derive the following equation for the scaled distribution $\hat{\rho}(z_1, z_2)$,

$$3\lambda\hat{\rho}(z_1, z_2) + \left(\lambda z_1 + \frac{1}{z_2} - \hat{\alpha}\right)\frac{\partial\hat{\rho}(z_1, z_2)}{\partial z_1} + \left(\lambda z_2 + \frac{1}{z_1} - \hat{\alpha}\right)\frac{\partial\hat{\rho}(z_1, z_2)}{\partial z_2} = 0,$$

where λ is related to the scaling rate, $\hat{\alpha}$ is a constant and z_i are the scaled lengths on either side of a valley. Numerically we see that the only sink from the system is at the origin, as we expected since we know valleys disappear as a whole.

Our mean-field assumption has one additional effect, it enforces a condition on the hill distribution namely that the sides of a hill are independent. Therefore by making this one assumption on the valleys we can predict both the hill and valley distributions, although the hill distribution has no separate equation to solve for, it is simply a consequence of the valley derivation. We note at this stage that the sole coarsening mechanism being the disappearance of valleys [**120**] results in a single sink in the system. This parallels the classic LSW theory where the one-dimensional radial distribution also has a single sink at the origin as small clusters disappear. Our system may thus be naturally viewed as a 2D generalisation of the LSW theory.

1.4 Summary

We begin in Section 2 where we discuss the dynamics of the system in question along with the particular coarsening rule. With this in mind we discuss our numerical simulations and confirm the scaling law of the average length in the system [**120**] before validating this numerically. In Section 2.9 we discuss the distribution and correlation data gathered and look at what conclusions can be drawn therein. Finally in Section 3 we consider our

Fokker-Planck type models to try and predict the two-point valley distribution. We then discuss a particular one-point reduction of the two-point model for which an associated evolution equation can be solved exactly.

2 The Hill-Valley Facet Model

2.1 Overview

The *Hill-Valley facet model*, derived in [**120**], consists of a series of initially random (positive) facets, l_i, where i is an indexing of the facets, $i \in \mathbb{Z}$, and such that each facet has a prescribed alternating slope. In particular, each facet has either slope 1 or -1 and the slope alternates as we go from length to length. Thus we have two types of vertex, a hill and a valley. The system then evolves in a deterministic manner, with each facet evolving subject to a fixed velocity law, $\mathcal{V}(l_i) := \mathcal{F}(l_{i-1}, l_i, l_{i+1})$, which is prescribed. Depending on the form of the velocity law there may be coarsening events where the system loses one or more of the facets before continuing on as before. As we shall see the particular velocity law used here does exhibit this coarsening behaviour.

The surface generated is therefore a piecewise linear function, $u(x)$, and could equally be described by indicating the points x_i on the x axis where facet joins occur and whether this indicates a hill or a valley.

2.2 Specific Dynamics and Velocity Law

As mentioned, each facet moves subject to a fixed velocity law. The particular velocity law we focus on here is as follows,

$$\mathcal{V}(l_i) := \mathcal{F}(l_{i-1}, l_i, l_{i+1}) = \frac{dl_i}{dt} = \mathcal{K}(-1)^i \left(\frac{1}{l_{i+1}} - \frac{1}{l_{i-1}} \right), \tag{2.2.1}$$

where i is even for up slopes and odd for down slopes, and is hence equivalent to multiplication by the slope of the particular facet, l_i, and the pre-factor $\mathcal{K}$ has units *length*^{2}/*time* ($[L]^2/[T]$). We see immediately the dichotomy between *up* and *down* slopes, due to the $(-1)^i$ term. This leads to interesting dynamics and in particular leads to the coarsening of the system.

2.2.1 Dynamics: The Dichotomy Effect

Consider the following two configurations, first a valley with symmetric side lengths, l, say, and the neighbouring lengths to the left and right of the valley are of length L, say, where $L > l$. Now let us consider the fate of this valley and, for now, assume the outer lengths are fixed. The velocity of the left and right lengths in the valley at this instance in time are,

$$-\mathcal{K}\left(\frac{1}{l}-\frac{1}{L}\right) \quad \text{and} \quad \mathcal{K}\left(\frac{1}{L}-\frac{1}{l}\right),$$

respectively. The main point to note here is that these are both negative, as a consequence of $L > l$, and so the valley shrinks in size. Since we have assumed that the outer lengths remain fixed it is clear to see that the valley will shrink with ever increasing speed.

Now consider a similar scenario but with a hill with lengths $l < L$, where L is the size of the outer lengths. By a similar argument the velocity of the hill facets are,

$$\mathcal{K}\left(\frac{1}{l}-\frac{1}{L}\right) \quad \text{and} \quad -\mathcal{K}\left(\frac{1}{L}-\frac{1}{l}\right),$$

respectively. Here we then see that these are both positive and so the hill grows.

Now we note that the assumption that the outer lengths were fixed was not consistent with the evolution of the system in which every length has some velocity and therefore evolves in time. We also note that in general a valley (hill) will not necessarily have equal side lengths. These assumptions aside we can see that, loosely speaking, small valleys will shrink and disappear whilst small hills are not favourable as the smaller they are in size relative to their neighbours the faster they will want to grow.

2.2.2 System Symmetries

The particular velocity law present here leads to interesting dynamics but further to this we note that the system displays a useful symmetry property which we will encounter frequently throughout the chapter. First we note that the system is *not* symmetric under a reflection in the horizontal x axes. Simply put this would swap valleys and hills around and cause a swap in the coarsening mechanism, namely hills would disappear.

The system *does*, however, display symmetry in any vertical line, i.e. any line parallel to the vertical solution axes, $u(x)$. Such a reflection wouldn't change the fate of any facet, since left and right lengths would swap but there would be an additional minus sign introduced through the swap in slope. As such the distribution of objects which we consider will reflect this symmetry property. As a direct consequence of this, for example, we see that there will only exist one one-point distribution regardless of orientation.

2.3 Finite Simulations

Eventually we would like to probe the statistics of this system as time proceeds. As a result we would like to simulate the system to acquire some data, however, the system lives on the infinite line and so isn't immediately amenable to simulation. In order to overcome this we propose a finite number of facets on the line and impose periodic boundary conditions on the system. As a result of this we are imposing that the system has the same total *up* length as *down* length. This is physically realistic to the full infinite system since any discrepancy would essentially mean the system gradually increases/decreases non-monotonically in one direction, which is not consistent with the system we are currently trying to describe. This shall be known as the *zero cut* case.

Introduction of a cut to the system means that the system tends to drift upwards or downwards. The case with some form of cut, e.g. more up length than down, can similarly be modelled but understandably the statistics of such a system may be altered from the zero cut case. We won't pursue the variation of cut any further here and will concentrate solely on the zero cut case.

In order for this finite system assumption to be accurate we must ensure that a *separation of scales* argument holds. In this context we assume that $\langle l\rangle(t)$, the average length in the system, is much less than the total length of the system, i.e. $\langle l\rangle(t) \ll \mathcal{L}$. In the finite system we have $\langle l\rangle(t) = \mathcal{L}/\mathcal{N}_\star$ where $\mathcal{N}_\star$ is the number of facets in the system and so we must have $\mathcal{L}/\mathcal{N}_\star \ll \mathcal{L}$ and hence $1 \ll \mathcal{N}_\star$. Thus as long as we have a sufficient number of lengths in the system we expect this separation of scale condition to hold and the system to mimic the behaviour of the infinite case.

2.4 Coarsening Events and Update

We have already mentioned that small valleys tend to shrink but we must now take care to characterise and understand the coarsening events which take place within the system. As a valley shrinks it will eventually shrink to zero and be removed, although this is only from a heuristic viewpoint. If we consider the analysis and proof of Theorems 8 and 9 presented by Watson et al. [**122**] we see similar dynamics there and that the argument carries over directly to the situation here. As a result we can conclusively say that the loss of a single length from the system is *impossible* and that the loss of a valley is the *only* method for which two adjacent lengths can shrink to zero, i.e. a hill *cannot* shrink to zero. These proofs are based on the assumption that we are in a fairly generic setup, that is, other lengths surrounding the shrinking valley remain large and so the valley in question is isolated from any other events. It is possible in a highly symmetric situation for there to be a loss of two or more consecutive valleys however this situation is unlikely in all but the most symmetric of situations.

Consideration of the structure of the problem leads us to the conclusion that when an event takes place the neighbouring facets still exist as separate entities which are joined at the point of extinction of the previously situated valley. Thus we have developed our general *coarsening rule*; valleys shrink to zero and are annihilated, whilst the neighbouring facets remain and are joined at the point of extinction.

2.5 Conservation of Total Length

We show that within this finite framework the total length of the system in preserved under the evolution described above in (2.2.1). We want to see if there is any change in the total length $\mathcal{L}$ as time proceeds, where,

$$\mathcal{L} = \sum_{i=1}^{\mathcal{N}_\star} l_i,$$

and $\mathcal{N}_\star$ is some even number of facets so as to be consistent with periodicity. We assume that the first length in the system is a down to keep our indexing consistent, then we see the following is true,

$$\begin{aligned}
\frac{d\mathcal{L}}{dt} &= \sum_{i=1}^{\mathcal{N}_\star} \frac{dl_i}{dt} = \sum_{i=1}^{\mathcal{N}_\star} (-1)^i \mathcal{K} \left(\frac{1}{l_{i+1}} - \frac{1}{l_{i-1}} \right), \\
&= \mathcal{K} \left[\sum_{i=2}^{(\mathcal{N}_\star - 1)} (-1)^i \left(\frac{1}{l_{i+1}} - \frac{1}{l_{i-1}} \right) + \left(\frac{1}{l_{\mathcal{N}_\star}} - \frac{1}{l_2} \right) + \left(\frac{1}{l_1} - \frac{1}{l_{(\mathcal{N}_\star - 1)}} \right) \right], \\
&= \mathcal{K} \left[\sum_{i=3}^{\mathcal{N}_\star} \frac{(-1)^{i-1}}{l_i} - \sum_{i=1}^{(\mathcal{N}_\star - 2)} \frac{(-1)^{i+1}}{l_i} + \left(\frac{1}{l_{\mathcal{N}_\star}} - \frac{1}{l_2} \right) + \left(\frac{1}{l_1} - \frac{1}{l_{(\mathcal{N}_\star - 1)}} \right) \right], \\
&= \mathcal{K} \left[\left(\frac{1}{l_{(\mathcal{N}_\star - 1)}} - \frac{1}{l_{\mathcal{N}_\star}} \right) - \left(\frac{1}{l_1} - \frac{1}{l_2} \right) + \left(\frac{1}{l_{\mathcal{N}_\star}} - \frac{1}{l_2} \right) + \left(\frac{1}{l_1} - \frac{1}{l_{(\mathcal{N}_\star - 1)}} \right) \right], \\
&= 0.
\end{aligned}$$

And so we have shown that the total length of the system is conserved under this evolution.

At this stage we could simulate the system for any total length $\mathcal{L}$, however we now show that we can non-dimensionalise the system to a canonical one.

2.6 Canonical Problem (Non-dimensionalisation)

In this section we discuss the non-dimensionalisation of the system as described above. To do this consider the following scalings of length and time,

$$l_i = L_i\mathcal{L}, \quad t = T\hat{t} \tag{2.6.1}$$

Inserting these scalings into the velocity law (2.2.1) we have,

$$\begin{aligned}\frac{\mathcal{L}}{\hat{t}}\frac{dL_i}{dT} &= \frac{\mathcal{K}}{\mathcal{L}}(-1)^i\left(\frac{1}{L_{i+1}} - \frac{1}{L_{i-1}}\right),\\ \frac{\mathcal{L}^2}{\mathcal{K}}\frac{1}{\hat{t}}\frac{dL_i}{dT} &= (-1)^i\left(\frac{1}{L_{i+1}} - \frac{1}{L_{i-1}}\right).\end{aligned}$$

By taking the free parameter $\hat{t}$, which is our scaling of time, and setting it equal to $\mathcal{L}^2/\mathcal{K}$ we are returned to the original velocity law, in non-dimensional form, namely,

$$\frac{dL_i}{dT} = (-1)^i\left(\frac{1}{L_{i+1}} - \frac{1}{L_{i-1}}\right). \tag{2.6.2}$$

Note that this scaling reduces the total length of the domain to $\mathcal{L}_\star = 1$ and gives us our canonical problem.

Canonical Problem:

Any problem of this form with total length of domain $\mathcal{L}$ and $\mathcal{N}_\star$ (even) facets can be reduced to the canonical problem of $\mathcal{N}_\star$ facets with unit total length, $\mathcal{L}_\star = 1$, via the scalings in (2.6.1). The form of the velocity law remains unchanged.

Therefore we have reduced a whole class of problems with varying length to just one problem, namely $\mathcal{N}_\star$ facets with total length $\mathcal{L}_\star = 1$,

$$\frac{dl_i}{dt} = (-1)^i\left(\frac{1}{l_{i+1}} - \frac{1}{l_{i-1}}\right), \qquad \mathcal{L}_\star = \sum_{i=1}^{\mathcal{N}_\star} l_i = 1. \tag{2.6.3}$$

2.7 The Scaling Hypothesis

In this section we present an analytical argument for the scale invariance of the system. To see if the system is indeed scale invariant consider the following scalings on length and time,

$$l \to \lambda l, \qquad t \to \lambda^q t.$$

What remains is to try and fix the value of q. We want to ensure that the system at a later time is statistically indistinguishable from the earlier time, so, in particular, we can think of λ scaling the lengths within the system so that the mean length is fixed. Consider these scalings in the velocity law (2.2.1) since we must consider how the system would evolve if the lengths were scaled to retain the statistical self-similarity property, hence,

$$\frac{\lambda^2}{\lambda^q}\frac{dl_i}{dt} = (-1)^i\left(\frac{1}{l_{i+1}} - \frac{1}{l_{i-1}}\right), \tag{2.7.1}$$

from which we can clearly see that to keep this invariant under these scalings that $q = 2$. Hence we have the following scalings of length and time,

$$l \to \lambda l, \qquad t \to \lambda^2 t.$$

Now, let us suppose that the mean length, $\langle l\rangle(t)$, is described by some function of time, $f(t)$, so that we have,

$$\langle l\rangle(t) = f(t).$$

This description must hold true after scaling the variables so the following must also be true,

$$\langle \lambda l\rangle(t) = f(\lambda^2 t),$$

which simplifies, since λ is just a number multiplying inside a mean, to

$$\lambda\langle l\rangle(t) = f(\lambda^2 t).$$

Now the number λ will vary depending on the state of the system, but we note that a particular choice of the value of λ simplifies the right hand side of the equation. By choosing $\lambda = t^{-\frac{1}{2}}$ we see that the right hand side reduces to some unknown constant,

$$t^{-\frac{1}{2}}\langle l\rangle(t) = f(t^{-1}t) = f(1).$$

From this we can easily read off our predicted scaling rate,

$$\langle l\rangle(t) \sim t^{\frac{1}{2}}. \tag{2.7.2}$$

2.8 Numerically Validating the Scaling Hypothesis

We present numerical data to validate our scaling rate (2.7.2). The system was simulated using $\mathcal{N}_\star$ = 2000, 5000 or 10000 facets and two different choices of initial condition as discussed later in Section 2.9.1. Data for the *current number of facets*, $\mathcal{N}(t)$, and *time*, t, were gathered. We simulated the problem in its canonical form and so the mean length of the system is simply $1/\mathcal{N}(t)$. In Figure 4.1 we see strong evidence to support the scaling hypothesis by plotting $1/\mathcal{N}(t) \equiv \langle l\rangle(t)$ against t on a log-log plot.

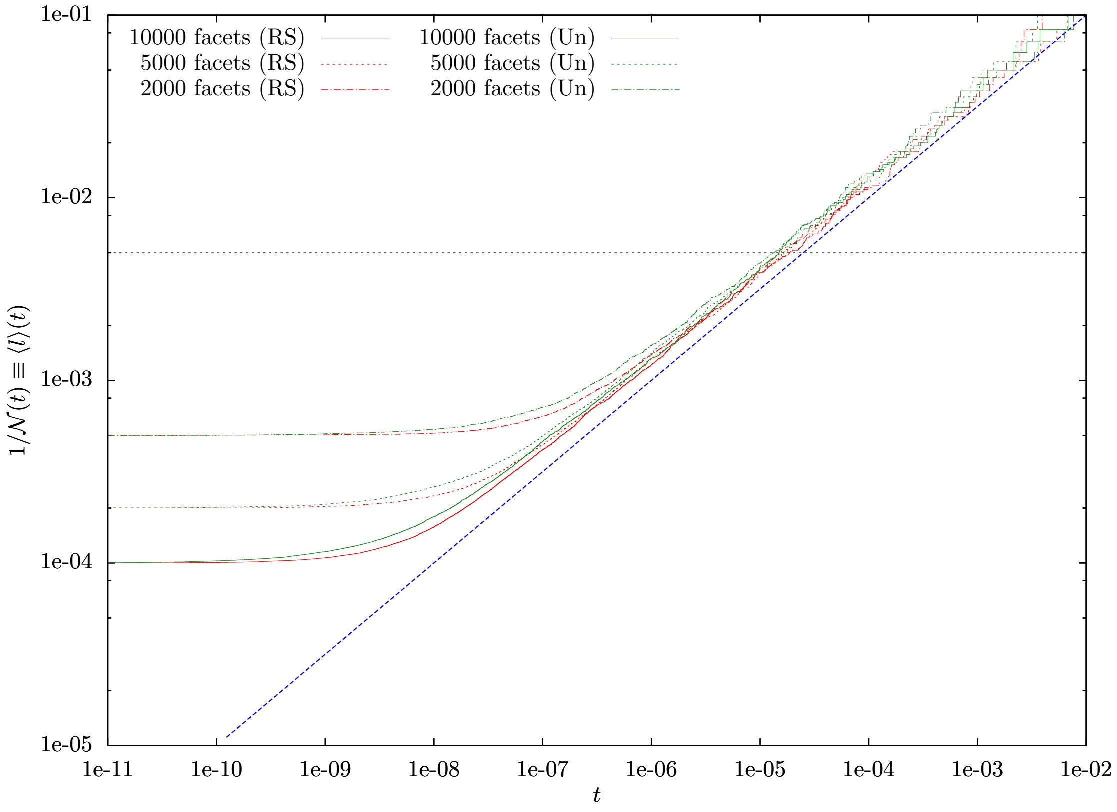


Figure 4.1 – $1/\mathcal{N}(t) \equiv \langle l \rangle(t)$ plotted against time, t, on a log-log scale for a selection of simulations. Data from two different initialisation methods, rejection sampling (RS) and Uniform (Un), is presented for comparison; more detail on these methods follows in Section 2.9.1. The dashed blue line is the predicted scaling rate $\langle l \rangle(t) = t^{\frac{1}{2}}$, showing a very good agreement to all the simulations after the initial transient phase. For computational efficiency we subsequently start individual simulations with two thousand facets and the black dotted line is then the time around which statistical data on the system is recorded; more detail on this can be found in Section 2.9.4.

2.9 Data Acquisition and Simulations

A significant amount of data was generated to both explore the system's intrinsic properties and also inform our theoretical predictions. We give detail on the precise nature of the simulations carried out before looking at the data itself and what information this reveals.

2.9.1 Initial Conditions

In order to initially seed the system we must select a set of lengths to be used. A set of equal lengths is an unstable steady state so we must, at the very least, perturb this situation in order to induce dynamics.

To begin with we chose to take a random number generator[1] with a uniform distribution of output on the interval $[0, 1]$, which we call $\mathfrak{U}(x)$. Repeatedly calling this number generator produces our list of lengths and we chose to add 1 to each length to ensure there were no very small lengths initially, hence we have a list of lengths each in the interval $[1, 2]$. We then scaled the problem into canonical form and in the process ensure that the total 'up' length is the same as the total 'down' length to keep the cut zero. This process produces a random initial condition each time a simulation starts and so we have a good degree of independence between each simulation.

From this initial condition we ran forty thousand simulations and took a mean of the distributions initially as well at at the later scaling recording time. It became clear that the initial condition we had chosen was far from the scaling distribution and so the program had to go through a large transient stage at the start of each simulation before arriving at the scaling regime, more detail on this together with associated figures can be found in Section 2.10.2. In order to try and reduce the time spent in this transient stage and get into the scaling regime as quickly as possible we chose to reconfigure the initial condition to make use of the *rejection sampling* method, using the late time distribution we had already found as the basis.

Rejection sampling works in the following way. We require three ingredients, the distribution we would like to match, $r(x)$, say, a simple distribution which is comparatively easy to sample from, in our case we take the uniform distribution as before, $\mathfrak{U}(x)$, and finally a constant M such that $r(x) \leqslant M\mathfrak{U}(x)$. Now we enter the following loop,

1. Select a point $Y \in [0, 3]$ by using $\mathfrak{U}(x)$ and then multiplying the output by 3.

[1] We made use of a freely available C++ random number generator based on the 'Keep It Simple Stupid' (KISS) method suggested by George Marsaglia in a *Usenet* posting from 1999.

2. Select a point $U \in [0,1]$ using $\mathfrak{U}(x)$.

3. Accept the value Y if $U \leqslant \frac{r(Y)}{Mg(Y)}$; else return to Step 1.

The basic principle of the method can perhaps be clarified by considering three separate cases. First, suppose that $r(Y) = M\mathfrak{U}(Y)$, then every choice of U will result in the condition in 3 being satisfied and the point being selected. This is clearly desirable since this must be a peak of the distribution from the condition $r(x) \leqslant M\mathfrak{U}(x)$. Next consider the case when $r(Y) = 0$. In this circumstance only an output of $U = 0$ from the uniform distribution will result in the point being selected, or, more loosely speaking, not very often at all, which is good since the probability distribution we are trying to sample is zero there. Finally suppose that $r(Y) = 0.5 \times M\mathfrak{U}(Y)$. We see that around half the time the condition in 3 is satisfied and half the time not. Thus this point is sometimes accepted and other times rejected. Generalisation of these examples helps understand how this method works.

We in fact take this procedure in two spatial dimensions and build up the profile one valley at a time. We make use of the final valley distribution as our initialisation distribution and repeat the loop until we have as many valleys as required. In doing so we ensure that our initial condition, at least for valleys, is much closer to the 'final' distribution we aim to find and so are reducing the amount of time spent in the transient phase. We will return to this point later in Section 2.10.2 where we will discuss the two-point distributions in more detail.

2.9.2 Update rule

The system of equations we wish to simulate, shown in (2.6.3), are highly coupled and so we simultaneously solve for the whole system using a standard fourth order *Runge-Kutta* method. The time step used is adaptive in nature and changes to reflect the size of the smallest length at that time, $l_{min}(t)$. In particular we found,

$$\Delta t = \min\left\{0.01, \frac{(l_{min}(t))^2}{5}\right\},$$

to be a good selection of adaptive time step to be used next at time t. The inclusion of '0.01' is to ensure that when there are only a few lengths left the time step is not too large, although in practise the second term in the minimum will generally control the size of Δt. We perform a check on the system after taking the time step Δt to ensure that no length has become negative, a sign of the time step being too large, and if necessary half the time step taken from the old values until we are satisfied a valid time step has taken place.

2.9.3 Annihilation tolerance

Motivated by our understanding that small valleys disappear we have chosen a tolerance, $\delta = 5 \times 10^{-6}$, and we then check after each time step to see if any valley has total extent less than δ. Should this be the case we pause the evolution and enter a system rewrite function which appropriately removes this valley from the system. In order to maintain the correct total length and cut we make sure that the left length in the valley, the down slope, is added to the length to the right of the valley, another down slope and similarly for the corresponding right length, an up slope. A typical example of this procedure can be seen in Figure 4.2.

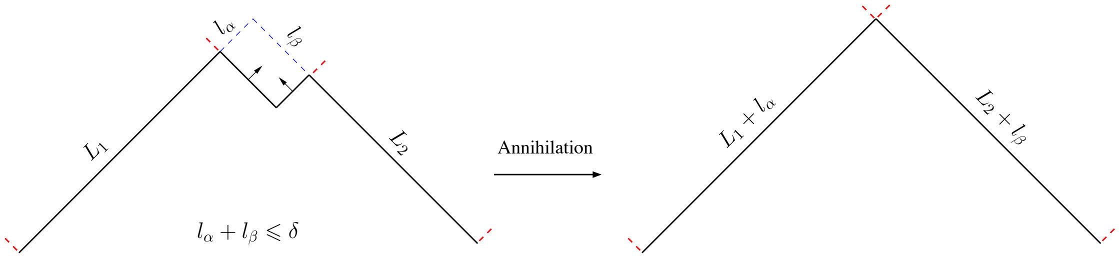


Figure 4.2 – Typical example of the annihilation procedure showing a small valley with total extent less that δ being removed and the corresponding lengths being appropriately added to the neighbouring lengths to create one large hill.

2.9.4 Data Collection and Scaled Variables

Data is gathered from simulations using a simple histogram type method. We collect data from the system when there are approximately only a tenth of the initial number of lengths remaining, $\sim \mathcal{N}_\star/10$. Justification of this choice can be seen in Figure (4.1) where we see that at this stage of the evolution it appears consistent scaling is taking place. There is a level of ambiguity with collecting data when exactly a tenth remains, for example, do we collect at the first time a tenth remains or when we are just about to drop below that value? To overcome this difficulty we instead make use of the predicted scaling law $\langle l \rangle(t) \sim t^{\frac{1}{2}}$, which we know closely matches the 'real' evolution, and the current clock time when two-fifths of the initial number remain, t_{curr}, say. From this time we then calculate the predicted time that we would have only a tenth remaining based on the scaling law and set this as the recording time t_{rec}, specifically,

$$t_{rec} = t_{curr} + \left(\left(\frac{1}{\left(\frac{\mathcal{N}_\star}{10}\right)} \right)^2 - \left(\frac{1}{\left(\frac{2\mathcal{N}_\star}{5}\right)} \right)^2 \right).$$

At the recording time specified we can simply calculate the mean length as $\langle l \rangle(t_{rec}) = 1/\mathcal{N}(t_{rec})$, since we are on a unit domain and so we have the following,

$$\langle l \rangle(t_{rec}) = \frac{1}{\mathcal{N}(t_{rec})}.$$

We expect, and can check numerically, that the system displays dynamic scaling of the distributions and so it is appropriate here, since we gather data at varying times throughout a simulation, that we scale the data against the average length at the recording time. Once the system has reached the appropriate time point, we take each length in the system and calculate the following value,

$$z_i := \frac{l_i}{\langle l \rangle(t_{rec})} = l_i \mathcal{N}(t_{rec}),$$

and so we are taking each length and seeing how it compares to the mean length of the system at the recording time. Thus regardless of the exact time of data collection the data we gather is scaled appropriately so that all the data across the many simulations we carry out can be taken and combined into one average data set.

To actually record the data we take a domain of size three, chosen from experience and simulation of the system, and subdivide into sixty equal pieces forming the base locations of histogram bars. We have chosen sixty as computationally we were limited when going to higher dimensional statistics, to be discussed later, and so felt for consistency it was sensible to use sixty 'bins' per axis regardless of the statistic being gathered. In summary, for the one point distribution we could have gone for a much higher number of data bins but, as we shall see in the subsequent figures, sixty is sufficient. As we calculate the values z_i we place 'one' into the appropriate bin of the histogram before scaling the total height of each bar by dividing by $\mathcal{N}(t_{rec}) \times b_w$, where $b_w = 3/60$ is the width of one bin, so that the histogram appropriately represents a probability distribution, i.e. the integral over z is 1. Once this procedure has been carried out we restart with a new initial condition and repeat. After each simulation a running average of the distributions is calculated and so we build up a picture of the ensemble statistics.

2.9.5 Higher order statistics

Unlike may other systems where concern is only placed on the one-point distribution of length/mass [**11**, **30**, **48**, **80**], we find in our case that the dichotomy in the velocity law leads to the emergence of interesting higher order statistics. By this we mean considering the probability of finding ordered pairs of lengths (l_1, l_2) and so on. We managed to collect data on the distributions as high as fourth order by extension of the method above, although only up to second order statistics are easily visualised and so discussion will be restricted therein. As noted earlier the computational extent of this procedure forced us to consider sixty bins per axis, but nevertheless the data is precise enough to draw various conclusions. In the next section we shall see, for example, that the two-point statistics for a valley are completely different from that of hills. We put forward some explanation of this and other characteristics of the distributions.

2.10 Data Analysis

As has been mentioned we ran a significant number of simulations, up to forty thousand in some circumstances, in order to build up the mean distributions we detail below. We will go into some detail of the output from these simulations and consider not only the distributions but also take a look at correlation data, which gives us some idea of how a facets evolution is governed by more than just its immediate neighbours.

We begin by considering the single point distributions for both 'up' lengths and 'down' lengths and comparisons between the two. Following on from this we will look at the two point distributions for 'hills' and 'valleys' and finally we will consider the correlation data.

2.10.1 One-point Distributions

The one-point distributions give us information on the spread of lengths within the system. The method of gathering data, detailed above, allows us to easily read off information such as the probability of finding a length which is smaller/larger than average. We initially consider the distribution of *up* lengths, $\varrho_u(z)$, calculated using the mean *up* length, $\langle l_u \rangle(t)$, and similarly for the down lengths, $\varrho_d(z)$. Note, however, that since we have zero cut the mean up length is equal to the mean down length, $\langle l_u \rangle(t) = \langle l_d \rangle(t)$.

In Figure 4.3 we see the mean one-point distributions for single lengths gathered from simulations. Earlier we mentioned the choice of 3 as the largest size compared to the mean we would consider in our distributions, the figure also demonstrates the reasoning behind that choice.

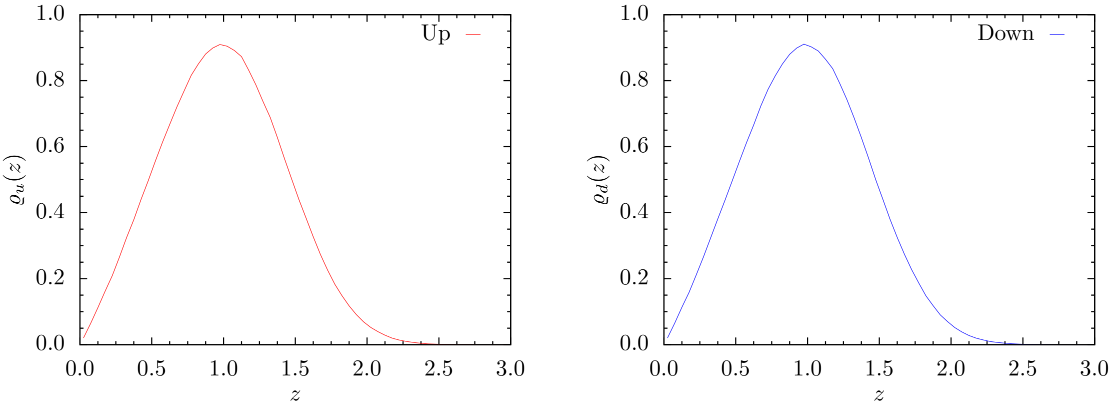


Figure 4.3 – The *up* and *down* distributions, $\varrho_u(z)$ and $\varrho_d(z)$ calculated from averaging the data from forty thousand independent simulations initialised using the rejection sampling method.

If we now consider the overlay of these two distributions, shown in Figure 4.4, we find that, in fact, there is only really one one-point distribution for the system. This is not surprising given our understanding of the evolution of the system and the symmetry properties mentioned earlier but these simulations nevertheless serve as a check that our program reflects this property. We therefore only have one one-point distribution to consider and from now on we will simply refer to this distribution as *the* one-point distribution, $\varrho(z)$. Repeating the simulations and gathering data for this single one-point distribution results in the final mean distribution, Figure 4.5.

2.10.2 Two-point Distributions

We now consider the possible two-point distributions which arise in the system. The two that we will consider are the hill distribution, $\varrho_h(z_1, z_2)$, which is the distribution of ordered lengths (z_1, z_2) such that z_1 is an up length and z_2 is a down length, and similarly we have the valley distribution $\varrho_v(z_1, z_2)$ where z_1 is an down length and z_2 is a up length. Note that under the noted symmetry condition a valley still remains a valley so we don't expect to see a single two-point distribution but indeed two distinct objects. They should, and indeed do, have a symmetry across the line $z_1 = z_2$, again as a result of the symmetry condition. We will comment further on these characteristics shortly but for now we simply refer to Figure 4.9 where we see the final result of these simulations.

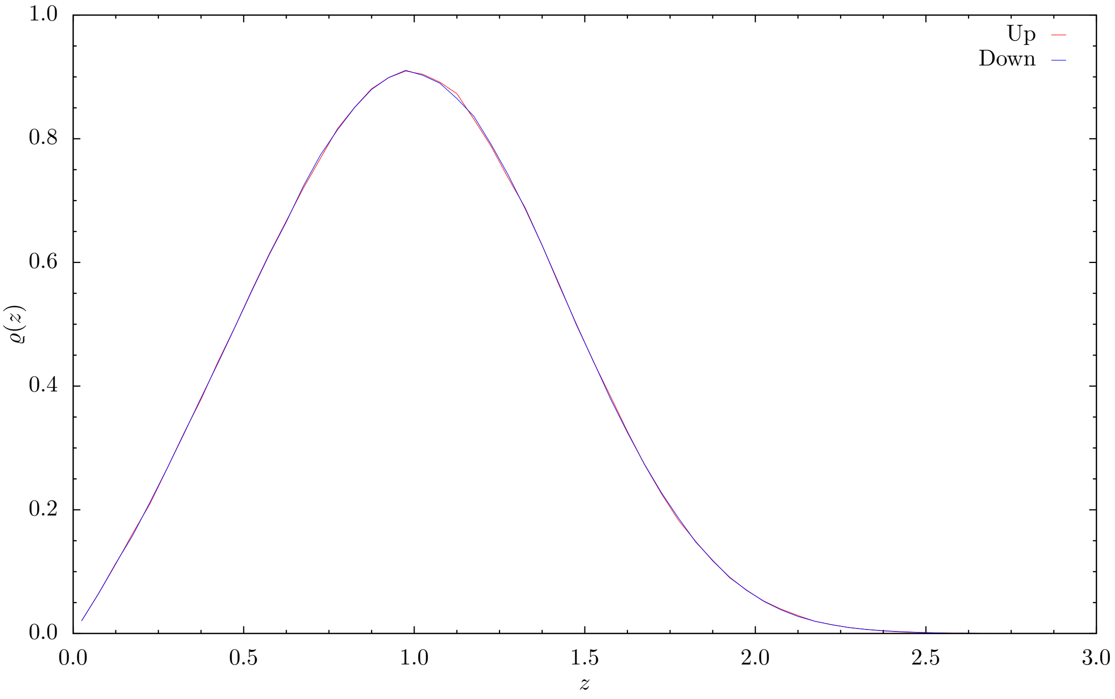


Figure 4.4 – The *up* and *down* distributions from Figure 4.3 overlaid showing the expected coincidence due to internal symmetries of the system.

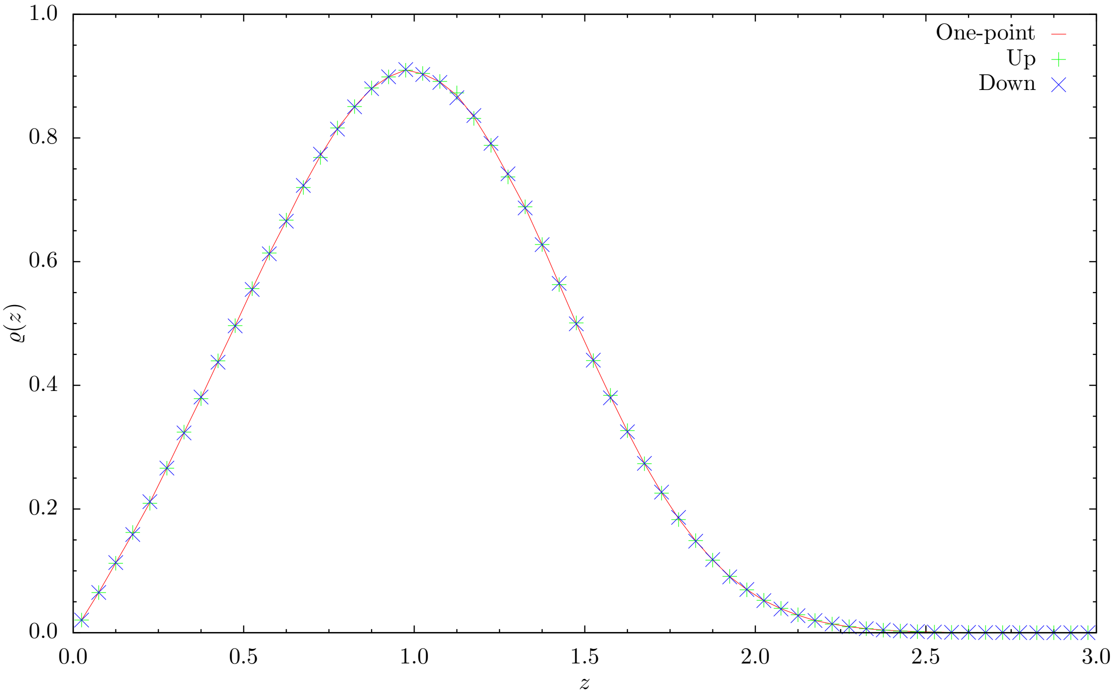


Figure 4.5 – *The* one-point distribution, $\varrho(z)$, together with the individual data from the separate up and down distributions plotted as points to emphasise the agreement.

At this stage we can clearly see the significant difference in the distributions for hills and valleys. The hill distribution is zero at the origin, which is consistent with our model analysis which told us that small hills want to grow as opposed to vanish, and so in general we don't find any small hills once the system has reached the scaling regime. We also note that the distribution approaches both axes at around 1.5 unlike the valley distribution which tends to be far from the axes other than near the origin. We interpret these regions close to the axes to be where there is a possibility of finding a hill with one large length and one corresponding small length, and further to this we see that this is indicative of the size of a hill when one of the neighbours is part of a vanishing valley. To put this another way, we infer from the distribution data that the size of the neighbouring lengths to a disappearing valley tend to be larger than average.

For the valley distribution we see a significantly different structure. The distribution shape is much narrower and has non-zero value all the way into the origin. This was to be expected since we know small valleys disappear and so on the whole we expect there to be small valleys in the system which are on the way to extinction. Perhaps surprisingly the narrow structure also suggests that valleys tend to be more 'even' in size, by this we mean both lengths are comparable in size. There is certainly not the same kind of spread in lengths when compared to the hill distribution. If we consider the hill distribution closely we see, as already mentioned, that we can have very asymmetric hills, in particular when a hill is losing one of its lengths during the coarsening process.

Of interest is a consideration of the ratio of lengths in a valley as they approach zero. It is not clear from the distribution whether 'most' valleys disappear in an even manner i.e. along the $z_1 = z_2$ line? Later we will consider these distributions in polar coordinates and we will return to this question at that stage. The distribution does certainly suggest that valleys disappear only via the origin and that there isn't any procedure in which a valley can disappear in any other manner.

Return to Initial Conditions

We now briefly return to the initial conditions for the simulations and consider the distributions at this time with knowledge of the final state seen in Figure 4.9. As already mentioned we first ran the simulations with a random generation of the surface from a uniform distribution. The hill and valley distributions after forty thousand runs can be seen in Figure 4.6 where we see the expected square shape of the initial distribution. This is as we would expect but we note that it is quite distinct from the final distribution which we are trying to capture, seen in Figure 4.9. Whilst we require a lack of sensitivity to initial conditions to have a true scaling state it is the belief that this initial condition is computationally inefficient which lead to the use of rejection sampling instead.

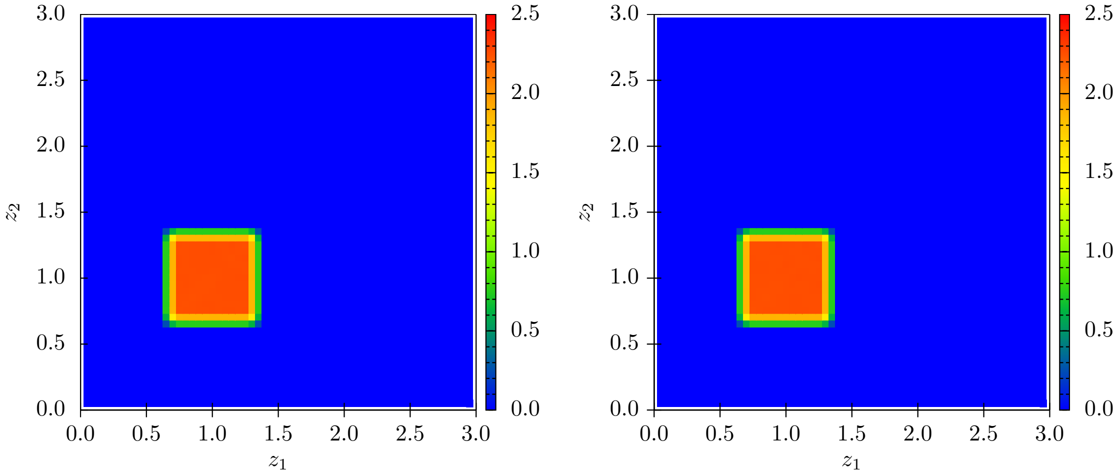


Figure 4.6 – The initial hill and valley distribution after forty thousand runs using the uniform initialisation method. As expected there is no difference in the initial distributions since at initialisation all lengths are independently drawn from a uniform one-point distribution and so immediately the two-point hill and valley distributions are simply the product of the uniform one-point distribution appropriately scaled.

In Figure 4.7 we see the same type of diagram but this time making use of the rejection sampling method of initialisation. The valley distribution obtained from forty thousand uniform initialisation simulations is the distribution which we now use to initialise. By using this information we expect that the valley distribution will remain very close to its initial form after the simulations have taken place, however, we have not referenced the hill distribution at all and so it is unclear, perhaps, what output to expect for the initial hill distribution? All we know is that our initialisation step creates valleys subject to the valley distribution but independently of each other. Since a hill is at the join of two valleys we are imposing that the sides of a hill are independent and this independence would suggest that the initial hill distribution, $\varrho_h^i(z_1, z_2)$, can be factorized into the product of two one-point distributions, $\varrho_h^i(z_1, z_2) = \varrho(z_1)\varrho(z_2)$. This product distribution can be seen in Figure 4.8.

If we now reconsider Figure 4.7 we see that the valley distribution does indeed remain very close to the initial distribution and that the hill distribution is very close to the product distribution, which can be seen in Figure 4.8.

With the two-point distributions now fairly well understood we now take a closer look at the valley distribution. The narrow region close to the origin is where valleys are as they disappear from the system but the resolution of the plot is insufficient to really understand the underlying structure here. Visually it is difficult to tell what kind of events are taking place. As already mentioned, are 'most' annihilation events along the symmetry line $z_1 = z_2$? To probe this question and others further we choose to not only bin data in cartesian co-ordinates, the scaled l_1, l_2-plane, but we also consider the distributions in polar co-ordinates r and θ, where $r = \sqrt{z_1^2 + z_2^2}$ and $\theta = \tan^{-1}(z_2/z_1)$. We could simply map the current distribution into the polar space, but this would lead to very little resolution in the area close to the origin (Consider that in the region $[0, 0.25] \times [0, 0.25]$ that there are only $5 \times 5 = 25$ bins.) If, instead, we re-run the simulations with a modification for polar co-ordinates we can insist on a much higher resolution around the origin.

Polar Co-ordinates

To collect data in polar co-ordinates we perform a similar procedure to the previous one but instead break the two polar axes into 60 bins and then calculate the polar length and angle of a valley/hill. The range of values used for collection of data are $\theta \in [0, \pi/2]$ and $r \in [0, 3\sqrt{2}]$, these were selected in order to cover at least the same area as the cartesian method. Since the lengths must be scaled, as performed earlier, the radial component must also be compared to the mean length of the system. The angle however does not require any scaling as this is invariant under the scaling of length. Performing this method of data collection we now have >180 bins in the quarter ball of radius 0.25 around the origin, a significant increase of resolution in the area of interest. In Figure 4.9 we see the hill and valley distributions in both cartesian and polar co-ordinates, where we have drawn the polar version with fixed axes.

We can now clearly see the benefit of polar co-ordinates by looking at the region close to the $r = 0$ line, as this is equivalent to the area around the origin in the previous plots. By gathering data in this manner we have managed to 'spread out' the information here, allowing for a better insight into the distribution of smaller valleys. We also note from this data that the distribution doesn't drop down significantly very close to the origin, which appears to be the case in the cartesian plot, in fact, the distribution maintains a larger

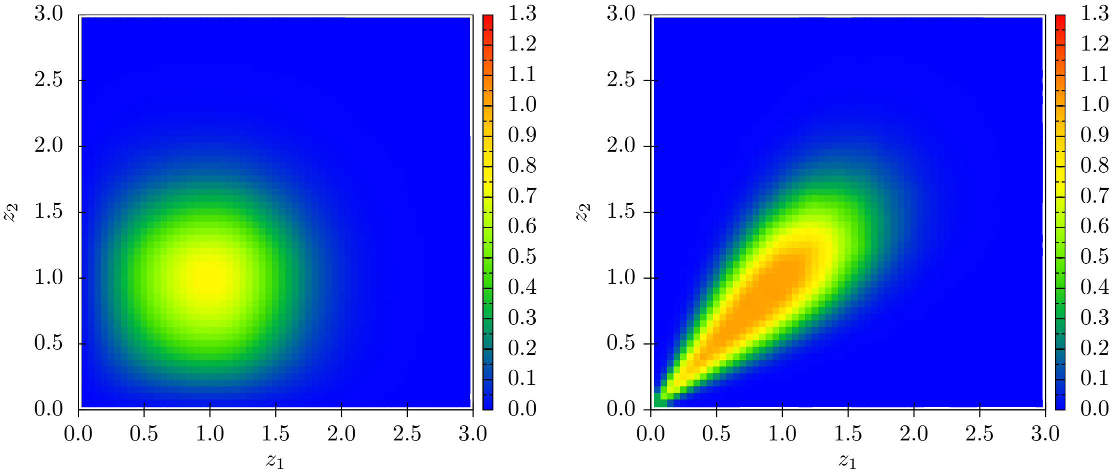


Figure 4.7 – The initial hill and valley distribution after forty thousand runs using the rejection sampling method of initialisation. Since we used existing valley distribution data, and didn't reference any hill data, we expect the valley distribution to more closely match the final distribution, as it does. Hills are created at the join of independent valleys and so we expect the initial hill distribution to resemble the product of the one-point distribution. See Figure 4.8

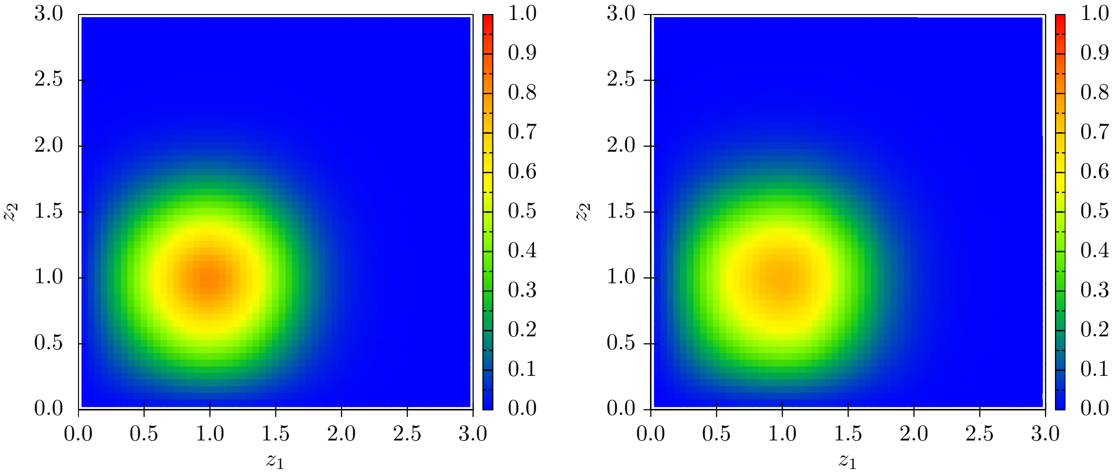


Figure 4.8 – The distribution found from the product of the one-point distribution $\varrho(z)$ with itself shown on the left and the initial hill distribution using rejection sampling redrawn from Figure 4.7 for comparison on the right. This agreement was to be expected since we seed individual valleys independently and consequentially hills are assumed to have independent side lengths. This independence then implies that the two-point is the product of the one-point marginal distribution as we see above.

value all the way into the origin and the cartesian method of data collection is perhaps slightly misleading here.

The polar plot of the valley distribution also highlights the possible variation in the ratio z_2/z_1, which is related to the angle θ through the relation $\theta = \tan^{-1}(z_2/z_1)$. This ratio gives us some idea of whether or not valleys disappear symmetrically and, in fact, the distribution seems to suggest that close to annihilation a valley can have quite a wide range of angle. In other words a valley can disappear with asymmetric side lengths, $z_1 \neq z_2$.

Functional form at $r = 0$

We have already mentioned the non-zero nature of the area close to the origin but would like to better understand this region and what particular form the distribution takes there. In order to achieve this we take the first, third and seventh closest histogram bins to the $r = 0$ line and from these extrapolate the value the distribution takes at precisely $r = 0$, we will call this function $\Phi(\theta)$. The data in the region close to the origin is quite rough since there aren't many small valleys at a given time and so we have made use of the symmetry line $z_1 = z_2$ to improve the resolution somewhat, this is evident by the symmetry of the curve around the $\pi/4$ mark. For comparison we have found the curve $f(\theta) = 0.08 \sin^8(2\theta)$ to be a very good fit and show this and the data curve together in Figure 4.10.

2.10.3 Conclusions From Distribution Data

The data gathered can be seen to confirm several aspects of the distributions which could have been anticipated by considering the specific dynamics. In particular, since there is an intrinsic symmetry in switching up facets and down facets, alternatively by mirroring the surface in any vertical line, we expect to find only one one-point distribution and this is confirmed by the numerical simulations, Figure 4.5. This same symmetry leads us to believe that the two-point distributions will be symmetric along the line $z_1 = z_2$ which is again confirmed by the simulations. Finally we note that the two-point distributions are significantly different, as we would expect due to the dichotomy in the velocity law, but further to this we note that neither of the distributions match the product of the one-point distribution seen in Figure 4.8. This implies that the lengths involved in both a hill or a valley are not independent of each other and are in fact correlated in some way. We test this correlation in the next section to confirm this assertion.

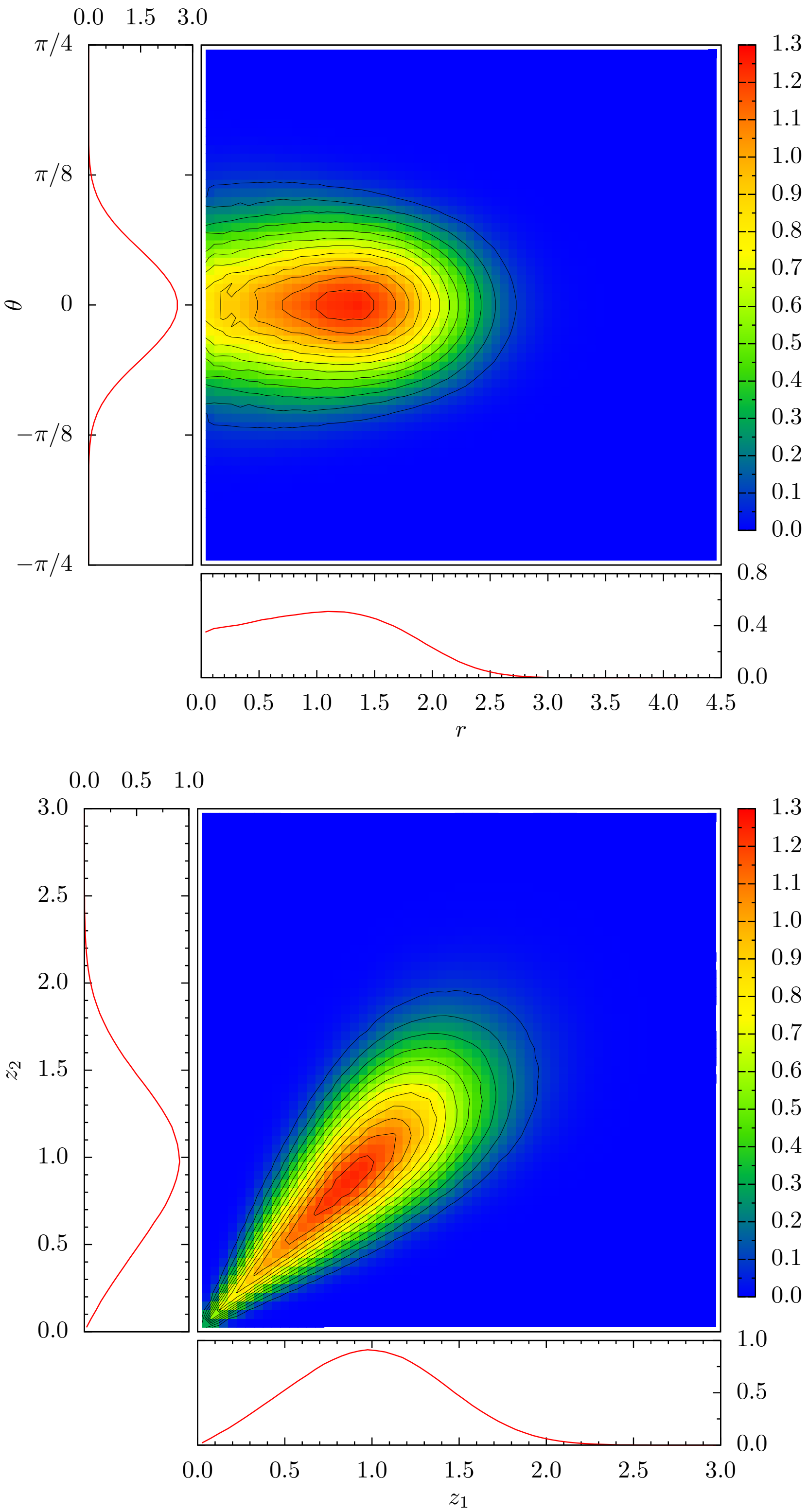


Figure 4.9 – The valley distribution in both polar (top) and cartesian (bottom) coordinates where the polar axes has been set to be the symmetry line $z_1 = z_2$ in the cartesian plot. Attached to each axes we have plotted the one-point marginal distributions found by integrating the data as appropriate. Contours on the two-point distribution are every 0.1 units of height. (Continued over page.)

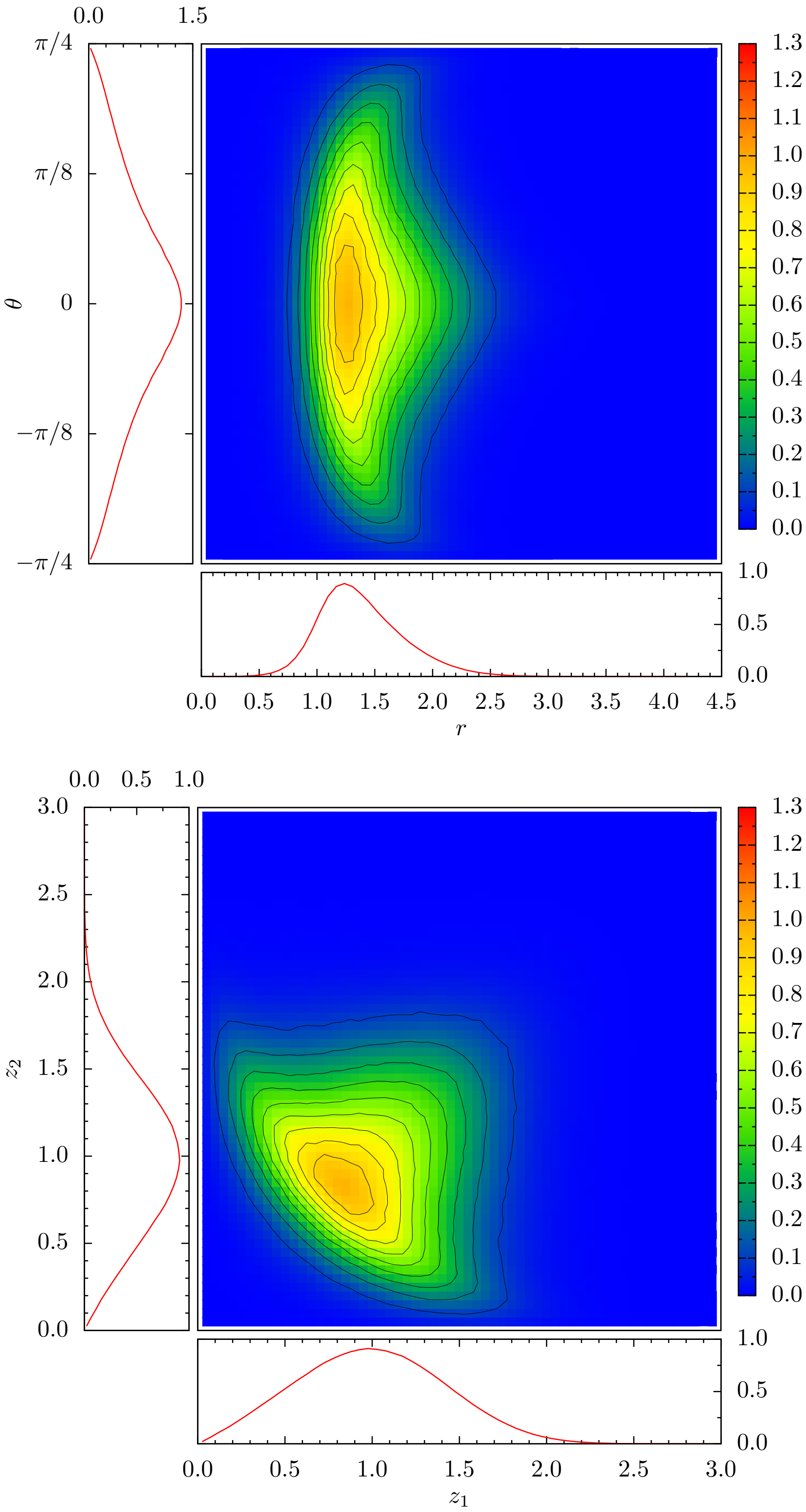


Figure 4.9 Continued – Corresponding hill distributions again in both polar and cartesian coordinates.

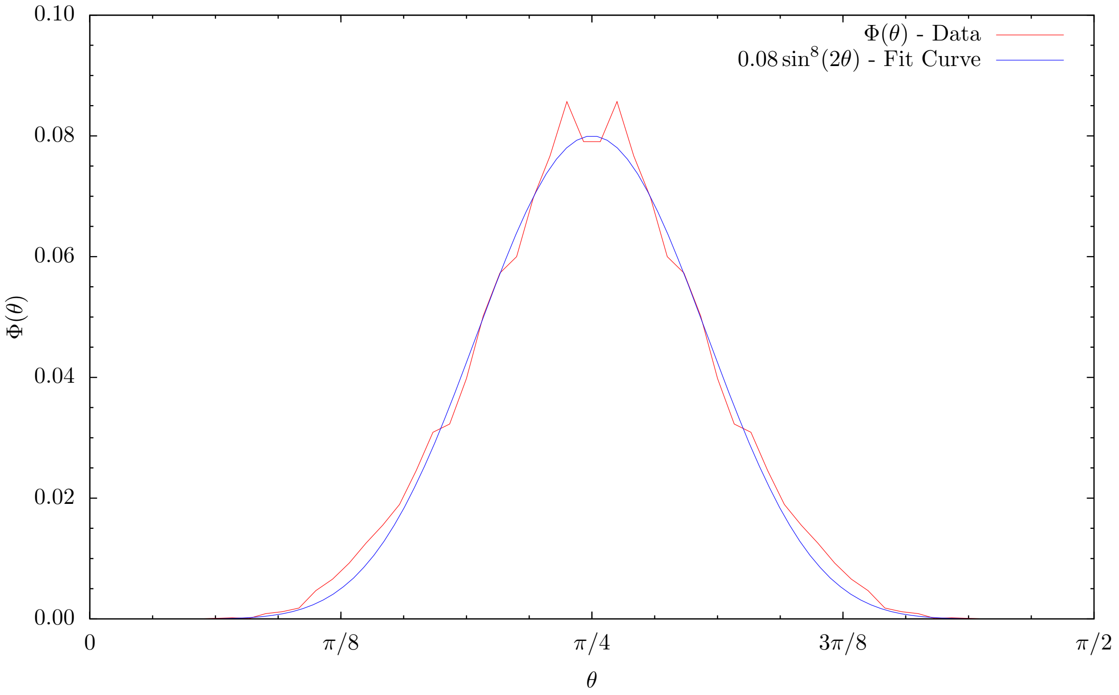


Figure 4.10 – The function $\Phi(\theta)$ found by numerically taking the limit $r \to 0^+$ in the polar distribution together with the fit curve $0.08\sin^8(2\theta)$.

2.10.4 Correlations

Correlations allow us to probe the independence of random variables and possibly allows us to infer an understanding of the relationships within the system. In our case we actually have several variables which we can consider the correlations of, such as an individual length to its neighbours, length of a valley/hill to its neighbours etc. We have already seen that the hill, and indeed the valley, distribution is not identical to the product of the one-point distribution. This leads us to suspect that the lengths within the system are highly correlated to each other, in particular we can clearly see from the distributions that the lengths of a valley and a hill are not independent. To investigate this further we again consider the *Pearson product-moment correlation coefficient*, $r_{X,Y}$, defined as,

$$r_{X,Y} = \frac{\mathbb{E}(XY) - \mathbb{E}(X)^2}{\mathbb{E}(X^2) - \mathbb{E}(X)^2}.$$

All the expected values can then simply be replaced by appropriate averages, since there is no weighting of particular outcomes, and so we can very easily expect the simulations to calculate this value as is proceeds. At the same time as we record data for the distribution we gather data on various correlations and calculate the correlation coefficient for that run. As we complete each run of the system we take an appropriate average of the same coefficient from multiple runs to build up a better picture of the correlations in the 'full' system and by doing so we generate one graph depicting the correlations in question.

First we consider the correlation between an individual length of particular type, e.g. downs, and its neighbouring lengths. We consider the correlation of a length with itself, which is equal to 1 since they are in perfect correlation, and then we also compare a length to immediate neighbours and then two, three away etc. We index the number of lengths away to the right, from the current length, by j. The specific formulation is as follows where $\hat{N}$ is the number of lengths being considered (e.g. in the case of downs we'd have $\hat{N} = N(t_{rec})/2$ since half of all lengths are down),

$$r_{L_i,L_{i+j}}(j) = \frac{\dfrac{1}{\hat{N}}\displaystyle\sum_{i=0}^{\hat{N}} L_i L_{i+j} - \dfrac{1}{\hat{N}^2}\left(\displaystyle\sum_{i=0}^{\hat{N}} L_i\right)^2}{\dfrac{1}{\hat{N}}\displaystyle\sum_{i=0}^{\hat{N}} L_i^2 - \dfrac{1}{\hat{N}^2}\left(\displaystyle\sum_{i=0}^{\hat{N}} L_i\right)^2},$$

which will return a number for each j which is interpreted as the correlation coefficient between a length and its neighbour j away. Note that as expected if $j = 0$ we are simply calculating the correlation of a length to itself and since these lengths are always equal the are completely dependent, or perfectly correlated. We choose to perform the same procedure with up lengths and down lengths only and consider the results side by side. In Figure 4.11 we see the output after five thousand simulations and the correlation coefficients averaged over the runs.

The correlation data provides several interesting pieces of information about the relationships within the system. First let us consider the $j = 1$ data which is considering the immediate right neighbour to a corresponding facet. The down data suggests a very strong correlation between a down facet and its neighbour, which forms a valley. This is perhaps not surprising since we can see at a glance from the two-point valley distribution that most valleys are quite close to the symmetry line and so tend to have quite similar side lengths. The hill distribution, on the other hand, is quite spread out suggesting that individual hills tend not to be as symmetric which is reflected in the correlation data which suggests a much weaker relationship between an up slope and its corresponding right neighbour.

At the $j = 2$ level we see that both up and down correlation coefficients are very similar but quite weak in size. This would suggest that for any length the neighbour two away is generally correlated in some way though this effect is relatively weak but not negligible.

The fact that the coefficients are the same is likely just a consequence of the symmetry conditions displayed by the system i.e. a down facet to the next down facet should be linked in the same way as up to up.

For $j \geqslant 3$ we see that the coefficients become much smaller and almost zero in size. This suggests that for neighbours three or more away there is little correlation between the lengths. A zero correlation is indicative of independence though this condition alone is not sufficient to prove independence.

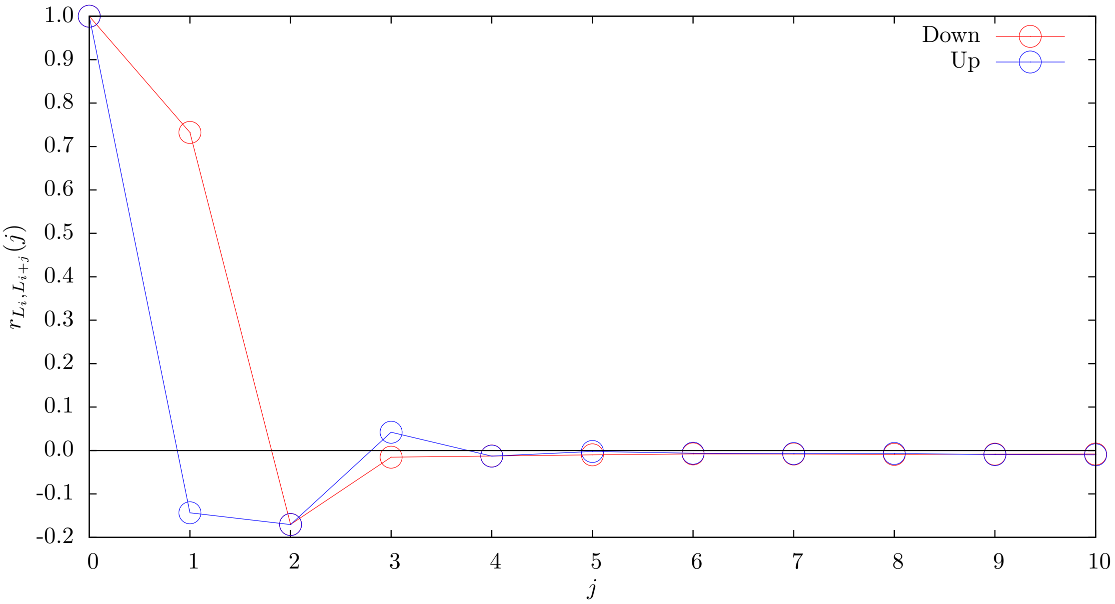


Figure 4.11 – Correlation coefficients for $0 \leqslant j \leqslant 10$ where j is the number of lengths away for both up and down lengths found by averaging data over five thousand independent simulations. Note the large correlation between a down length and its right neighbour (forming a valley), small but significant correlations otherwise up to two lengths away for either up or down, and relatively small correlations further away. Identical correlations two lengths away is as a result of the symmetries of the system.

3 The Fokker-Planck Equation and Mean-Field Models

The Fokker-Planck equation is a general framework for the evolution of a probability distribution. Our system is purely deterministic in nature and so we neglect any noise or diffusion terms in the full, general equation. What remains to be determined is what velocity the distribution evolves under. Generally this is related to the dynamical system velocity but in our case we have a velocity law that is not self contained, that is, a valley does not evolve based on information about itself alone but requires information about the surroundings. This forces us to consider higher order statistics than just the two-point valley distribution and thus we no longer have a 'closed' system. This type of infinite hierarchy, where the equation requires information on higher order statistics, is often found in the literature [**11**, **30**]. As is generally the case we must make some sort of mean-field type assumption to close the system and make it amenable to solution. We will follow this typical pattern and derive a two-point Fokker-Planck type equation for the scaled valley distribution.

3.1 The One-point Degeneracy

We first give some brief detail on the outcome of trying to find a prediction of the one-point distribution. Since there is only one one-point distribution we choose to consider a down length for illustration. We suppress a significant portion of the detail of the derivation since it is more informative to follow this through for the two-point distribution. Essentially here we only need to consider the velocity of a typical down length,

$$\frac{dl}{dt} = \left(\frac{1}{l'} - \frac{1}{l''}\right),$$

where l' is the length to the left of our down length and l'' to the right. Now, since the Fokker-Planck equation requires a self contained velocity law we must remove any reference to the neighbouring lengths by some means. The usual approach in this circumstance is to effectively assume that the neighbouring lengths are of average length, but therein lies the problem. By taking this approach here we force the velocity of a typical length of any size to be zero and thus under this assumption there are no dynamics at all.

The particular form of our velocity law is such that taking the typical mean-field type approach here results in a degenerate theory of evolution and so we refrain from pursuing this avenue any further. Later we consider a significantly different one-point theory as a reduction of the full two-point theory but for now we simply consider the traditional one-point approach to be unfruitful with regards to our particular dynamics.

We now pursue a Fokker-Planck type equation for the two-point valley distribution which doesn't appear at a glance to degenerately reduce as in the one-point case.

3.2 The Fokker-Planck Equation for Valleys - Traditional Method

We develop an evolution equation governing the two-point valley distribution following the 'traditional' Fokker-Planck type derivation as comparable to methods seen in, for example, [**11**, **48**]. We take the standard Fokker-Planck equation and modify it to take account of the particular dynamics present in our model. We begin with the number density, n, since it is more straightforward to account for any sinks or sources by a simple accountancy argument. In our case we know that valleys disappear as a whole which corresponds to a sink at the origin. No particular additional terms are required in the equation since this is simply a flux out of the region of interest and is captured automatically.

We begin by considering the number density of valleys with left length between l_1 and $l_1 + dl_1$ and corresponding right length between l_2 and $l_2 + dl_2$ at time t, we denote this by $n_2^V(l_1, l_2, t)dl_1 dl_2$. From this we can derive that the total number of valleys at time t, $\mathcal{N}(t)$, in the system is,

$$\mathcal{N}(t) = \int_0^\infty \int_0^\infty n_2^V(l_1, l_2, t)\; dl_1 dl_2. \tag{3.2.1}$$

In a similar manner we can also define the single length number density, $n(l,t)dl$, and the number of single lengths between l and $l + dl$, noting that there is only one density of this type by the symmetry condition mentioned earlier. Finally there are two triple length number densities defined as $n_3^u(l_1, l_2, l_3, t)dl_1 dl_2 dl_3$ and $n_3^d(l_1, l_2, l_3, t)dl_1 dl_2 dl_3$ with appropriate intervals of lengths given by the dl_i and the superscript u or d indicating the orientation of the first length reading left to right. By the same symmetry argument we can see that there is, in fact, only one type of triple length number density since reading one type from right to left simply gives us information on the other. The symmetry condition is therefore given by,

$$n_3^u(l_1, l_2, l_3, t)dl_1 dl_2 dl_3 = n_3^d(l_3, l_2, l_1, t)dl_1 dl_2 dl_3,$$

that is, the number of triples can be, in a number density sense, interpreted as the same whether read left to right or vice-versa. There is then only really one triple-point density to consider though care must be taken to ensure we are consistent in our calculations, as a result we will maintain the two triple-point notation for clarity.

We draw statistical data from a finite system though our aim is to replicate the statistics of an infinite system. As such these densities should be interpreted on a density per length basis. For consistency we choose this length to be the same as used in numerical simulations denoted by $\mathcal{L}$. From this we can easily write down another way of interpreting the total number of valleys as,

$$\mathcal{N}(t) = \frac{\mathcal{L}}{2\,\langle l \rangle(t)}, \tag{3.2.2}$$

where $\langle l \rangle(t)$ is the average length in the system, regardless of orientation, and the factor 2 accounts for the fact a valley contains 2 lengths. With these definitions in mind we write down the standard Fokker-Planck type equation, assuming only a deterministic part, for the evolution of this number profile, namely,

$$\frac{\partial}{\partial t} n_2^V(l_1, l_2, t) = -\frac{\partial}{\partial l_1}\left[\int_0^\infty \mathcal{V}(l_1) n_3^u(l', l_1, l_2, t)\, dl'\right] - \frac{\partial}{\partial l_2}\left[\int_0^\infty \mathcal{V}(l_2) n_3^d(l_1, l_2, l'', t)\, dl''\right], \tag{3.2.3}$$

where,

$$\mathcal{V}(l_1) = \frac{1}{l'} - \frac{1}{l_2} \quad \text{and} \quad \mathcal{V}(l_2) = \frac{1}{l''} - \frac{1}{l_1},$$

are derived directly from the $\mathcal{CDS}$ velocity law and we see that the n_3 are such that the lengths l_1 and l_2 form the valley and the additional lengths are then attached appropriately.

At this point we note that due to the dynamics of the system it is not sufficient to only know about the lengths l_1 and l_2 but we also require information about the neighbouring lengths, which are probabilistically captured by the use of the n_3. By taking the integral over all possible lengths l' or l'' together with the appropriate n_3 triple we are effectively calculating a mean of all the possible velocities.

In particular, consider a valley which has lengths l_1 and l_2. We do not know with this information alone how this valley will evolve, indeed its evolution is entirely dependent on the surrounding system. To overcome this and to reduce the system to one which can be considered as deterministic (a prerequisite of this form of the Fokker-Planck equation) we must make some form of assumption on the surroundings. By integrating the corresponding velocity and appropriate n_3 against the unknown length l', for example, we can consider the unknown lengths as being average in size. This is an important point to note, however, since this type of derivation and Fokker-Planck type approach therefore makes use of a mean-field approximation to ensure that we can proceed to flow the distributions.

3.2.1 Scaling Variables

We now propose a scaling solution to equation (3.2.3) and find our scaling solution formally by scaling the unknown $n_2^V(l_1, l_2, t)$. Similar scalings will also be performed on the one and three-point distributions where necessary. First we scale with respect to the total number of valleys in order that our predicted solution represents a true probability distribution, namely,

$$n_2^V(l_1, l_2, t) = \mathcal{N}(t)\rho_2^V(l_1, l_2, t).$$

It is easy to see the probability nature of the object $\rho_2^V(l_1, l_2, t)$ by integrating both sides with respect to l_1 and l_2 since in doing so the left hand side becomes $\mathcal{N}(t)$ and the right hand side becomes $\mathcal{N}(t)$ times the integral of ρ. Next we anticipate, and have seen evidence numerically, that a scaling solution to the problem does exist. Scaling the individual lengths by the mean length and introducing an appropriate factor of $1/\langle l\rangle^2(t)$ in front of the distribution ensures we are still dealing with a probability distribution, integral 1, but we have removed any dependence on time. Performing this scaling we have,

$$n_2^V(l_1, l_2, t) = \mathcal{N}(t)\rho_2^V(l_1, l_2, t) = \frac{\mathcal{N}(t)}{\langle l\rangle^2(t)}\hat{\rho}_2^V(z_1, z_2), \quad z_i := \frac{l_i}{\langle l\rangle(t)}.$$

Explicitly calculating the integral of both sides with respect to l_1 and l_2 ensures that $\hat{\rho}_2^V(z_1, z_2)$ does represent a probability distribution,

$$\begin{aligned}\int_0^\infty\int_0^\infty n_2^V(l_1, l_2, t)\, dl_1 dl_2 &= \int_0^\infty\int_0^\infty \frac{\mathcal{N}(t)}{\langle l\rangle^2(t)}\hat{\rho}_2^V(z_1, z_2)\, dl_1 dl_2,\\ \mathcal{N}(t) &= \mathcal{N}(t)\int_0^\infty\int_0^\infty \frac{1}{\langle l\rangle^2(t)}\hat{\rho}_2^V(z_1, z_2)\ \langle l\rangle^2(t) dz_1 dz_2,\\ 1 &= \int_0^\infty\int_0^\infty \hat{\rho}_2^V(z_1, z_2)\, dz_1 dz_2.\end{aligned}$$

Note that if we were to explicitly calculate the integral with respect to one or the other variables on the right hand side we would get the marginal one-point distribution which we shall denote $\hat{\rho}(z)$. As a direct consequence of the above equation it is clear to see the following,

$$\begin{aligned}1 &= \int_0^\infty\int_0^\infty \hat{\rho}_2^V(z_1, z_2)\, dz_1 dz_2,\\ &= \int_0^\infty \hat{\rho}(z_1)\, dz_1 = \int_0^\infty \hat{\rho}(z)\, dz.\end{aligned} \tag{3.2.4}$$

Finally, returning to our scaling and inserting the form of $\mathcal{N}(t)$ from (3.2.2) we have,

$$n_2^V(l_1, l_2, t) = \frac{\mathcal{L}}{2\langle l\rangle^3(t)}\hat{\rho}_2^V(z_1, z_2), \quad z_i := \frac{l_i}{\langle l\rangle(t)}. \tag{3.2.5}$$

We must also scale the functions n_3^u and n_3^d appropriately. In order to do this we note that, from an accountancy perspective, the number of valleys in the system is equal to the number of triples of either type. To see this consider running through the system from left to right counting the down lengths one by one, it is clear this gives $\mathcal{N}(t)$ since there is one down per valley, but then noting that the addition of a further length to the right of a valley doesn't change the number of objects being counted we see the result. Applying an only slightly modified procedure to the one above to the n_3 gives,

$$n_3^u(l_1,l_2,l_3,t) = \frac{\mathcal{L}}{2\langle l\rangle^4(t)}\hat{\rho}_3^u(z_1,z_2,z_3), \quad \text{and} \quad n_3^d(l_1,l_2,l_3,t) = \frac{\mathcal{L}}{2\langle l\rangle^4(t)}\hat{\rho}_3^d(z_1,z_2,z_3). \tag{3.2.6}$$

3.2.2 Scaled Fokker-Planck equation

We now consider these scaled quantities in the Fokker-Planck equation (3.2.3). We will insert the scaling form one term at a time for clarity starting with the left hand side of the equation and substitute using the scaling form (3.2.5),

$$\begin{aligned}
\frac{\partial}{\partial t}n_2^V(l_1,l_2,t) &= \frac{\partial}{\partial t}\left(\frac{\mathcal{L}}{2\langle l\rangle^3(t)}\hat{\rho}_2^V(z_1,z_2)\right), \\
&= \frac{\partial}{\partial t}\left(\frac{\mathcal{L}}{2\langle l\rangle^3(t)}\right)\hat{\rho}_2^V(z_1,z_2) + \frac{\mathcal{L}}{2\langle l\rangle^3(t)}\frac{\partial}{\partial t}\hat{\rho}_2^V(z_1,z_2), \\
&= -\frac{3\mathcal{L}\langle l\rangle'(t)}{2\langle l\rangle^4(t)}\hat{\rho}_2^V(z_1,z_2) - \frac{\mathcal{L}}{2\langle l\rangle^3(t)}\left[\frac{l_1\langle l\rangle'(t)}{\langle l\rangle^2(t)}\frac{\partial}{\partial z_1}\hat{\rho}_2^V(z_1,z_2)\right. \\
&\qquad\qquad\left.+\frac{l_2\langle l\rangle'(t)}{\langle l\rangle^2(t)}\frac{\partial}{\partial z_2}\hat{\rho}_2^V(z_1,z_2)\right], \\
&= -\frac{\mathcal{L}\langle l\rangle'(t)}{2\langle l\rangle^4(t)}\left[3\hat{\rho}_2^V(z_1,z_2) + z_1\frac{\partial}{\partial z_1}\hat{\rho}_2^V(z_1,z_2) + z_2\frac{\partial}{\partial z_2}\hat{\rho}_2^V(z_1,z_2)\right].
\end{aligned}$$

If we now consider the right hand side of equation (3.2.3) and this time substitute using (3.2.6) we have,

$$\begin{aligned}
&-\frac{\partial}{\partial l_1}\left[\int_0^\infty\left(\frac{1}{l'}-\frac{1}{l_2}\right)n_3^u(l',l_1,l_2,t)\,dl'\right] - \frac{\partial}{\partial l_2}\left[\int_0^\infty\left(\frac{1}{l''}-\frac{1}{l_1}\right)n_3^d(l_1,l_2,l'',t)\,dl''\right] \\
&= -\frac{1}{\langle l\rangle(t)}\frac{\partial}{\partial z_1}\left(\int_0^\infty\frac{1}{\langle l\rangle(t)}\left(\frac{1}{z'}-\frac{1}{z_2}\right)\frac{\mathcal{L}}{2\langle l\rangle^4(t)}\hat{\rho}_3^u(z',z_1,z_2)\ \langle l\rangle(t)dz'\right) \\
&\qquad -\frac{1}{\langle l\rangle(t)}\frac{\partial}{\partial z_2}\left(\int_0^\infty\frac{1}{\langle l\rangle(t)}\left(\frac{1}{z''}-\frac{1}{z_1}\right)\frac{\mathcal{L}}{2\langle l\rangle^4(t)}\hat{\rho}_3^d(z_1,z_2,z'')\ \langle l\rangle(t)dz''\right), \\
&= -\frac{\mathcal{L}}{2\langle l\rangle^5(t)}\left[\frac{\partial}{\partial z_1}\left(\int_0^\infty\left(\frac{1}{z'}-\frac{1}{z_2}\right)\hat{\rho}_3^u(z',z_1,z_2)\,dz'\right)\right. \\
&\qquad\left.+\frac{\partial}{\partial z_2}\left(\int_0^\infty\left(\frac{1}{z''}-\frac{1}{z_1}\right)\hat{\rho}_3^d(z_1,z_2,z'')\,dz''\right)\right].
\end{aligned}$$

Equating left and right hand sides we have,

$$
\begin{aligned}
&-\frac{\mathcal{L}\,\langle l\rangle'(t)}{2\,\langle l\rangle^4(t)}\left[3\hat{\rho}_2^V(z_1,z_2)+z_1\frac{\partial}{\partial z_1}\hat{\rho}_2^V(z_1,z_2)+z_2\frac{\partial}{\partial z_2}\hat{\rho}_2^V(z_1,z_2)\right]\\
&\quad=-\frac{\mathcal{L}}{2\,\langle l\rangle^5(t)}\left[\frac{\partial}{\partial z_1}\left(\int_0^\infty\left(\frac{1}{z'}-\frac{1}{z_2}\right)\hat{\rho}_3^u(z',z_1,z_2)\,dz'\right)\right.\\
&\qquad\qquad\left.+\frac{\partial}{\partial z_2}\left(\int_0^\infty\left(\frac{1}{z''}-\frac{1}{z_1}\right)\hat{\rho}_3^d(z_1,z_2,z'')\,dz''\right)\right], \qquad (3.2.7)
\end{aligned}
$$

and a simple rearrangement of which gives,

$$
\begin{aligned}
&\langle l\rangle'(t)\langle l\rangle(t)\left[3\hat{\rho}_2^V(z_1,z_2)+z_1\frac{\partial}{\partial z_1}\hat{\rho}_2^V(z_1,z_2)+z_2\frac{\partial}{\partial z_2}\hat{\rho}_2^V(z_1,z_2)\right]\\
&=\frac{\partial}{\partial z_1}\left(\int_0^\infty\left(\frac{1}{z'}-\frac{1}{z_2}\right)\hat{\rho}_3^u(z',z_1,z_2)\,dz'\right)+\frac{\partial}{\partial z_2}\left(\int_0^\infty\left(\frac{1}{z''}-\frac{1}{z_1}\right)\hat{\rho}_3^d(z_1,z_2,z'')\,dz''\right).
\end{aligned}
\tag{3.2.8}
$$

From equation (3.2.8) we can clearly see the only explicit dependence on time is in the form of the term $\langle l\rangle'(t)\langle l\rangle(t)$. Since we are assuming that a self similar solution exists we must have no dependence on time in the equation, and therefore by necessity we must have,

$$\langle l\rangle'(t)\langle l\rangle(t)=\lambda,$$

where λ is a constant. We can easily derive, based on this assumption, the form of $\langle l\rangle(t)$,

$$\langle l\rangle(t)=\mathcal{A}t^{\frac{1}{2}},$$

where $\mathcal{A}$ is a constant and we note that this is consistent with the scaling argument shown in (2.7.2) but is derived from a different perspective, that of necessity. From here onwards we simply set $\langle l\rangle'(t)\langle l\rangle(t)=\lambda$ and substitute into (3.2.8) to find the following Fokker-Planck type equation,

$$
\begin{aligned}
&\lambda\left[3\hat{\rho}_2^V(z_1,z_2)+z_1\frac{\partial}{\partial z_1}\hat{\rho}_2^V(z_1,z_2)+z_2\frac{\partial}{\partial z_2}\hat{\rho}_2^V(z_1,z_2)\right]\\
&=\frac{\partial}{\partial z_1}\left(\int_0^\infty\left(\frac{1}{z'}-\frac{1}{z_2}\right)\hat{\rho}_3^u(z',z_1,z_2)\,dz'\right)+\frac{\partial}{\partial z_2}\left(\int_0^\infty\left(\frac{1}{z''}-\frac{1}{z_1}\right)\hat{\rho}_3^d(z_1,z_2,z'')\,dz''\right).
\end{aligned}
\tag{3.2.9}
$$

3.2.3 Mean-Field and Closure Condition

Equation (3.2.9) is seen to depend on the higher order statistics $\hat{\rho}_3$ and is therefore not a closed system for the desired object $\hat{\rho}_2^V$. In fact, if we were to perform a similar procedure on the three-point distributions we would find dependence on the four-point distribution and so on. This is a common occurrence in these types of derivation, seeking an equation for one statistic requires information about higher order statistics and thus forms an infinite hierarchy. To circumvent this problem we 'close' the system by making what could be considered a typical mean-field assumption. In this circumstance we choose to assume that each of the valleys is independent of one another and thus the additional length, z' or z'', in the two three-point distributions present in (3.2.9) are assumed to be independent of the valley lengths z_1, z_2. This results in the following factorisation of the $\hat{\rho}_3$,

$$\hat{\rho}_3^u(z', z_1, z_2) = \hat{\rho}(z')\hat{\rho}_2^V(z_1, z_2) \quad \text{and} \quad \hat{\rho}_3^d(z_1, z_2, z'') = \hat{\rho}_2^V(z_1, z_2)\hat{\rho}(z''). \tag{3.2.10}$$

Note that the various $\hat{\rho}$ distributions still satisfy the symmetry properties explained earlier since the scalings performed are identical for all lengths and these do nothing to break this symmetry. As a result we can immediately see the following holds,

$$\hat{\rho}_2^V(z_1, z_2) = \hat{\rho}_2^V(z_2, z_1).$$

Similarly we note that the three-point distributions still satisfy the same reversal of arguments symmetry as before,

$$\hat{\rho}_3^u(z', z_1, z_2) = \hat{\rho}_3^d(z_2, z_1, z'),$$

and that our chosen mean field assumption doesn't invalidate this symmetry condition,

$$\hat{\rho}_3^u(z', z_1, z_2) = \hat{\rho}(z')\hat{\rho}_2^V(z_1, z_2) = \hat{\rho}_2^V(z_2, z_1)\hat{\rho}(z') = \hat{\rho}_3^d(z_2, z_1, z').$$

We now continue by taking our mean field closure condition (3.2.10) and inserting it into (3.2.9) and aim to simplify the equation as much as possible,

$$\begin{aligned}
&\lambda \left[3\hat{\rho}_2^V(z_1, z_2) + z_1\frac{\partial}{\partial z_1}\hat{\rho}_2^V(z_1, z_2) + z_2\frac{\partial}{\partial z_2}\hat{\rho}_2^V(z_1, z_2)\right] \\
&= \frac{\partial}{\partial z_1}\left(\int_0^\infty \left(\frac{1}{z'} - \frac{1}{z_2}\right)\hat{\rho}(z')\hat{\rho}_2^V(z_1, z_2)\, dz'\right) + \frac{\partial}{\partial z_2}\left(\int_0^\infty \left(\frac{1}{z''} - \frac{1}{z_1}\right)\hat{\rho}_2^V(z_1, z_2)\hat{\rho}(z'')\, dz''\right), \\
&= \frac{\partial}{\partial z_1}\left(\hat{\rho}_2^V(z_1, z_2)\int_0^\infty \frac{\hat{\rho}(z')}{z'}\, dz'\right) - \frac{1}{z_2}\frac{\partial}{\partial z_1}\left(\hat{\rho}_2^V(z_1, z_2)\int_0^\infty \hat{\rho}(z')\, dz'\right) \\
&\quad + \frac{\partial}{\partial z_2}\left(\hat{\rho}_2^V(z_1, z_2)\int_0^\infty \frac{\hat{\rho}(z'')}{z''}\, dz''\right) - \frac{1}{z_1}\frac{\partial}{\partial z_2}\left(\hat{\rho}_2^V(z_1, z_2)\int_0^\infty \hat{\rho}(z'')\, dz''\right).
\end{aligned} \tag{3.2.11}$$

In light of (3.2.4) if we consider the remaining integrals in (3.2.11) we see that there remains only one unknown, namely,

$$\int_0^\infty \frac{\hat{\rho}(z)}{z}\, dz.$$

If we are to have any hope of finding a solution to the equation we must have that this integral is finite, and as such we will therefore use the following definition,

$$\hat{\alpha} := \int_0^\infty \frac{\hat{\rho}(z)}{z}\,dz < \infty. \tag{3.2.12}$$

Making use of (3.2.12) and (3.2.4) in (3.2.11) we now have the following equation,

$$\begin{aligned}\lambda &\left[3\hat{\rho}_2^V(z_1,z_2) + z_1\frac{\partial}{\partial z_1}\hat{\rho}_2^V(z_1,z_2) + z_2\frac{\partial}{\partial z_2}\hat{\rho}_2^V(z_1,z_2)\right]\\ &= \frac{\partial}{\partial z_1}\left(\hat{\alpha}\hat{\rho}_2^V(z_1,z_2)\right) - \frac{1}{z_2}\frac{\partial}{\partial z_1}\left(\hat{\rho}_2^V(z_1,z_2)\right) + \frac{\partial}{\partial z_2}\left(\hat{\alpha}\hat{\rho}_2^V(z_1,z_2)\right) - \frac{1}{z_1}\frac{\partial}{\partial z_2}\left(\hat{\rho}_2^V(z_1,z_2)\right).\end{aligned} \tag{3.2.13}$$

Collecting terms from both sides and simplifying this can then be written as our final mean-field Fokker-Planck type equation for the scaling state,

$$3\lambda\hat{\rho}_2^V(z_1,z_2) + \left(\lambda z_1 + \frac{1}{z_2} - \hat{\alpha}\right)\frac{\partial}{\partial z_1}\hat{\rho}_2^V(z_1,z_2) + \left(\lambda z_2 + \frac{1}{z_1} - \hat{\alpha}\right)\frac{\partial}{\partial z_2}\hat{\rho}_2^V(z_1,z_2) = 0. \tag{3.2.14}$$

3.3 Implications for the Hill Distribution

Whilst we will place most of our focus on the two-point valley distribution it is worth noting what we can say about the related two-point hill distribution, $\hat{\rho}_2^H(z_1,z_2)$. This distribution could be found by use of a separate Fokker-Planck equation taking account of all the events that a hill experiences but since we have chosen to focus on the valley distribution we won't pursue this avenue any further. We know from our numerical simulations that the hill and valley distributions are quite different in character so we don't expect this equation to be identical to the valley equation. We note here, however, one interesting consequence of our previous mean-field assumption that valleys are independent. Taking the first part of (3.2.10) which states,

$$\hat{\rho}_3^u(z',z_1,z_2) = \hat{\rho}(z')\hat{\rho}_2^V(z_1,z_2), \tag{3.3.1}$$

we can then take the integral over z_2 on both sides of the equation which gives,

$$\begin{aligned}\int_0^\infty \hat{\rho}_3^u(z',z_1,z_2)\,dz_2 &= \int_0^\infty \hat{\rho}(z')\hat{\rho}_2^V(z_1,z_2)\,dz_2,\\ \hat{\rho}_2^H(z',z_1) &= \hat{\rho}(z')\hat{\rho}(z_1).\end{aligned} \tag{3.3.2}$$

As a result we see that our assumption that valleys are independent imposes the assumption that the lengths that make up a hill are independent and that the two-point distribution is simply the product of the one-point marginal distributions. We can see from our numerical simulations that this certainly isn't the case but nevertheless we continue to pursue our valley equation derived based on this independence assumption.

3.4 The Transport Equation for Valleys - A Different Approach

Here we present a different method of derivation of a Fokker-Planck type equation for valleys. The previous derivation built the governing equation as an hierarchy of equations which needed a mean-field type condition to close the equation. Ultimately though, the use of higher order statistics is only in order to understand what the neighbours to a valley are 'on average' and through this procedure we find an effective velocity for a valley based on this mean-field. Since we neglect any noise component in the full Fokker-Planck equation the end result is a transport equation with some velocity defined. Instead of building the equation from the Fokker-Planck equation, we assume we have an object which we know exists, the probability distribution of lengths, and write down the transport equation with an appropriately chosen velocity. This velocity will still contain a mean-field assumption since a generic valley needs external information to know about its individual velocity.

We believe this method is cleaner in the sense that all assumptions from the dynamical system side are stated in advance which sets the velocity. Once the velocity is established we simply insert this into the generic transport equation framework and make no reference to the dynamical system thereafter. In doing so we make clear the distinction between the *actual* distribution, which we call $\varrho(l_1, l_2, t)$, and our *predicted* distribution, $\rho(l_1, l_2, t)$, and simply aim to have a self contained PDE problem.

3.4.1 Dynamical System versus PDE

In the previous derivation the dynamical system and the PDE which we are trying to derive are influenced by each other at various stages throughout the derivation, for example influencing the mean-field assumptions and use of higher order statistics. We wish to untangle this and simply state the information and assumptions which we will use from the dynamical system up front and then, in the PDE regime, make no further mention of the dynamical system which we are trying to model. In this way we must observe the following pairs of information, shown in Figure 4.12, and ensure we maintain a distinction at all times.

By 'closure' in Figure 4.12 we mean that in the PDE regime we do not want any reference to dynamical system quantities, such as higher order statistics for triples or

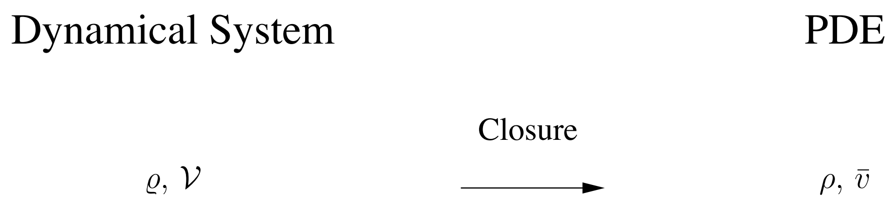


Figure 4.12 – Diagram of dynamical system quantities, ϱ and $\mathcal{V}$, and their PDE counterparts, ρ and $\bar{v}$. By closure we are indicating that to find a self consistent PDE model from the full dynamical system we may be forced to make some kind of mean-field type assumption.

quadruples, but instead want all the information in the PDE setting to be self-consistent. In doing so we aim to have a self contained PDE problem which contains no mention of and requires no information from the dynamical system which it is derived. As a valley in the dynamical system setting must have information about its neighbours in order for the dynamics to proceed, we must 'close' the system in an appropriate way so that a valley moves in some typical manner with no specific information about the neighbouring valleys. This will essentially be a type of *mean-field* closure in that we will assume the surroundings are mean in length thus reducing any requirement to know their exact lengths.

On the left hand side of Figure 4.12 we have a distribution of lengths, ϱ, which represents the dynamical system and can be obtained approximately via numerically simulations. The dynamical system evolves via the velocity law $\mathcal{V}$ which is simply the velocity law stated earlier in (2.6.3), namely,

$$\mathcal{V}(l_i) := \frac{dl_i}{dt} = (-1)^i \left(\frac{1}{l_{i+1}} - \frac{1}{l_{i-1}} \right).$$

On the right hand side of Figure 4.12 we have the predicted distribution, ρ, which we are trying to write a governing equation for, together with the velocity field, $\bar{v}$, that the distribution will be subject to. As noted this must be self-consistent and we must make suitable modifications to the velocity law (2.6.3) to remove reference to the right and left neighbours to the valley in question. We have a valley with lengths (l_1, l_2) and corresponding velocities from the dynamical system,

$$\mathcal{V}(l_1) = \left(\frac{1}{l_L} - \frac{1}{l_2} \right) \quad \text{and} \quad \mathcal{V}(l_2) = \left(\frac{1}{l_R} - \frac{1}{l_1} \right),$$

where l_L and l_R are the left and right neighbouring lengths to our valley respectively. In order to close the system we simply make the typical *mean-field* assumption that the neighbouring lengths are both 'mean' and so we must calculate the following expected value which we will define as $\alpha(t)$,

$$\alpha(t) := \left\langle \frac{1}{l} \right\rangle = \int_0^\infty \int_0^\infty \frac{\rho(l_l, l_r, t)}{l_r} \, dl_l dl_r = \int_0^\infty \frac{\rho_m(l_r, t)}{l_r} \, dl_r, \tag{3.4.1}$$

where ρ_m is the marginal distribution obtained by integration over one of the spatial coordinates. Hence we have our velocity vector $\bar{v}$, which is the self consistent velocity which the distribution ρ evolves under,

$$\bar{v} := \begin{pmatrix} \frac{dl_1}{dt} \\ \frac{dl_2}{dt} \end{pmatrix} = \begin{pmatrix} \left\langle \frac{1}{l} \right\rangle - \frac{1}{l_2} \\ \left\langle \frac{1}{l} \right\rangle - \frac{1}{l_1} \end{pmatrix}. \tag{3.4.2}$$

3.4.2 The Distribution, Derived Objects and Flow

We have already introduced the distribution $\rho(l_1, l_2, t)$, which is the distribution we actually seek as our estimate of the 'real' distribution, ϱ, on the dynamical system side. For now, however, we assume that it is a given object and can deduce other objects from this single starting point. First we take the distribution and derive the expected value of the total length of a valley as follows,

$$\langle L_{\mathrm{V}} \rangle(t) = \int_0^\infty \int_0^\infty (l_1 + l_2)\rho(l_1, l_2, t) \, dl_1 dl_2. \tag{3.4.3}$$

Note that by splitting this integral into two components we can easily show that the mean valley length is simply twice the mean individual length size,

$$\begin{aligned} \langle L_{\mathrm{V}} \rangle(t) &= \int_0^\infty \int_0^\infty (l_1 + l_2)\rho(l_1, l_2, t) \, dl_1 dl_2, \\ &= \int_0^\infty \int_0^\infty l_1 \rho(l_1, l_2, t) \, dl_1 dl_2 + \int_0^\infty \int_0^\infty l_2 \rho(l_1, l_2, t) \, dl_1 dl_2, \\ &= \int_0^\infty l_1 \rho_m(l_1, t) \, dl_1 + \int_0^\infty l_2 \rho_m(l_2, t) \, dl_2, \\ &=: \langle l \rangle(t) + \langle l \rangle(t), \\ &= 2\langle l \rangle(t), \end{aligned} \tag{3.4.4}$$

where $\langle l \rangle(t)$ then denotes the mean individual length. We now make use of this mean valley size, $\langle L_{\mathrm{V}} \rangle(t)$, and define the following object,

$$n(l_1, l_2, t) = \frac{\rho(l_1, l_2, t)}{\langle L_{\mathrm{V}} \rangle(t)}. \tag{3.4.5}$$

We can interpret n as the number density of valleys per unit length, that is, we know the probability distribution of valleys and so by dividing by the mean length of a valley we are effectively considering the number of valleys of a particular size which will exist, probabilistically speaking, on a unit interval. Another way of looking at this definition is to suppose that each unit interval is populated with, on average, $N_V(t)$ valleys. Since we are considering a unit interval it is clear that $N_V(t) = 1/\langle L_{\mathrm{V}}\rangle(t)$ and so to find the probability of finding a valley of a particular length in the unit interval we simply divide the number of valleys of that length by the total number of valleys in the interval to get a probability,

$$\rho(l_1, l_2, t) = \frac{n(l_1, l_2, t)}{N_V(t)} = \langle L_{\mathrm{V}}\rangle(t) n(l_1, l_2, t).$$

A simple rearrangement returns to our original definition (3.4.5). We now consider our proposed model in which this probability density function flows via the velocity field, defined in (3.4.2), and so proceed initially by considering the standard transport equation. Instead of considering the flow for the probability ρ directly, we instead make use of the number density definition and write down the transport equation for n,

$$\frac{\partial n}{\partial t} + \nabla \cdot (n\bar{v}) = 0, \tag{3.4.6}$$

where we have suppressed the arguments of n for simplicity. Note that since there are no terms on the right hand side we are in a situation where there is no creation of valleys, only transport via the velocity field, $\bar{v}$. By doing this we are able, in general, to make adjustments to the equation to account for any sinks and sources that may be present in the system. This is much more appropriate in the n regime since it amounts to a simple counting argument whereas in the ρ regime this is somewhat harder to picture. In the case we are considering here the only place where valleys can disappear is at the origin, we have previously shown that this is the case. This loss from the system can easily be justified for now by considering that n gives some idea of the numbers of valleys of a particular size. Since some valleys disappear from the system and there is no mechanism for creation we see that it is reasonable to expect that there is flow out of the area of interest but that it is already accounted for in the equation as stated. As a result of this and the lack of any other sinks or sources we can now treat (3.4.6), together with the velocity field (3.4.2), as our governing equation for flow.

3.4.3 Integration of Governing Equation

We now wish to take the integral of the whole governing equation, (3.4.6), and consider what this can tell us about the population density, n. First let us consider our definition of population density, (3.4.5), and integrate over all of positive space,

$$\begin{aligned}\int_{0^+}^{+\infty}\int_{0^+}^{+\infty} n(l_1,l_2,t)\ dl_1dl_2 &= \int_{0^+}^{+\infty}\int_{0^+}^{+\infty} \frac{\rho(l_1,l_2,t)}{\langle L_{\mathrm{V}}\rangle(t)}\ dl_1dl_2,\\ &= \frac{1}{\langle L_{\mathrm{V}}\rangle(t)}\int_{0^+}^{+\infty}\int_{0^+}^{+\infty} \rho(l_1,l_2,t)\ dl_1dl_2,\\ &= \frac{1}{\langle L_{\mathrm{V}}\rangle(t)},\end{aligned}$$

where we have written 0^+ to indicate that strictly speaking we are taking the integral over a finite area in the positive domain and then taking the limit as the lower bounds tend to zero from above and simultaneously as the upper bound goes to infinity.

If we consider the velocity field (3.4.2) we see that the axes present singularities in the velocity field and, for this reason, we must take great care here. This is the justification for taking a positive area and then considering the limit to zero. In Figure 4.13 we see the area, $\Omega_{\epsilon,\delta}$, which we are initially taking the integral over.

Performing integration over the area shown against the whole of the governing equation (3.4.6) and taking account of the divergence theorem we now have the following equation,

$$\frac{\partial}{\partial t}\int_{\Omega_{\epsilon,\delta}} n\ dA + \int_{\partial\Omega_{\epsilon,\delta}} n\bar{v}\cdot\bar{m}\ ds = 0,$$

where dA is an area element in the domain, $\partial\Omega_{\epsilon,\delta}$ is the boundary of the domain $\Omega_{\epsilon,\delta}$, $\bar{m}$ is an outward unit vector, normal to the boundary $\partial\Omega_{\epsilon,\delta}$ and ds is a line segment on the oriented boundary.

Let us consider the first term on the left hand side. Eventually we would like to take the limit as both ϵ and δ go to zero from above and hence we are taking the integral over the whole positive domain. Taking this limit immediately in this circumstance we see that we have already defined this quantity as equal to $1/\langle L_{\mathrm{V}}\rangle(t)$,

$$\lim_{\epsilon\to0^+,\,\delta\to0^+}\left(\frac{\partial}{\partial t}\int_{\Omega_{\epsilon,\delta}} n\ dA\right) = \frac{\partial}{\partial t}\int_{0^+}^{+\infty}\int_{0^+}^{+\infty} n\ dl_1dl_2 = \frac{\partial}{\partial t}\left(\frac{1}{\langle L_{\mathrm{V}}\rangle(t)}\right).$$

Now let us consider the remaining term on the left hand side, the boundary term. The boundary of the area $\Omega_{\epsilon,\delta}$ has three distinct regions, the two straight sections parallel to the axes and the quarter ball area around the origin. We will treat each separately, although we shall see that both of the straight segments essentially have the same structure. We have to compute the following,

$$\int_{\partial\Omega_{\epsilon,\delta}} n\bar{v}\cdot\bar{m}\ ds = \int_{\partial\Omega^1_{\epsilon,\delta}} n\bar{v}\cdot\bar{m}_1\ ds + \int_{\partial\Omega^2_{\epsilon,\delta}} n\bar{v}\cdot\bar{m}_2\ ds + \int_{\partial\Omega^c_{\epsilon,\delta}} n\bar{v}\cdot\bar{m}_c\ ds,$$

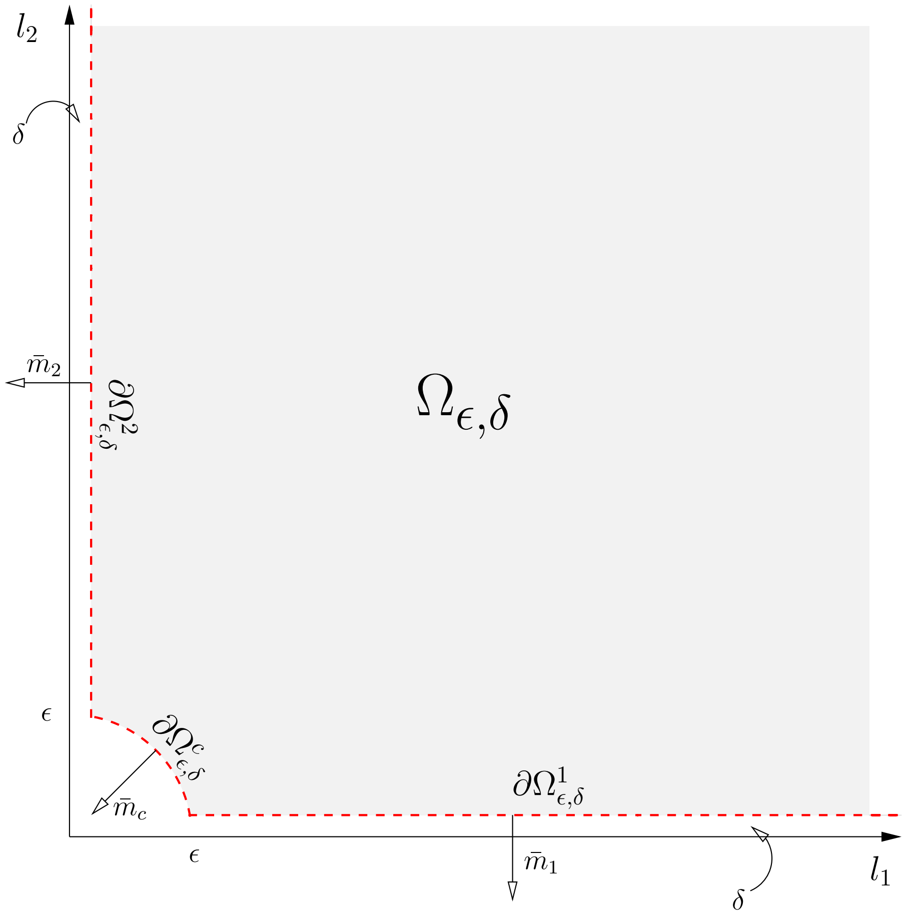

l_2
δ
$\bar{m}_2$
$\partial\Omega^2_{\epsilon,\delta}$
$\Omega_{\epsilon,\delta}$
ϵ
$\partial\Omega^c_{\epsilon,\delta}$
$\bar{m}_c$
$\partial\Omega^1_{\epsilon,\delta}$
ϵ
$\bar{m}_1$
δ
l_1

Figure 4.13 – Illustration of area over which we take the integral of the governing equation. Note that the area avoids the l_1 and l_2 axes where we have singularities in the velocity field and similarly the origin.

where we have indexed the three separate regions. Consider region 1, the straight line segment parallel to the l_1 axis. Here the normal vector to the boundary is the vector $\bar{m}_1 = (0, -1)$ and by taking the dot product with the velocity vector $\bar{v}$ we have the following,

$$\int_{\partial\Omega^1_{\epsilon,\delta}} n\bar{v}\cdot\bar{m}_1 \, ds = \int_{\epsilon^\sim}^{+\infty} n(l_1,\delta,t)\left(\frac{1}{l_1} - \alpha(t)\right)\, dl_1,$$

where we have written $\epsilon^\sim$ to indicate that the intersection of the arc, labelled section c, and the line in section 1 would not occur at precisely ϵ but it would be very close to that value. Since we anticipate taking limits later it is sufficient to just consider it small and close to ϵ in size, in particular, once the δ-limit is taken this will go to ϵ as we expect.

Proceeding to take the limit as δ goes to 0 from above we have,

$$\lim_{\delta\to 0^+}\left(\int_{\epsilon^{\sim}}^{+\infty} n(l_1,\delta,t)\left(\frac{1}{l_1}-\alpha(t)\right)\,dl_1\right)=\int_{\epsilon}^{+\infty} n(l_1,0^+,t)\left(\frac{1}{l_1}-\alpha(t)\right)\,dl_1.$$

If we now consider region 2, the straight line segment parallel to the l_2 axis, together with the unit vector $\bar{m}_2=(-1,0)$ we can then follow a very similar derivation to the one above to find,

$$\begin{aligned}\lim_{\delta\to 0^+}\left(\int_{\partial\Omega^2_{\epsilon,\delta}} n\bar{v}\cdot\bar{m}_2\,ds\right)&=-\lim_{\delta\to 0^+}\left(\int_{+\infty}^{\epsilon^{\sim}} n(\delta,l_2,t)\left(\frac{1}{l_2}-\alpha(t)\right)\,dl_2\right),\\&=\int_{\epsilon}^{+\infty} n(0^+,l_2,t)\left(\frac{1}{l_2}-\alpha(t)\right)\,dl_2,\end{aligned}$$

where the main alteration comes from considering that orienting the line means we introduce a further minus from the line segment dl_2. For the remaining region, labelled c, we consider a polar type representation of the arc. In this representation we have simply that the unit vector is $\bar{m}_c=(-\cos\theta,-\sin\theta)$ and we replace l_1 and l_2 in the velocity law by the polar values $r\cos\theta$ and $r\sin\theta$ respectively. We also make the transformation that $s=\pi/2-\theta$ and so ds is transformed to $-d\theta$. Hence we have,

$$\begin{aligned}&\int_{\frac{\pi}{2}^{\sim}}^{0^{\sim}} n(\epsilon\cos\theta,\epsilon\sin\theta,t)\left(\alpha(t)-\frac{1}{\epsilon\sin\theta},\alpha(t)-\frac{1}{\epsilon\cos\theta}\right)\cdot(-\cos\theta,-\sin\theta)\;(-\epsilon d\theta),\\=&\int_{0^{\sim}}^{\frac{\pi}{2}^{\sim}} n(\epsilon\cos\theta,\epsilon\sin\theta,t)\left(-\alpha(t)\epsilon(\cos\theta+\sin\theta)+\frac{\cos\theta}{\sin\theta}+\frac{\sin\theta}{\cos\theta}\right)\,d\theta,\\=&-\int_{0^{\sim}}^{\frac{\pi}{2}^{\sim}} n(\epsilon\cos\theta,\epsilon\sin\theta,t)\left(\alpha(t)\epsilon(\cos\theta+\sin\theta)-\frac{2}{\sin 2\theta}\right)\,d\theta.\end{aligned}$$

We now consider taking the limit as $\epsilon\to 0^+$ and note the following; if the product of the first term inside the bracket and the n term tends to a constant value $C\neq 0$, say, then this would imply that n must behave like $1/\epsilon$ as it approaches the origin. This, however, would imply that the product of n and the second term in the brackets would blow up in the limit. Thus to ensure we have a finite limit of the complete term we must have that the first term product vanishes and we are left with,

$$\begin{aligned}&\lim_{\epsilon\to 0^+}\left(-\int_{0^{\sim}}^{\frac{\pi}{2}^{\sim}} n(\epsilon\cos\theta,\epsilon\sin\theta,t)\left(-\frac{2}{\sin 2\theta}\right)\,d\theta\right),\\=&\lim_{\epsilon\to 0^+}\left(2\int_{0^{\sim}}^{\frac{\pi}{2}^{\sim}}\frac{n(\epsilon\cos\theta,\epsilon\sin\theta,t)}{\sin 2\theta}\,d\theta\right),\\=&2\int_{0}^{\frac{\pi}{2}}\frac{n^{\star}(\theta,t)}{\sin 2\theta}\,d\theta,\end{aligned}$$

where we have used the following definition,

$$n^{\star}(\theta,t) = \lim_{\epsilon\to 0^+} n(\epsilon\cos\theta, \epsilon\sin\theta, t).$$

Taking the limit $\epsilon \to 0^+$ in the regions 1 and 2 and combining all these results together we have the following,

$$\frac{\partial}{\partial t}\left(\frac{1}{\langle L_{\mathrm{V}}\rangle(t)}\right) = -\left[\int_{0^+}^{+\infty} n(l_1,0^+,t)\left(\frac{1}{l_1}-\alpha(t)\right)\,dl_1\;,\right.$$
$$\left. + \int_{0^+}^{+\infty} n(0^+,l_2,t)\left(\frac{1}{l_2}-\alpha(t)\right)\,dl_2 + 2\int_0^{\frac{\pi}{2}}\frac{n^{\star}(\theta,t)}{\sin 2\theta}\,d\theta\right].$$

Labelling the right hand side as $-\mathcal{R}(t)$ we have the following relationship,

$$\frac{\partial}{\partial t}\left(\frac{1}{\langle L_{\mathrm{V}}\rangle(t)}\right) = -\mathcal{R}(t),$$

$$\mathcal{R}(t) = \int_{0^+}^{+\infty} n(l_1,0^+,t)\left(\frac{1}{l_1}-\alpha(t)\right)\,dl_1 + \int_{0^+}^{+\infty} n(0^+,l_2,t)\left(\frac{1}{l_2}-\alpha(t)\right)\,dl_2$$
$$+ 2\int_0^{\frac{\pi}{2}}\frac{n^{\star}(\theta,t)}{\sin 2\theta}\,d\theta.$$

In the next section we want to consider the original distribution ρ, rather than the number density n, so we will substitute into the expression above for n to get,

$$\int_{0^+}^{+\infty} \frac{\rho(l_1,0^+,t)}{\langle L_{\mathrm{V}}\rangle(t)}\left(\frac{1}{l_1}-\alpha(t)\right)\,dl_1 + \int_{0^+}^{+\infty} \frac{\rho(0^+,l_2,t)}{\langle L_{\mathrm{V}}\rangle(t)}\left(\frac{1}{l_2}-\alpha(t)\right)\,dl_2$$
$$+ 2\int_0^{\frac{\pi}{2}}\frac{\rho^{\star}(\theta,t)}{\langle L_{\mathrm{V}}\rangle(t)\sin 2\theta}\,d\theta,$$
$$=\frac{1}{\langle L_{\mathrm{V}}\rangle(t)}\left[\int_{0^+}^{+\infty} \rho(l_1,0^+,t)\left(\frac{1}{l_1}-\alpha(t)\right)\,dl_1 + \int_{0^+}^{+\infty} \rho(0^+,l_2,t)\left(\frac{1}{l_2}-\alpha(t)\right)\,dl_2\right.$$
$$\left. +2\int_0^{\frac{\pi}{2}}\frac{\rho^{\star}(\theta,t)}{\sin 2\theta}\,d\theta\right],$$
$$=\frac{1}{\langle L_{\mathrm{V}}\rangle(t)}\mathcal{R}_{\rho}(t) = \mathcal{R}(t), \tag{3.4.7}$$

where,

$$\rho^{\star}(\theta,t) = \lim_{\epsilon\to 0^+} \rho(\epsilon\cos\theta, \epsilon\sin\theta, t).$$

3.4.4 Return to the Governing Equation

If we now return to the governing equation (3.4.6) and substitute in for n using (3.4.5) we have,

$$\frac{\partial}{\partial t}\left(\frac{\rho}{\langle L_{\mathrm{V}}\rangle(t)}\right) + \frac{1}{\langle L_{\mathrm{V}}\rangle(t)}\nabla\cdot(\rho\bar{v}) = 0,$$
$$\frac{\partial}{\partial t}\left(\frac{1}{\langle L_{\mathrm{V}}\rangle(t)}\right)\rho + \frac{1}{\langle L_{\mathrm{V}}\rangle(t)}\frac{\partial\rho}{\partial t} + \frac{1}{\langle L_{\mathrm{V}}\rangle(t)}\nabla\cdot(\rho\bar{v}) = 0.$$

Making note of (3.4.7) at this stage and multiplying through by $\langle L_{\rm V}\rangle(t)$ we have,

$$\frac{\partial\rho}{\partial t}+\nabla\cdot(\rho\bar{v})=\mathcal{R}_\rho(t)\rho. \tag{3.4.8}$$

This is our governing equation for the probability distribution, ρ. Almost identical in form to the original equation for n we see the addition of a source term on the right hand side. This can easily be understood by remembering that valleys flow out of the system but in order to maintain the properties of a probability distribution (integral 1, for example) we must not lose any 'mass' from the probability distribution, thus the source term.

3.4.5 Scaling Solution

We now proceed to consider if scaling solutions to (3.4.8) exist and propose the following scaling of the spatial coordinates,

$$l_i(t)\to\frac{2l_i(t)}{\langle L_{\rm V}\rangle(t)}=:z_i. \tag{3.4.9}$$

Accordingly we therefore have the following scaling on the distribution ρ in order to maintain its probability structure,

$$\rho(l_1,l_2,t):=\frac{4}{\langle L_{\rm V}\rangle^2(t)}\hat{\rho}(z_1,z_2). \tag{3.4.10}$$

We note the following simplification in order to reduce the occurrence of the factor 2. We found earlier that $\langle l\rangle(t)=\langle L_{\rm V}\rangle(t)/2$, (3.4.4), and so we simply have the following scalings,

$$z_i=\frac{l_i(t)}{\langle l\rangle(t)}\qquad\text{and}\qquad\rho(l_1,l_2,t):=\frac{\hat{\rho}(z_1,z_2)}{\langle l\rangle^2(t)}.$$

We will now consider equation (3.4.8) term by term and substitute in for the proposed scaling form above. We begin with the first term on the left hand side,

$$\begin{aligned}\frac{\partial\rho(l_1,l_2,t)}{\partial t}&=\frac{\partial}{\partial t}\left(\frac{1}{\langle l\rangle^2(t)}\right)\hat{\rho}(z_1,z_2)+\frac{1}{\langle l\rangle(t)}\frac{\partial\hat{\rho}(z_1,z_2)}{\partial t},\\&=-2\frac{\langle l\rangle'(t)}{\langle l\rangle^3(t)}\hat{\rho}(z_1,z_2)+\frac{1}{\langle l\rangle(t)}\left[\frac{\partial\hat{\rho}(z_1,z_2)}{\partial z_1}l_1\frac{\partial}{\partial t}\left(\frac{1}{\langle l\rangle(t)}\right)\right.\\&\qquad\left.+\frac{\partial\hat{\rho}(z_1,z_2)}{\partial z_2}l_2\frac{\partial}{\partial t}\left(\frac{1}{\langle l\rangle(t)}\right)\right],\\&=-2\frac{\langle l\rangle'(t)}{\langle l\rangle^3(t)}\hat{\rho}(z_1,z_2)-\frac{\langle l\rangle'(t)}{\langle l\rangle^3(t)}\left[\frac{\partial\hat{\rho}(z_1,z_2)}{\partial z_1}\frac{l_1}{\langle l\rangle(t)}+\frac{\partial\hat{\rho}(z_1,z_2)}{\partial z_2}\frac{l_2}{\langle l\rangle(t)}\right],\\&=-2\frac{\langle l\rangle'(t)}{\langle l\rangle^3(t)}\hat{\rho}(z_1,z_2)-\frac{\langle l\rangle'(t)}{\langle l\rangle^3(t)}\left[z_1\frac{\partial\hat{\rho}(z_1,z_2)}{\partial z_1}+z_2\frac{\partial\hat{\rho}(z_1,z_2)}{\partial z_2}\right].\end{aligned} \tag{3.4.11}$$

Now let us consider the second term on the left hand side,

$$\begin{aligned}\nabla\cdot(\rho\bar{v}) &= \frac{\partial}{\partial l_1}\left[\left(\alpha(t)-\frac{1}{l_2}\right)\rho(l_1,l_2,t)\right]+\frac{\partial}{\partial l_2}\left[\left(\alpha(t)-\frac{1}{l_1}\right)\rho(l_1,l_2,t)\right],\\ &= \frac{1}{\langle l\rangle(t)}\left(\frac{\partial}{\partial z_1}\left[\left(\frac{\hat{\alpha}}{\langle l\rangle(t)}-\frac{1}{z_2\langle l\rangle(t)}\right)\frac{\hat{\rho}(z_1,z_2)}{\langle l\rangle^2(t)}\right]\right.\\ &\qquad\left.+\frac{\partial}{\partial z_2}\left[\left(\frac{\hat{\alpha}}{\langle l\rangle(t)}-\frac{1}{z_1\langle l\rangle(t)}\right)\frac{\hat{\rho}(z_1,z_2)}{\langle l\rangle^2(t)}\right]\right),\\ &= \frac{1}{\langle l\rangle^4(t)}\left(\frac{\partial}{\partial z_1}\left[\left(\hat{\alpha}-\frac{1}{z_2}\right)\hat{\rho}(z_1,z_2)\right]+\frac{\partial}{\partial z_2}\left[\left(\hat{\alpha}-\frac{1}{z_1}\right)\hat{\rho}(z_1,z_2)\right]\right),\end{aligned}\tag{3.4.12}$$

where we have made use of the following,

$$\begin{aligned}\alpha(t) &= \int_0^\infty\int_0^\infty \frac{\rho(l_1,l_2,t)}{l_2}\,dl_1dl_2,\\ &= \frac{1}{\langle l\rangle^2(t)}\int_0^\infty\int_0^\infty \frac{\hat{\rho}(z_1,z_2)}{z_2\langle l\rangle(t)}\,\langle l\rangle^2(t)dz_1dz_2,\\ &= \frac{1}{\langle l\rangle(t)}\int_0^\infty\int_0^\infty \frac{\hat{\rho}(z_1,z_2)}{z_2}\,dz_1dz_2 = \frac{1}{\langle l\rangle(t)}\int_0^\infty \frac{\hat{\rho}(z_2)}{z_2}\,dz_2 =: \frac{\hat{\alpha}}{\langle l\rangle(t)}.\end{aligned}$$

The term on the right hand side of (3.4.8) is simple to treat requiring only a trivial substitution, namely,

$$\mathcal{R}_\rho(t)\rho(l_1,l_2,t)=\frac{\mathcal{R}_\rho(t)}{\langle l\rangle^2(t)}\hat{\rho}(z_1,z_2)=\frac{\mathcal{R}_\rho(t)}{\langle L_{\text{V}}\rangle(t)}\frac{2}{\langle l\rangle(t)}\hat{\rho}(z_1,z_2)=\frac{2\mathcal{R}(t)}{\langle l\rangle(t)}\hat{\rho}(z_1,z_2).$$

We now want to combine these three parts back together as one equation but before we do so we consider the following relationship between $\langle l\rangle(t)$ and $\mathcal{R}(t)$,

$$\mathcal{R}(t)=-\frac{\partial}{\partial t}\left(\frac{1}{\langle L_{\text{V}}\rangle(t)}\right)=-\frac{\partial}{\partial t}\left(\frac{1}{2\langle l\rangle(t)}\right)=\frac{1}{2}\frac{\langle l\rangle'(t)}{\langle l\rangle^2(t)}.$$

Making note of this relationship we now have,

$$\begin{aligned}-4\frac{\mathcal{R}(t)}{\langle l\rangle(t)}\hat{\rho}(z_1,z_2)-2\frac{\mathcal{R}(t)}{\langle l\rangle(t)}\left[z_1\frac{\partial\hat{\rho}(z_1,z_2)}{\partial z_1}+z_2\frac{\partial\hat{\rho}(z_1,z_2)}{\partial z_2}\right]&\\ +\frac{1}{\langle l\rangle^4(t)}\left(\frac{\partial}{\partial z_1}\left[\left(\hat{\alpha}-\frac{1}{z_2}\right)\hat{\rho}(z_1,z_2)\right]+\frac{\partial}{\partial z_2}\left[\left(\hat{\alpha}-\frac{1}{z_1}\right)\hat{\rho}(z_1,z_2)\right]\right)&\\ =\frac{2\mathcal{R}(t)}{\langle l\rangle(t)}\hat{\rho}(z_1,z_2).&\end{aligned}\tag{3.4.13}$$

Multiplying through by $\langle l\rangle^4(t)$ and then rearranging we have,

$$\begin{aligned}6\mathcal{R}(t)\langle l\rangle^3(t)\hat{\rho}(z_1,z_2)+\left[\left(2\mathcal{R}(t)\langle l\rangle^3(t)z_1+\frac{1}{z_2}-\hat{\alpha}\right)\frac{\partial\hat{\rho}(z_1,z_2)}{\partial z_1}\right]&\\ +\left[\left(2\mathcal{R}(t)\langle l\rangle^3(t)z_2+\frac{1}{z_1}-\hat{\alpha}\right)\frac{\partial\hat{\rho}(z_1,z_2)}{\partial z_2}\right]=0.&\end{aligned}\tag{3.4.14}$$

Now if we let $\lambda := \langle l\rangle'(t)\,\langle l\rangle(t) = 2\mathcal{R}(t)\,\langle l\rangle^3(t)$ be a constant, an assertion that is necessary to remove the time dependence in the equation and is consistent with the scaling behaviour predicted earlier in (2.7.2), we have,

$$3\lambda\hat{\rho}(z_1,z_2) + \left(\lambda z_1 + \frac{1}{z_2} - \hat{\alpha}\right)\frac{\partial}{\partial z_1}\hat{\rho}(z_1,z_2) + \left(\lambda z_2 + \frac{1}{z_1} - \hat{\alpha}\right)\frac{\partial}{\partial z_2}\hat{\rho}(z_1,z_2) = 0. \quad (3.4.15)$$

Using the same scaling procedure we also have the following conditions on the distribution,

$$\hat{\alpha} = \int_0^\infty\int_0^\infty \frac{\hat{\rho}(z_1,z_2)}{z_2}\,dz_1dz_2, \quad 1 = \int_0^\infty\int_0^\infty \hat{\rho}(z_1,z_2)\,dz_1dz_2$$
$$\text{and} \quad 1 = \int_0^\infty\int_0^\infty z_1\hat{\rho}(z_1,z_2)\,dz_1dz_2.$$

We see that equation (3.4.15) matches (3.2.14) from earlier but as was stated we feel that the second method is more transparent in its derivation. This equation is of hyperbolic type and so we proceed to consider the characteristic curves of the equation.

3.4.6 Characteristics of Scaled Hyperbolic Equation

The characteristics of the equation are found by plotting the solution curves to the following equations,

$$\begin{aligned}\frac{dz_1(\xi)}{d\xi} &= \lambda z_1(\xi) + \frac{1}{z_2(\xi)} - \hat{\alpha},\\ \frac{dz_2(\xi)}{d\xi} &= \lambda z_2(\xi) + \frac{1}{z_1(\xi)} - \hat{\alpha},\end{aligned}$$

where ξ is just a dummy variable which represents time elapsed along the curve. For illustration purposes we set $\lambda = \hat{\alpha}^2/4$, a condition which will be derived in the one-point model which follows but is used here for illustration, and therefore numerically solve the equations,

$$\frac{dz_1(\xi)}{d\xi} = \frac{\hat{\alpha}^2}{4}z_1(\xi) + \frac{1}{z_2(\xi)} - \hat{\alpha}, \quad (3.4.16)$$
$$\frac{dz_2(\xi)}{d\xi} = \frac{\hat{\alpha}^2}{4}z_2(\xi) + \frac{1}{z_1(\xi)} - \hat{\alpha}. \quad (3.4.17)$$

Note that the characteristics now have a singularity at $z_i = 2/\hat{\alpha}$, $i \in \{1,2\}$ and any characteristic which directly approaches this point must terminate there. The numerical procedure is a simple RK4 integration step from a set of starting points in the z_1, z_2-plane which builds up the profile. Note that these equations are symmetric in z_1 and z_2 so we need only calculate the characteristics in one half of the plane separated by the symmetry

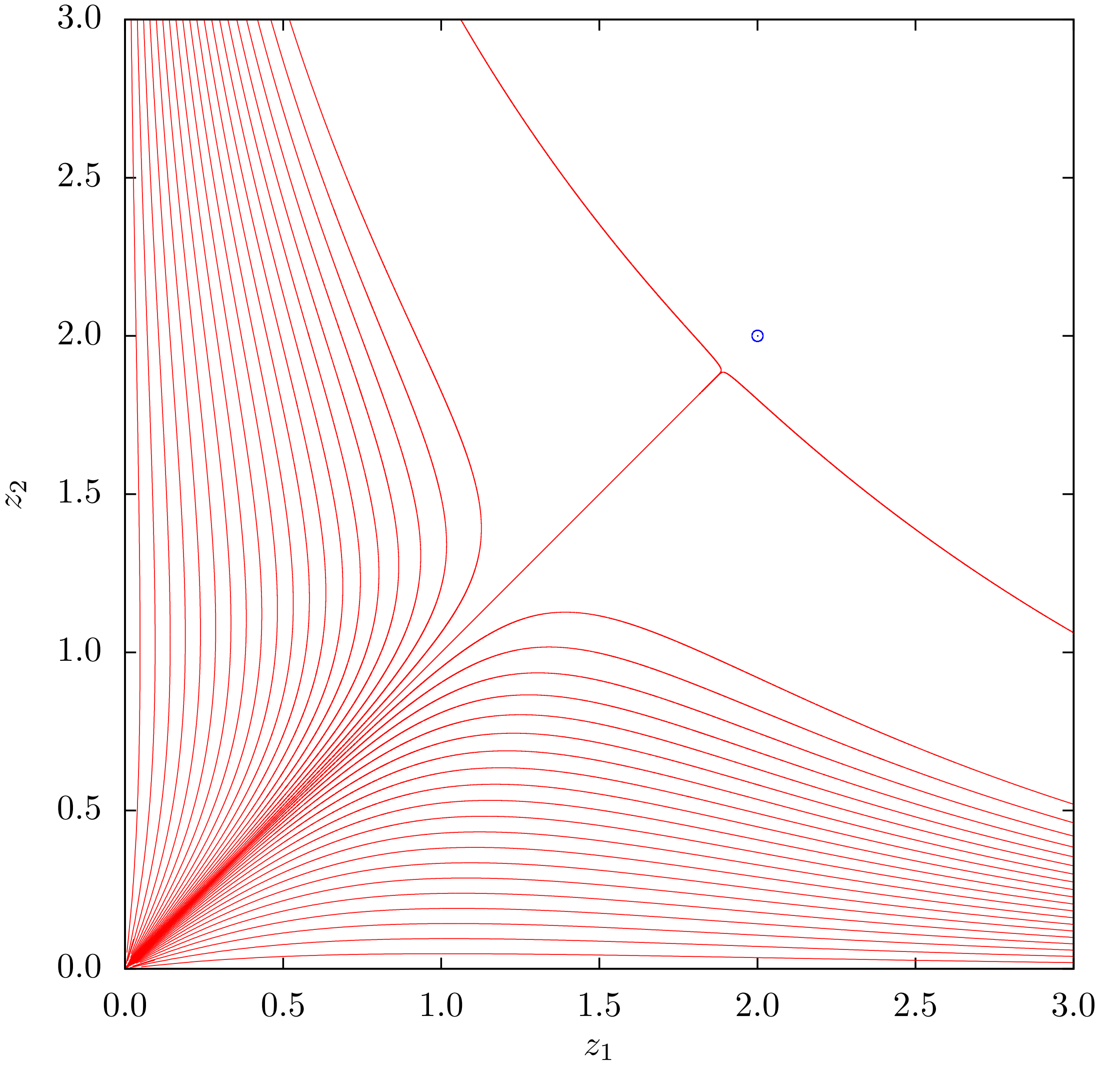


Figure 4.14 – Characteristic curves found by numerically solving equations (3.4.16) and (3.4.17) by setting $\lambda = \hat{\alpha}^2/4$, a condition derived during the one-point model reduction which follows, and then choosing $\hat{\alpha} = 1$. The blue point indicates the point $(2, 2)$, a saddle-node point that trajectories cannot pass for this choice of λ and $\hat{\alpha}$. The region to the top right of the diagram with no characteristic curves roughly illustrates the region which has $\hat{\rho} = 0$, and is the 2D counterpart to region $z \geqslant 1/\sqrt{\lambda}$ derived in the 1D theory that follows.

line $z_1 = z_2$. In Figure (4.14) we see a selection of the characteristic curves found by using the numerical procedure above.

Note that the curves all appear to come from the origin and that there is a characteristic curve along the symmetry line $z_1 = z_2$ which as predicted terminates at the point $(2/\hat{\alpha}, 2/\hat{\alpha})$. This is interesting because this tells us that the data which is on this line, a symmetric valley, will stay symmetric for all time. We know that this likely wouldn't be the case in the full dynamical system, nevertheless we exploit this feature in the next section where we attempt to find a one-point theory for the system. Note that this property would be true of any characteristic curve but the symmetry line is simple to deal with and can be understood more clearly than another randomly chosen characteristic.

As an additional consequence of the information in the characteristic profile we can see that there are no characteristics along the axes since these are essentially singularities in the velocity. This implies that there is no information transported along these lines and so it seems reasonable to expect that these are zero. This is consistent with our belief that valleys disappear through the origin and only through the origin. Returning to our definition of $\mathcal{R}(t)$ (3.4.7) we see that the terms along the axes can be assumed to be zero and we are left with,

$$\mathcal{R}(t) = \frac{2}{\langle L_{\mathrm{V}} \rangle (t)} \int_0^{\frac{\pi}{2}} \frac{\rho^\star(\theta, t)}{\sin 2\theta}\, d\theta = \frac{1}{\langle l \rangle (t)} \int_0^{\frac{\pi}{2}} \frac{\rho^\star(\theta, t)}{\sin 2\theta}\, d\theta.$$

Performing the appropriate scaling on the right hand side and considering our definition of λ we see the following,

$$\lambda = 2\mathcal{R}(t) \langle l \rangle^3 (t) = \frac{2}{\langle l \rangle (t)} \left[\int_0^{\frac{\pi}{2}} \frac{1}{\langle l \rangle^2 (t)} \frac{\hat{\rho}^\star(\theta)}{\sin 2\theta}\, d\theta \right] \langle l \rangle^3 (t) = 2 \int_0^{\frac{\pi}{2}} \frac{\hat{\rho}^\star(\theta)}{\sin 2\theta}\, d\theta. \qquad (3.4.18)$$

Condition (3.4.18) represents the boundary condition that is required to solve the the hyperbolic system. The other conditions can be thought of as constraints on the solution rather than boundary or initial information. As a result we must understand the function $\hat{\rho}^\star(\theta)$. Unfortunately there could be an infinite number of possibilities that would satisfy the condition in (3.4.18). The inability to characterise *the* form of $\hat{\rho}^\star(\theta)$ which is necessary to find the scaling solution to either (3.4.15) or (3.2.14) prevents us from pursuing this further.

We recognise from the characteristic profile that the two-point equation predicts dynamics that remains on a given characteristic for all times, by definition, and so we probe this possibility further by looking at a particular choice of characteristic and hoping to build a one-point theory instead.

3.5 Simplified Model and Associated One-point Theory

We present a particular one-point reduction of the full two-point model as a means of probing the possible solution structure. In Figure 4.14 we see the characteristic profile and in particular note that there is a curve along the symmetry line which suggests that, based on all the previous assumptions, a valley which is symmetric in side lengths will remain this way under the predicted scaled flow and associated equation. Of course the hyperbolic structure of the equation means that this is true of any characteristic curve but the symmetry line is particularly nice to probe further so we choose this one.

Since in the full 2D model a valley will remain symmetric as it evolves under the hyperbolic equation we choose to assume that *all* valleys are symmetric and live on this line. Thus the full problem is reduced to a one-dimensional problem where each valley is characterised by a single length l. By reducing the full two-dimensional velocity to a one dimensional scalar along the line in question we can probe the dynamics of this reduced problem and hope to find clues to the dynamics of the full problem.

3.5.1 Derivation

In our restricted model we consider the distribution of valleys with equal length sides and assume they remain in this set for all times, that is valleys always remain symmetric. Each valley is then characterised by the associated length of either side and we denote the distribution of such valleys $\tilde{\rho}(l,t)$. In the full two-dimensional model this corresponds to the symmetry line $l_1 = l_2$. Along this line the velocity of a valley with equal side lengths in the l_1, l_2-space is given by the vector,

$$\bar{v} := \begin{pmatrix} \left\langle \frac{1}{l} \right\rangle - \frac{1}{l} \\ \left\langle \frac{1}{l} \right\rangle - \frac{1}{l} \end{pmatrix}.$$

Note that in the full 2D model a valley which has symmetric side lengths will remain in the set of symmetric valleys based on this reduced mean-field velocity, so our reduced 1D model is, in at least this sense, consistent with the full model. We choose, therefore, to consider the effective velocity, $\tilde{v}$, of this valley in the symmetric line direction, which is found by combining the contributions from the two-dimensional vector into a single scalar quantity $\tilde{v}$,

$$\tilde{v}(l) = \sqrt{2}\left(\left\langle \frac{1}{l} \right\rangle - \frac{1}{l}\right).$$

This is the effective velocity under which a symmetric valley flows. We define the following two quantities for simplicity,

$$\alpha(t) = \int_0^\infty \frac{\tilde{\rho}(l,t)}{l}\, dl \quad \text{and} \quad \langle \tilde{l} \rangle(t) = \int_0^\infty l\tilde{\rho}(l,t)\, dl.$$

Note that $\alpha(t)$ is actually the same quantity as found in the previous derivations of the 2D model, e.g. (3.4.1), but simply using the single coordinate l which is really the distance from the origin along the symmetry line $l_1 = l_2$ in the full 2D model. We once again consider the number density and write the transport equation in $\tilde{n}(l,t)$ first before using the transformation $\tilde{n}(l,t) = \tilde{\rho}(l,t)/\langle \tilde{l} \rangle(t)$ to change the equation into one which governs $\tilde{\rho}$.

Our starting equation is then,

$$\frac{\partial \tilde{n}(l,t)}{\partial t} + \frac{\partial}{\partial l}[\tilde{v}(l)\tilde{n}(l,t)] = 0,$$

where we note, as was the case before, that the right hand side being zero indicates there are no additional sinks or sources to be accounted for. Inserting the scaling to the probability density we have,

$$\frac{\partial}{\partial t}\left(\frac{1}{\langle \tilde{l} \rangle(t)}\right)\tilde{\rho} + \frac{1}{\langle \tilde{l} \rangle(t)}\frac{\partial \tilde{\rho}}{\partial t} + \frac{1}{\langle \tilde{l} \rangle(t)}\frac{\partial}{\partial l}[\tilde{v}\tilde{\rho}] = 0,$$

$$\frac{\partial \tilde{\rho}}{\partial t} + \frac{\partial}{\partial l}[\tilde{v}\tilde{\rho}] = -\langle \tilde{l} \rangle(t)\frac{\partial}{\partial t}\left(\frac{1}{\langle \tilde{l} \rangle(t)}\right)\tilde{\rho} = \frac{\langle \tilde{l} \rangle'(t)}{\langle \tilde{l} \rangle(t)}\tilde{\rho}, \tag{3.5.1}$$

where we have suppressed arguments for simplicity. Returning to the original equation in $\tilde{n}$ and integrating with respect to l we have,

$$\frac{\partial}{\partial t}\left(\frac{1}{\langle \tilde{l} \rangle(t)}\right) - \lim_{l\to 0^+}(\tilde{v}\tilde{n}) = 0,$$

where the second term on the left hand side is simply accounting for the boundaries of the line, namely at zero and infinity, and we have assumed that $\tilde{n}(l,t)$ decays faster than $1/l$ at infinity and so has no contribution there. Continuing the calculation and substituting for $\tilde{n}$ we have,

$$-\frac{\langle \tilde{l} \rangle'(t)}{\langle \tilde{l} \rangle^2(t)} - \lim_{l\to 0^+}\left(\tilde{v}\frac{\tilde{\rho}}{\langle \tilde{l} \rangle(t)}\right) = 0,$$

$$\frac{\langle \tilde{l} \rangle'(t)}{\langle \tilde{l} \rangle(t)} = -\lim_{l\to 0^+}(\tilde{v}\tilde{\rho}). \tag{3.5.2}$$

Upon considering that $\tilde{\rho}$ must overcome the singularity in the velocity at the origin, we see that $\tilde{\rho}(0^+,t) = 0$, by contradiction. Making use of this we can then derive the following relationship,

$$\begin{aligned}\frac{\langle\tilde{l}\rangle'(t)}{\langle\tilde{l}\rangle(t)} &= -\lim_{l\to 0^+}\left(\tilde{v}\tilde{\rho}\right),\\ &= -\lim_{l\to 0^+}\sqrt{2}\left(\left\langle\frac{1}{l}\right\rangle - \frac{1}{l}\right)[\tilde{\rho}(l,t)-\tilde{\rho}(0,t)],\\ &= \sqrt{2}\frac{\partial\tilde{\rho}}{\partial l}\bigg|_{l=0^+} \qquad (3.5.3)\end{aligned}$$

3.5.2 Scaling Solution

We try a scaling solution of the following form,

$$\tilde{\rho}(l,t) = \frac{1}{\langle\tilde{l}\rangle(t)}\check{\rho}\left(z\right), \quad \text{where} \quad z := \frac{l}{\langle\tilde{l}\rangle(t)},$$

and as a consequence we can derive the following on $\alpha(t)$,

$$\alpha(t) = \int_0^\infty \frac{\tilde{\rho}(l,t)}{l}\,dl = \frac{1}{\langle\tilde{l}\rangle^2(t)}\int_0^\infty \frac{\check{\rho}(z)}{z}\,dz =: \frac{\check{\alpha}}{\langle\tilde{l}\rangle^2(t)}.$$

Scaling each term in equation (3.5.1) in turn and substituting using (3.5.2) and (3.5.3) we have,

$$\frac{\partial\tilde{\rho}}{\partial t} = -\frac{\langle\tilde{l}\rangle'(t)}{\langle\tilde{l}\rangle^2(t)}\check{\rho} - \frac{l\,\langle\tilde{l}\rangle'(t)}{\langle\tilde{l}\rangle^3(t)}\frac{\partial\check{\rho}}{\partial z} = -\frac{\langle\tilde{l}\rangle'(t)}{\langle\tilde{l}\rangle^2(t)}\left(\check{\rho} + z\frac{\partial\check{\rho}}{\partial z}\right),$$

$$\frac{\partial}{\partial l}[\tilde{v}\tilde{\rho}] = \frac{\sqrt{2}}{\langle\tilde{l}\rangle(t)}\frac{\partial}{\partial z}\left[\left(\alpha(t) - \frac{1}{z\langle\tilde{l}\rangle(t)}\right)\frac{\check{\rho}}{\langle\tilde{l}\rangle(t)}\right] = \frac{\sqrt{2}}{\langle\tilde{l}\rangle^3(t)}\frac{\partial}{\partial z}\left[\left(\check{\alpha} - \frac{1}{z}\right)\check{\rho}\right],$$

$$\frac{\langle\tilde{l}\rangle'(t)}{\langle\tilde{l}\rangle(t)}\tilde{\rho} = \frac{\sqrt{2}}{\langle\tilde{l}\rangle^3(t)}\check{\rho}\frac{\partial\check{\rho}}{\partial z}\bigg|_{z=0^+}.$$

Combining these results together and simplifying we have,

$$-\langle\tilde{l}\rangle'(t)\langle\tilde{l}\rangle(t)\frac{\partial}{\partial z}\left(z\check{\rho}\right) + \sqrt{2}\frac{\partial}{\partial z}\left[\left(\check{\alpha} - \frac{1}{z}\right)\check{\rho}\right] = \sqrt{2}\check{\rho}\frac{\partial\check{\rho}}{\partial z}\bigg|_{z=0^+}. \qquad (3.5.4)$$

A simple substitution of the scaling form into (3.5.3) shows the following,

$$\begin{aligned}\frac{\langle\tilde{l}\rangle'(t)}{\langle\tilde{l}\rangle(t)} =& \sqrt{2}\frac{\partial\tilde{\rho}}{\partial l}\bigg|_{l=0^+} = \frac{\sqrt{2}}{\langle\tilde{l}\rangle^2(t)}\frac{\partial\check{\rho}}{\partial z}\bigg|_{z=0^+},\\ \langle\tilde{l}\rangle'(t)\langle\tilde{l}\rangle(t) =& \sqrt{2}\frac{\partial\check{\rho}}{\partial z}\bigg|_{z=0^+} =: \sqrt{2}\check{\lambda}.\end{aligned}$$

Taking this definition of $\check{\lambda}$ in (3.5.4) we have,

$$\frac{\partial}{\partial z}\left[\left(\check{\lambda}z - \check{\alpha} + \frac{1}{z}\right)\check{\rho}\right] = -\check{\lambda}\check{\rho}.$$

Finally, performing the differentiation on the left hand side and collecting terms we have,

$$\left[\check{\lambda}z - \check{\alpha} + \frac{1}{z}\right]\check{\rho}' + \left[\check{\lambda} - \frac{1}{z^2}\right]\check{\rho} = -\check{\lambda}\check{\rho},$$

$$\left[\check{\lambda}z - \check{\alpha} + \frac{1}{z}\right]\check{\rho}' = \left[\frac{1}{z^2} - 2\check{\lambda}\right]\check{\rho}.$$

Thus our final problem and associated conditions are,

$$\left[\check{\lambda}z - \check{\alpha} + \frac{1}{z}\right]\check{\rho}' = \left[\frac{1}{z^2} - 2\check{\lambda}\right]\check{\rho}, \tag{3.5.5}$$

$$\check{\lambda} = \left.\frac{\partial\check{\rho}}{\partial z}\right|_{z=0^+}, \qquad \check{\alpha} = \int_0^\infty \frac{\check{\rho}(z)}{z}\,dz \qquad \int_0^\infty z\check{\rho}(z)\,dz = 1 \qquad \text{and} \qquad \int_0^\infty \check{\rho}(z)\,dz = 1.$$

Rearranging (3.5.5) further to place this in a form where we can consider solution behaviour we have,

$$\frac{\check{\rho}'}{\check{\rho}} = \frac{\left[\frac{1}{z^2} - 2\lambda\right]}{\left[\check{\lambda}z - \check{\alpha} + \frac{1}{z}\right]},$$

$$= \frac{\left[1 - 2\check{\lambda}z^2\right]}{z\left[\check{\lambda}z^2 - \check{\alpha}z + 1\right]}. \tag{3.5.6}$$

3.5.3 Condition on $\check{\lambda}$ and $\check{\alpha}$ and Associated Solution

We now consider the right hand side of equation (3.5.6) and in particular want to consider the behaviour for various values of $\check{\lambda}$ and $\check{\alpha}$. A naive inspection of the right hand side would suggest it behaves like $-1/z$ for large z but we then see that there is an issue with this outcome since we subsequently want to take the integral of the equation which gives $\check{\rho} \sim 1/z$ and so the solution, $\check{\rho}$, does not have finite integral and thus can't be a probability distribution.

Next we consider the case where the quadratic in the denominator on the right hand side has distinct roots and consider the region close to one of the roots $z^\star$ say. Near the root the right hand side will behave like $-1/(z^\star - z)$ which upon integrating we have $\check{\rho} \sim z^\star/(z^\star - z)$ which clearly blows up as we approach the root and thus again does not have finite integral.

Finally we consider the case where the quadratic has a repeated root at $z^\star$. In this circumstance we have that the right hand side will behave like $-1/(z^\star - z)^2$ near the root which upon integration gives us $\check{\rho} \sim \exp\{-1/(z^\star - z)\}$ which does not suffer from the same issues as previous attempts. As a result to get a solution with satisfies the properties of a probability distribution we must have a repeated root in the quadratic in the denominator on the right hand side of (3.5.6) or equivalently,

$$\check{\lambda} = \frac{1}{4}\check{\alpha}^2.$$

Under this condition the solution will have compact support on $[0, z^\star]$ with $z^\star = 1/\sqrt{\check{\lambda}}$ and will decay exponentially fast to the point $z^\star$. Note then that our distribution will be supported on $[0, z^\star]$ and zero elsewhere and so when we write the various integral conditions we could safely interchange ∞ with $z^\star$. With this condition on $\check{\lambda}$ and $\check{\alpha}$ we simplify (3.5.6) to the following,

$$\frac{\check{\rho}'}{\check{\rho}} = \frac{2\left(2 - \check{\alpha}^2 z^2\right)}{z(\check{\alpha} z - 2)^2}.$$

We perform a second scaling of the solution to simplify this equation. We use the following,

$$\check{\rho}(z) = \check{\alpha} R(\check{\alpha} z) =: \check{\alpha} R(s), \quad \text{where} \quad s := \check{\alpha} z.$$

As a result the four conditions attached to equation (3.5.5) become,

$$R'(s)\bigg|_{s=0^+} = \frac{1}{4}, \quad \int_0^\infty \frac{R(s)}{s}\, ds = 1, \quad \int_0^\infty sR(s)\, ds = \check{\alpha} \quad \text{and} \quad \int_0^\infty R(s)\, ds = 1.$$

Inserting the scaling form we then have,

$$\frac{R'(s)}{R(s)} = \frac{2(2 - s^2)}{s(s-2)^2},$$

which we can integrate and then subsequently take the exponential of both sides to give,

$$R(s) = \frac{As}{(s-2)^3} \exp\left(\frac{2}{s-2}\right), \tag{3.5.7}$$

where A is a constant. Making use of these two statements we can easily calculate the derivative at the origin,

$$R'(s)\bigg|_{s=0^+} = \frac{1}{4} = \frac{Ae^{-1}}{(-2)^3},$$

and as a result we have $A = -2e$. Thus our final solution is as follows,

$$R(s) = \frac{-2s}{(s-2)^3} \exp\left(1 + \frac{2}{s-2}\right) = \frac{-2s}{(s-2)^3} \exp\left(\frac{s}{s-2}\right). \tag{3.5.8}$$

We can then calculate $\check{\alpha}$ by use of the condition,

$$\int_0^\infty sR(s)\, ds = \check{\alpha},$$

where we find $\check{\alpha} = -2e\mathrm{Ei}(-1) \approx 1.19269472$, where Ei is the exponential integral.

3.5.4 Comparison of One-point Theory and Numerics

The one-point theory developed here is a very specific sub-model of the full two dimensional model but nevertheless has given us an equation which we can solve to find a one-dimensional distribution. Even though this model was derived in the physically unrealistic symmetric valley situation it is still worth considering how closely the predicted distribution matches the one-point distribution found during numerical simulations. We note that various scalings were made in the derivation above so must take care to undo this back to the distribution $\breve{\rho}(z)$ initially and further to this we must scale by a factor of $\sqrt{2}$ since this distribution is really on the diagonal line in the z_1, z_2-plane so is stretched by this geometric factor.

In Figure 4.15 we see the distribution from the data $\varrho(z)$ plotted together with the appropriately scaled $\breve{\rho}(z)$. Perhaps unsurprisingly these are quite distinct in character with widely differing peak locations and approach to the origin. The derivation of the one-point prediction is really completely distinct from considering the true one-point distribution and so on this basis wouldn't expect to see a neat match in the output.

An interesting point to note comes from considering the dynamical system proposed in [**89**] as a means of studying the CKS equation, which we have detailed in Chapter 3. In a subsequent paper, [**11**], the authors analysed the dynamical system proposed and predicted the one-point distribution from a similar Fokker-Planck type derivation to our model here. Despite coming from a fundamentally different problem and indeed an entirely different derivation it is perhaps surprising to see that the results found are qualitatively the same. In fact, by setting $\check{\lambda} = 1$ in (3.5.6) and recalling the definition of $\breve{\alpha}$ we have exactly the same result as found in [**11**]. We therefore consider these two problems to be of the same universality class as each other. Further to this we note that the distributions predicted here and in [**11**] also qualitatively match the one-point prediction made in the LSW theory, [**68**, **118**], and so interestingly this too is considered to be of the same universality class.

We note here once again that our one-point derivation, whilst well justified, is not completely faithful to the original problem and so we may have inadvertently pushed our hill-valley model into a different regime. It is interesting, however, to note that this reduction to a one-point model ties this hill-valley model to a completely different dynamical system. Further to this we perhaps see evidence that care must be taken in the models and approaches used to analyse coarsening dynamical systems, since there is always the possibility of removing important information when taking the various mean-field approximations or indeed model simplifications as seen here.

4 Overall Conclusions

We have taken a faceted system together with an associated dynamics which displays coarsening behaviour. Significant numerical data on the system was generated which, in particular, showed the one and two-point distributions exhibiting the dynamic scaling property. Exploring this further we tried to predict the statistical information within the system by considering the effective behaviour and developing evolution equations for the distributions. Consideration of the behaviour displayed led us to believe that this model could be considered a kind of two-dimensional LSW problem.

In the two-point case we were able to derive an evolution equation but we were unable to fully understand the behaviour at the origin which represents the single sink within the system. The difficulty in finding the angular profile $\hat{\rho}^{\star}(\theta)$ and associated numbers λ and $\hat{\alpha}$ such that the two-point distribution predicted by (3.4.15) satisfied the corresponding constraints, and where the function $\hat{\rho}^{\star}(\theta)$ had been rationally derived means that for the time-being predictions of the two-point distribution are incomplete. The problem has, however, generated a wealth of interesting analytical questions worthy of further study.

We were able to reduce the full two-point problem to a simple one-point one which we were able to solve for. A direct calculation of a one-point evolution equation seemed impractical at first but through exploring the two-point model we were able to find a well justified prediction of a one-point distribution. This one-point problem forced us to constrain the parameters in the equation, $\check{\lambda} = \check{\alpha}^2/4$, and a similar restriction in the two-point case appears to lead to a saddle-node point and a vacuum region, see Figure 4.14, analogous to the compact support seen in the one-point case.

Further analysis of the two-point problem may lead to an understanding of both any constraints on the parameters in the equation and the origin behaviour. With these better understood it is possible that a solution to the two-point evolution equation that we have derived could be found.

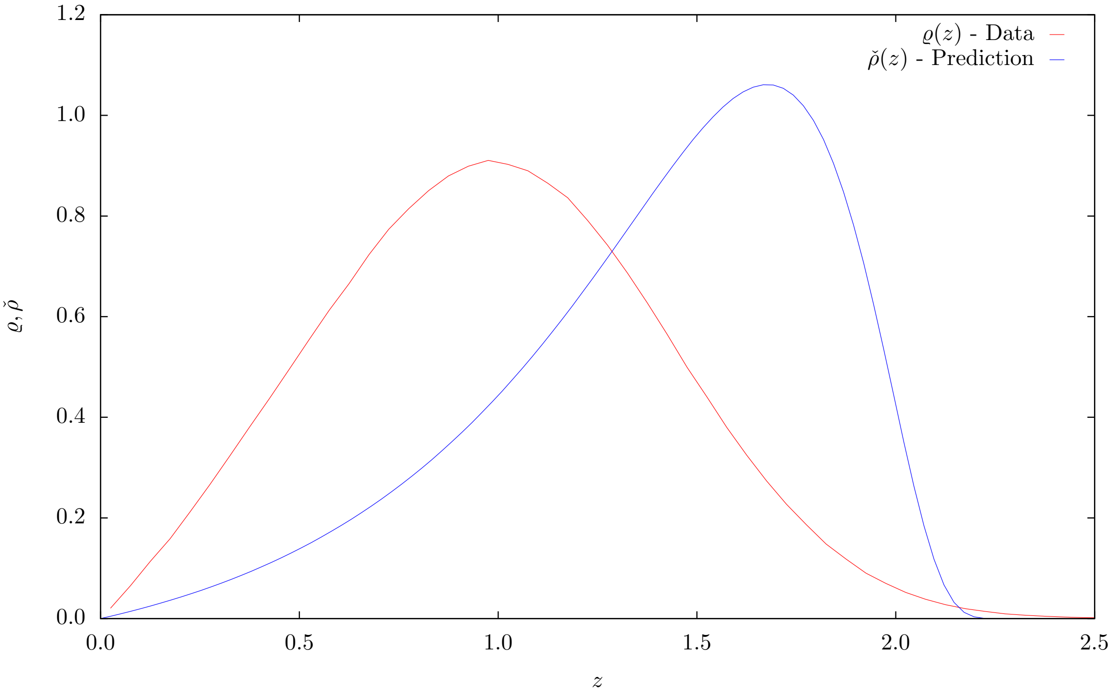


Figure 4.15 – Comparison of one-point prediction $\check{\rho}(z)$, derived from (3.5.8), and simulation data $\varrho(z)$. Note the significantly different approach to the origin and the peak location.

References

[1] D.J. Aldous. Deterministic and stochastic models for coalescence (aggregation and coagulation): A review of the mean-field theory for probabilists. *Bernoulli*, 5(1):3–48, February 1999.

[2] E.J. Allen and P. Bastien. On coagulation and the stellar mass-spectrum. *Astrophysical Journal*, 452(2):652–670, October 1995.

[3] J.C. Arce, C. Schaadt, and H.J. Bart. Extraction with spinodal decomposition - experiment and simulation. *Chemical Engineering & Technology*, 29(4):487–494, April 2006.

[4] M.J. Baines, M.E. Hubbard, and P.K. Jimack. A moving mesh finite element algorithm for the adaptive solution of time-dependent partial differential equations with moving boundaries. *Applied Numerical Mathematics*, 54(3-4):450–469, AUG 2005. 16th Chemnitz Finite Element Symposium, Ehrenfriedersdorf, GERMANY, SEP 22-24, 2003.

[5] G.S. Bales and A. Zangwill. Morphological instability of a terrace edge during step-flow growth. *Physical Review B*, 41(9):5500–5508, March 1990.

[6] AL Barabasi, R Albert, and H Jeong. Mean-field theory for scale-free random networks. *Physica a*, 272(1-2):173–187, October 1999.

[7] F.S. Bates. Polymer-polymer phase-behavior. *Science*, 251(4996):898–905, February 1991.

[8] G. Beckett and J.A. Mackenzie. On a uniformly accurate finite difference approximation of a singularly perturbed reaction–diffusion problem using grid equidistribution. *Journal of Computational and Applied Mathematics*, 131:381–405, 2001.

[9] G. Beckett, J.A. Mackenzie, A. Ramage, and D.M. Sloan. On the numerical solution of one-dimensional PDEs using adaptive methods based on equidistribution. *Journal of Computational Physics*, 167:372–392, 2001.

[10] I. Bena, C.Q. Misbah, and A. Valance. Nonlinear evolution of a terrace edge during step-flow growth. *Physical Review B*, 47(12):7408–7419, March 1993.

[11] D. ben-Avraham, O. Gromenko, and P. Politi. Deterministic reaction models with power-law forces. *Journal of Physics A: Mathematical and Theoretical*, 42(49), 2009.

[12] A.J. Benson, M. Kamionkowski, and S.H. Hassani. Self-consistent theory of halo mergers. *Monthly Notices of the Royal Astronomical Society*, 357(3):847–858, March 2005.

[13] A. W. Bowman and Adelchi Azzalini. *Applied smoothing techniques for data analysis: The kernel approach with S-Plus illustrations.* Oxford University Press, 1997.

[14] A.J. Bray. Theory of phase-ordering kinetics. *Advances in Physics*, 43(3):357–459, 1994.

[15] C.J. Budd and M.D. Piggott. The geometric integration of scale-invariant ordinary and partial differential equations. *Journal of Computational and Applied Mathematics*, 128:399–422, 2001.

[16] C.J. Budd and J.F. Williams. Moving mesh generation using the parabolic Monge-Ampère equation. *Siam Journal On Scientific Computing*, 31(5):3438–3465, 2009.

[17] J.W. Cahn. Spinodal decomposition in cubic crystals. *Acta Metallurgica*, 10(MAR), 1962.

[18] J.W. Cahn. Phase separation by spinodal decomposition in isotropic systems. *Journal of Chemical Physics*, 42(1), 1965.

[19] J.W. Cahn and J.E. Hilliard. Free energy of a nonuniform system. 1 Interfacial free energy. *Journal of Chemical Physics*, 28(2):258–267, 1958.

[20] W.M. Cao, W.Z. Huang, and R.D. Russell. A moving mesh method based on the geometric conservation law. *SIAM Journal on Scientific Computing*, 24(1):118–142, AUG 21 2002.

[21] J. Carr and R.L. Pego. Self-similarity in a coarsening model in one dimension. *Proceedings of the Royal Society A*, 436:569–583, 1992.

[22] J. Carr and R.L. Pego. Self-similarity in a cut-and-paste model of coarsening. *Proceedings of the Royal Society of London Series A-Mathematical Physical and Engineering Sciences*, 456(1997):1281–1290, May 2000.

[23] P. Cermelli and M.E. Jabbour. Possible mechanism for the onset of step-bunching instabilities during the epitaxy of single-species crystalline films. *Physical Review B*, 75(16):165409, April 2007.

[24] P. Cermelli and M.E. Jabbour. Step bunching during the epitaxial growth of a generic binary-compound thin film. *Journal of the Mechanics and Physics of Solids*, 58(5):810–827, May 2010.

[25] C. Clarke. *The Science of Ice Cream.* The Royal Society of Chemistry, 2004.

[26] R. Cuerno and A.L. Barabasi. Dynamic scaling of ion-sputtered surfaces. *Physical Review Letters*, 74(23):4746–4749, June 1995.

[27] S.B. Dai and R.L. Pego. Universal bounds on coarsening rates for mean-field models of phase transitions. *Siam Journal On Mathematical Analysis*, 37(2):347–371, 2005.

[28] C. de Boor. Good approximation by splines with variable knots. II. In *Conference on the Numerical Solution of Differential Equations*, 1974.

[29] P.G. Degennes. Dynamics of fluctuations and spinodal decomposition in polymer blends. *Journal of Chemical Physics*, 72(9):4756–4763, 1980.

[30] B. Derrida, C. Godrèche, and I. Yekutieli. Scale-invariant regimes in one-dimensional models of growing and coalescing droplets. *Physical Review A*, 44(10), November 1991.

[31] E.A. Dorfi and L.O'C. Drury. Simple adaptive grids for 1-D initial value problems. *Journal of Computational Physics*, 69:175–195, 1987.

[32] G. Ehrlich and F.G. Hudda. Atomic view of surface self-diffusion - Tungsten on Tungsten. *Journal of Chemical Physics*, 44(3), 1966.

[33] C.L. Emmott and A.J. Bray. Coarsening dynamics of a one-dimensional driven Cahn-Hilliard system. *Physical Review E*, 54(5):4568–4575, November 1996.

[34] M. Escobedo and S. Mischler. Scalings for a ballistic aggregation equation. *Journal of Statistical Physics*, 141(3):422–458, November 2010.

[35] F. Family. Dynamic scaling and phase-transitions in interface growth. *Physica A*, 168(1):561–580, September 1990.

[36] F. Family and P. Meakin. Kinetics of droplet growth-processes - simulations, theory, and experiments. *Physical Review A*, 40(7):3836–3854, October 1989.

[37] J.A. Floro, M.B. Sinclair, E. Chason, L.B. Freund, R.D. Twesten, R.Q. Hwang, and G.A. Lucadamo. Novel SiGe island coarsening kinetics: Ostwald ripening and elastic interactions. *Physical Review Letters*, 84(4):701–704, January 2000.

[38] S.K. Friedlander. *Smoke, Dust, and Haze: Fundamentals of Aerosol Dynamics*. Oxford University Press, USA, March 2000.

[39] T. Frisch and A.Verga. Effect of step stiffness and diffusion anisotropy on the meandering of a growing vicinal surface. *Physical Review Letters*, 96(17), 2006.

[40] T. Frisch and A. Verga. Nonlinear evolution of the step meandering instability of a growing crystal surface. *Physica D*, 235:15–20, 2007.

[41] H. Furukawa. A dynamic scaling assumption for phase-separation. *Advances In Physics*, 34(6):703–750, Nov-Dec 1985.

[42] R.M. Furzeland, J.G. Verwer, and P.A. Zegeling. A numerical study of three moving-grid methods for one-dimensional partial differential equations which are based on the method of lines. *Journal of Computational Physics*, 89:349–388, 1990.

[43] T. Gallay and A. Mielke. Convergence results for a coarsening model using global linearization. *Journal of Nonlinear Science*, 13:311–346, 2003.

[44] F. Gillet, Z. Csahok, and C. Misbah. Continuum nonlinear surface evolution equation for conserved step-bunching dynamics. *Physical Review B*, 63, 2001.

[45] K.B. Glasner and T.P. Witelski. Coarsening dynamics of dewetting films. *Physical Review E*, 67(1):016302, January 2003.

[46] A.A. Golovin, A.A. Nepomnyashchy, S.H. Davis, and M.A. Zaks. Convective Cahn-Hilliard models: From coarsening to roughening. *Physical Review Letters*, 86(8):1550–1553, February 2001.

[47] M.B. Gratton and T.P. Witelski. Coarsening of unstable thin films subject to gravity. *Physical Review E*, 77(1):016301, January 2008.

[48] M.B. Gratton and T.P. Witelski. Transient and self-similar dynamics in thin film coarsening. *Physica D-Nonlinear Phenomena*, 238(23-24):2380–2394, December 2009.

[49] M.E. Gurtin and P.W. Voorhees. On the effects of elastic stress on the motion of fully faceted interfaces. *Acta Materialia*, 46(6):2103–2112, March 1998.

[50] R. Hannes. *The Fokker-Planck equation: methods of solution and applications.* Springer, 1996.

[51] D.F. Hawken, J.J. Gottlieb, and J.S. Hansen. Review of some adaptive node-movement techniques in finite-element and finite-difference solutions of partial differential equations. *Journal of Computational Physics*, 95:254–302, 1991.

[52] J.R. Heffelfinger and C.B. Carter. Mechanisms of surface faceting and coarsening. *Surface Science*, 389(1-3):188–200, November 1997.

[53] C-L. Horne. Probing the dynamic scaling hypothesis in 1-D coarsening particle systems. Master's thesis, University of Glasgow, 2010.

[54] W. Huang. Practical aspects of formulation and solution of moving mesh partial differential equations. *Journal of Computational Physics*, 171:753–775, 2001.

[55] W. Huang, Y. Ren, and R.D. Russell. Moving mesh methods based on moving mesh partial-differential equations. *Journal of Computational Physics*, 113(2):279–290, AUG 1994.

[56] W. Huang, Y. Ren, and R.D. Russell. Moving mesh partial differential equations (MMPDEs) based on the equidistribution principle. *SIAM Journal on Numerical Analysis*, 31(3):709–730, June 1994.

[57] W. Huang and R.D. Russell. Adaptive mesh movement — the MMPDE approach and its applications. *Journal of Computational and Applied Mathematics*, 128:383–398, 2001.

[58] S. Kempf, S. Pfalzner, and T.K. Henning. N-particle-simulations of dust growth - I. Growth driven by Brownian motion. *Icarus*, 141(2):388–398, October 1999.

[59] T. Kiorboe. Formation and fate of marine snow: small-scale processes with large-scale implications. *Scientia Marina*, 65:57–71, September 2001.

[60] R.V. Kohn and F. Otto. Upper bounds on coarsening rates. *Communications In Mathematical Physics*, 229(3):375–395, September 2002.

[61] R.V. Kohn and X.D. Yan. Upper bound on the coarsening rate for an epitaxial growth model. *Communications On Pure and Applied Mathematics*, 56(11):1549–1564, November 2003.

[62] J. Krug. Power laws in surface physics: the deep, the shallow and the useful. *Physica A-Statistical Mechanics and Its Applications*, 340(4):647–655, September 2004.

[63] J. Krug. Kinetic pattern formation at solid surfaces. *Chapter from 'Collective dynamics of nonlinear and disordered systems'*, pages 5–37, 2005.

[64] Y. Kuramoto and T. Tsuzuki. Persistent propagation of concentration waves in dissipative media far from thermal equilibrium. *Progress of Theoretical Physics*, 55(2):356–369, 1976.

[65] J.S. Langer. Theory of spinodal decomposition in alloys. *Annals of Physics*, 65(1-2), 1971.

[66] J.S. Langer, M. Bar-on, and H.D. Miller. New computational method in the theory of spinodal decomposition. *Phys. Rev. A*, 11(4):1417–1429, Apr 1975.

[67] F. Leroy, D. Karashanova, M. Dufay, J.-M. Debierre, T. Frisch, J.-J. Metois, and P. Mueller. Step bunching to step-meandering transition induced by electromigration on Si(111) vicinal surface. *Surface Science*, 603(3):507–512, February 2009.

[68] I.M. Lifshitz and V.V. Slyozov. The kinetics of precipitation from supersaturated solid solutions. *Journal of Physics and Chemistry of Solids*, 19(1-2):35–50, 1961.

[69] L. Lindfors, P. Skantze, U. Skantze, M. Rasmusson, A. Zackrisson, and U. Olsson. Amorphous drug nanosuspensions. 1. Inhibition of Ostwald ripening. *Langmuir*, 22(3):906–910, January 2006.

[70] B.D. Marsh. Crystal size distribution (CSD) in rocks and the kinetics and dynamics of crystallization 1. Theory. *Contributions To Mineralogy and Petrology*, 99(3):277–291, 1988.

[71] I.N. McCave. Size spectra and aggregation of suspended particles in the deep ocean. *Deep-Sea Research Part A-Oceanographic Research Papers*, 31(4):329–352, 1984.

[72] G. Menon, B. Niethammer, and R.L. Pego. Dynamics and self-similarity in min-driven clustering. *Transactions of the American Mathematical Society*, 362(12):6591–6618, December 2010.

[73] C. Misbah, O. Pierre-Louis, and Y. Saito. Crystal surfaces in and out of equilibrium: A modern view. *Reviews of Modern Physics*, 82(1):981–1040, Jan-Mar 2010.

[74] A. Mock, D.A. Jerram, and C. Breitkreuz. Using quantitative textural analysis to understand the emplacement of shallow-level rhyolitic laccoliths - A case study from the Halle Volcanic Complex, Germany. *Journal of Petrology*, 44(5):833–849, May 2003.

[75] W.W. Mullins. The statistical self-similarity hypothesis in grain-growth and particle coarsening. *Journal of Applied Physics*, 59(4):1341–1349, February 1986.

[76] W.W. Mullins. The statistical particle growth law in self-similar coarsening. *Acta Metallurgica Et Materialia*, 39(9):2081–2090, September 1991.

[77] H.S. Niwa. School size statistics of fish. *Journal of Theoretical Biology*, 195(3):351–361, December 1998.

[78] S.A. Norris, S.H. Davis, S.J. Watson, and P.W. Voorhees. Faceted interfaces in directional solidification. *Journal of Crystal Growth*, 310(2):414–427, January 2008.

[79] S.A. Norris and S.J. Watson. Geometric simulation and surface statistics of coarsening faceted surfaces. *Acta Materialia*, 55(19):6444–6452, November 2007.

[80] S.A. Norris and S.J. Watson. A mean-field theory for coarsening faceted surfaces. *arXiv:0910.2212v1*, 2009.

[81] S. Okuzumi, H. Tanaka, T. Takeuchi, and M. Sakagami. Electrostatic barrier against dust growth in protoplanetary disks. I. Classifying the evolution of size distribution. *Astrophysical Journal*, 731(2):95, April 2011.

[82] W. Ostwald. *Lehrbruck der Allgemeinen Chemie*, volume 2. Leipzig, 1896.

[83] F. Otto, T. Rump, and D. Slepcev. Coarsening rates for a droplet model: Rigorous upper bounds. *Siam Journal On Mathematical Analysis*, 38(2):503–529, 2006.

[84] D. Pardo, L. Demkowicz, C. Torres-Verdin, and L. Tabarovsky. A goal-oriented *hp*-adaptive finite element method with electromagnetic applications. Part I: Electrostatics. *International Journal For Numerical Methods In Engineering*, 65(8):1269–1309, February 2006.

[85] V.B. Patravale, A.A. Date, and R.M. Kulkarni. Nanosuspensions: a promising drug delivery strategy. *Journal of Pharmacy and Pharmacology*, 56(7):827–840, July 2004.

[86] K. Pesz and G.J. Rodgers. Kinetics of growing and coalescing droplets. *Journal of Physics A-Mathematical and General*, 25(4):705–713, February 1992.

[87] O. Pierre-Louis, G. Danker, J. Chang, K. Kassner, and C. Misbah. Nonlinear dynamics of vicinal surfaces. *Journal of Crystal Growth*, 275(1-2):56–64, February 2005.

[88] O. Pierre Louis and C. Misbah. Out-of-equilibrium step meandering on a vicinal surface. *Physical Review Letters*, 76(25):4761–4764, June 1996.

[89] P. Politi and D. ben-Avraham. From the conserved Kuramoto-Sivashinsky equation to a coalescing particles model. *Physica D*, 238:156–161, 2009.

[90] P. Politi, G. Grenet, A. Marty, A. Ponchet, and J. Villain. Instabilities in crystal growth by atomic or molecular beams. *Physics Reports-Review Section of Physics Letters*, 324(5-6):271–404, February 2000.

[91] L. Ratke and P.W. Voorhees. *Growth and coarsening: Ostwald ripening in material processing*. Springer-Varlag, 2002.

[92] G. Reiter. Dewetting of thin polymer-films. *Physical Review Letters*, 68(1):75–78, January 1992.

[93] Y. Ren and R.D. Russell. Moving mesh techniques based upon equidistribution and their stability. *SIAM Journal on Scientific Computing*, 13(6):1265–1286, November 1992.

[94] F.M. Ross, J. Tersoff, and R.M. Tromp. Coarsening of self-assembled Ge quantum dots on Si(001). *Physical Review Letters*, 80(5):984–987, February 1998.

[95] F. Sanchez, F. Rigato, R. Bachelet, and J. Fontcuberta. Formation of step bunching in the epitaxial growth of SrRuO3 thin films. *Journal of Crystal Growth*, 310(14):3348–3350, July 2008.

[96] T.E.W. Schuman. Theoretical aspects of the size distribution of fog particles. *Quarterly Journal of the Royal Meterological Society*, 66:195–207, 1940.

[97] R.L. Schwoebel. Step motion on crystal surfaces 2. *Journal of Applied Physics*, 40(2), 1969.

[98] J.H. Seinfeld and S.N. Pandis. *Atmospheric Chemistry and Physics: From Air Pollution to Climate Change.* Wiley-Blackwell, November 1997.

[99] D.K. Shangguan and J.D. Hunt. Dynamical study of the pattern-formation of faceted cellular array growth. *Journal of Crystal Growth*, 96(4):856–870, August 1989.

[100] J. SIlk and S.D. White. The development of structure in the expanding universe. *The Astrophysical Journal*, 223, 1978.

[101] Jeffrey S. Simonoff. *Smoothing Methods in Statistics.* Springer, 1996.

[102] G.I. Sivashinsky. Nonlinear analysis of hydrodynamic instability in laminar flames I. Derivation of basic equations. *Acta Astronautica*, 4(11-12):1177 – 1206, 1977.

[103] P. Smereka, X.Q. Li, G. Russo, and D.J. Srolovitz. Simulation of faceted film growth in three dimensions: microstructure, morphology and texture. *Acta Materialia*, 53(4):1191–1204, February 2005.

[104] C. Solans, P. Izquierdo, J. Nolla, N. Azemar, and M.J. Garcia-Celma. Nano-emulsions. *Current Opinion in Colloid & Interface Science*, 10(3-4):102–110, Oct 2005.

[105] J. Stangl, V. Holy, and G. Bauer. Structural properties of self-organized semiconductor nanostructures. *Reviews of Modern Physics*, 76(3):725–783, July 2004.

[106] T. Tang and J. Xu, editors. *Adaptive Computations: Theory and Algorithms.* Science Press Beijing, 2007.

[107] J. Tersoff, Y.H. Phang, Z.Y. Zhang, and M.G. Lagally. Step-bunching instability of vicinal surfaces under stress. *Physical Review Letters*, 75(14):2730–2733, October 1995.

[108] J. Tersoff and R.M. Tromp. Shape transition in growth of strained islands - spontaneous formation of quantum wires. *Physical Review Letters*, 70(18):2782–2785, May 1993.

[109] V. Tonchev, B. Ranguelov, H. Omi, and A. Pimpinelli. Scaling and universality in models of step bunching: the 'C+-C-' model. *European Physical Journal B*, 73(4):539–546, February 2010.

[110] P.G.J. Vandongen and M.H. Ernst. Dynamic scaling in the kinetics of clustering. *Physical Review Letters*, 54(13):1396–1399, 1985.

[111] P.G. Vekilov and J.I.D. Alexander. Dynamics of layer growth in protein crystallization. *Chemical Reviews*, 100(6):2061–2089, June 2000.

[112] R.D. Vengrenovich, Y.V. Gudyma, and S.V. Yarema. Ostwald ripening of quantum-dot nanostructures. *Semiconductors*, 35(12):1378–1382, 2001.

[113] A.D. Verga. Anisotropic dynamics of a vicinal surface under the meandering step instability. *Physical Review B*, 80(17):174115, November 2009.

[114] T. Vicsek and F. Family. Dynamic scaling for aggregation of clusters. *Physical Review Letters*, 52(19):1669–1672, 1984.

[115] S.A. Vitale and J.L. Katz. Liquid droplet dispersions formed by homogeneous liquid-liquid nucleation: 'the ouzo effect'. *Langmuir*, 19(10):4105–4110, May 2003.

[116] M. von Smoluchowski. Drei vorträge über diffusion, brownsche molekularbewegung und koagulation von kolloidteilchen. *Physikalische Zeitschrift*, 17:557–571, 1916.

[117] P.W. Voorhees. The theory of Ostwald ripening. *Journal of Statistical Physics*, 38(1-2):231–252, 1985.

[118] C. Wagner. Theorie der alterung von niederschlagen durch umlosen (ostwald-reifung). *Zeitschrift Fur Elektrochemie*, 65(7-8):581–591, 1961.

[119] S.J. Watson. Crystal growth, coarsening and the convective Cahn-Hilliard equation. *Chapter from 'Free Boundary Problems: Theory and Applications'*, 147:329–342, 2003.

[120] S.J. Watson. Emergent symmetries and equivariant universality in driven coarsening of nanofaceted crystals. *Preprint*, 2011.

[121] S.J. Watson and S.A. Norris. Scaling theory and morphometrics for a coarsening multiscale surface, via a principle of maximal dissipation. *Physical Review Letters*, 96(17):176103, May 2006.

[122] S.J. Watson, F. Otto, B.Y. Rubinstein, and S.H. Davis. Coarsening dynamics of the convective Cahn-Hilliard equation. *Physica D-Nonlinear Phenomena*, 178(3-4):127–148, April 2003.

[123] J.A.D. Wattis. An introduction to mathematical models of coagulation-fragmentation processes: A discrete deterministic mean-field approach. *Physica D-Nonlinear Phenomena*, 222(1-2):1–20, October 2006.

[124] C. Wild, N. Herres, and P. Koidl. Texture formation in polycrystalline diamond films. *Journal of Applied Physics*, 68(3):973–978, August 1990.

[125] Q.H. Xie, A. Madhukar, P. Chen, and N.P. Kobayashi. Vertically self-organized InAs quantum box islands on GaAs(100). *Physical Review Letters*, 75(13):2542–2545, September 1995.

[126] D.-H. Yeon, P.-R. Cha, J.S. Lowengrub, A. Voigt, and K. Thornton. Linear stability analysis for step meandering instabilities with elastic interactions and Ehrlich-Schwoebel barriers. *Physical Review E*, 76, 2007.

[127] P.A. Zegeling. r-refinement for evolutionary PDEs with finite elements or finite differences. *Applied Numerical Mathematics*, 26:97–104, 1998.

[128] P.A. Zegeling. Theory and application of adaptive moving grid methods. *Chapter from 'Adaptive Computations: Theory and Algorithms'*, 2007.

[129] J.Z. Zhu, L.Q. Chen, J. Shen, and V. Tikare. Coarsening kinetics from a variable-mobility Cahn-Hilliard equation: Application of a semi-implicit Fourier spectral method. *Physical Review E*, 60(4):3564–3572, October 1999.

[130] M. Zinkeallmang, L.C. Feldman, and M.H. Grabow. Clustering on surfaces. *Surface Science Reports*, 16(8):377–463, 1992.